

ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ
ВОЛГОГРАДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

ИЗВЕСТИЯ

ВОЛГОГРАДСКОГО
ГОСУДАРСТВЕННОГО
ТЕХНИЧЕСКОГО
УНИВЕРСИТЕТА

Серия

АКТУАЛЬНЫЕ ПРОБЛЕМЫ УПРАВЛЕНИЯ,
ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ
И ИНФОРМАТИКИ В ТЕХНИЧЕСКИХ СИСТЕМАХ

Выпуск 9

№ 11(71) Межвузовский сборник научных статей 2010
Издается с января 2004 г.



Волгоград
2010

Учредитель: ГОУ высшего профессионального образования
"Волгоградский государственный технический университет"

Сборник зарегистрирован в Управлении регистрации и лицензионной работы в сфере массовых коммуникаций федеральной службы по надзору за соблюдением законодательства в сфере массовых коммуникаций и охране культурного наследия ПИ № ФС77-25660 от 13 сентября 2006 г.

Главный редактор сборника "Известия ВолгГТУ"

д-р хим. наук, проф., член-корр. РАН *И. А. Новаков*

Редакционная коллегия серии: *В. А. Камаев* (научный редактор серии)
С. А. Фоменков (ответственный секретарь)

Аверченков В. И., д.т.н., профессор БрГТУ
Бершадский А. М., д.т.н., профессор ПензГУ
Бутенко Л. Н., д.х.н., профессор ВолгГТУ
Волкова Г. А., д.т.н., профессор МГТУ СТАНКИН
Горобцов А. С., д.т.н., профессор ВолгГТУ
Дворянкин А. М., д.т.н., профессор ВолгГТУ
Заболеева-Зотова А. В., д.т.н., профессор ВолгГТУ
Кравец А. Г., д.т.н. ВолгГТУ
Курейчик В. М., д.т.н., профессор ТагГРТУ
Лукьянов В. С., д.т.н., профессор ВолгГТУ
Петрова И. Ю., д.т.н., профессор АстрГТУ
Проталинский О. М., д.т.н., профессор АстрГТУ
Фоменков С. А., д.т.н., профессор ВолгГТУ
Шилин А. Н., д.т.н., профессор ВолгГТУ

Печатается по решению редакционно-издательского совета
Волгоградского государственного технического университета

Известия Волгоградского государственного технического университета: межвуз. сб. науч. ст. № 11(71) / ВолгГТУ. – Волгоград: УИНЛ ВолгГТУ, 2010. – 160 с. – (Сер. Актуальные проблемы управления, вычислительной техники и информатики в технических системах. Вып. 9).
ISBN 978-5-9948-0607-4

Содержит работы ученых высших учебных заведений России по следующей тематике: математическое моделирование и численные методы, системный анализ и обработка информации, управление в социальных и экономических системах, автоматизированное проектирование, построение сетей и защита информации.

Ил. 74. Табл. 12. Библиогр.: 235 назв.

СОДЕРЖАНИЕ

I. МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ И ЧИСЛЕННЫЕ МЕТОДЫ

<i>Вершинина И. П.</i>	Математическая модель динамики неньютоновской жидкости во вращающемся цилиндрическом канале.....	5
<i>Галишикова В. В.</i>	Вычислительные методы решения почти сингулярных систем линейных алгебраических уравнений в алгоритмах геометрически нелинейного анализа устойчивости стержневых систем.....	9
<i>Голубятникова М. В.</i>	Моделирование процесса экстракции на основе модифицированного интервального метода расчета.....	13
<i>Гончаров Ал-др Аф., Гончаров Ант. Ал-др.</i>	Моделирование процесса самоторможения клиновых механизмов свободного хода.....	16
<i>Иванюк В. А., Егорова И. Е.</i>	Разработка программной реализации генетического алгоритма для решения задач многомерной оптимизации.....	23
<i>Крушель Е. Г., Степанченко И. В., Степанченко О. В.</i>	Об организации вычислений в цифровых управляющих системах с двойной шкалой времени.....	26
<i>Кумунжиев К. В., Черненко В. Е., Малыханов А. А.</i>	Алгоритм построения траектории движения агента в модели транспортной системы.....	29
<i>Чернышев В. В.</i>	Моделирование динамики шагающей машины с цикловыми движителями как системы твердых тел с упругодиссипативными связями.....	32
<i>Шаронов Н. Г.</i>	Синтез алгоритма управления движением шагающей машины «Восьминог» в зависимости от параметров локального препятствия.....	35
<i>Шаронов Н. Г., Калинин Я. В.</i>	Синтез алгоритма управления приводом цикловых движителей шагающей машины в особых условиях.....	40

II. СИСТЕМНЫЙ АНАЛИЗ И ОБРАБОТКА ИНФОРМАЦИИ

<i>Артемов Е. М., Привалов О. О., Степанченко И. В., Петров В. О.</i>	Применение алгоритмов машинного зрения для получения панорамного изображения медико-биологического препарата при микроскопии.....	44
<i>Бутенко Дм. В., Бутенко Л. Н., Кошечкин Я. С., Козлов М. С., Прошин А. Д.</i>	Концептуальный анализ системы антикризисного управления.....	47
<i>Гребенюк П. Е., Зотов П. В., Чмутин А. М., Шарипов Р. Р.</i>	Программное вмешательство в цифровые изображения и его идентификация.....	49
<i>Камаев В. А., Кудряшов П. П., Стец А. А.</i>	Использование контурного анализа в задаче контроля качества абразивных материалов.....	53
<i>Матлин А. О., Фоменков С. А.</i>	Построение автоматизированной системы создания интерактивных тренажеров.....	57
<i>Орлова Ю. А.</i>	Расширенная нечеткая атрибутивная грамматика над фреймовой структурой текста технического задания.....	60
<i>Розалиев В. Л., Бобков А. С., Федоров О. С.</i>	Применение нейронных сетей и грануляции при построении автоматизированной системы определения эмоциональной реакции человека.....	63
<i>Щербаков М. В., Щербакова Н. Л., Бребельс А.</i>	Методика выбора значимых параметров для краткосрочного прогнозирования энергопотребления.....	68

<i>Щербаков М. В., Щербакова Н. Л., Панченко Д. П., Бребельс А., Тюков А. П., Аль-Гунаид М. А.</i>	
Специфика применения интеллектуальных моделей анализа данных для повышения энергетической эффективности.....	72
III. УПРАВЛЕНИЕ В СОЦИАЛЬНЫХ И ЭКОНОМИЧЕСКИХ СИСТЕМАХ	
<i>Брумиштейн Ю. М., Ватулева В. П., Мангаладзе Н. Ф., Тналиева В. Я.</i>	
Анализ некоторых нетрадиционных возможностей построения и использования тестовых заданий с предлагаемыми наборами ответов.....	77
<i>Брумиштейн Ю. М., Горюнова Т. А., Пугина Н. Н.</i>	
Анализ методов и математических моделей обеспечения вуза кадрами высшей квалификации.....	81
<i>Брумиштейн Ю. М., Захаров Д. А., Аксенова Ю. Ю.</i>	
Анализ возможных подходов к оценке уровней оснащённости медоборудованием стационарных медучреждений.....	85
<i>Дворянкин А. М., Кривенко Е. В.</i>	
Обзор свободно распространяемых средств автоматизированного тестирования ВЕБ-приложений.....	88
<i>Жукова И. Г., Кульцова М. Б., Аникин А. В.</i>	
Модели и инструментальные средства интеграции распределенных ресурсов в открытых образовательных сетях.....	91
<i>Ковалев И. В.</i>	
Программный комплекс автоматизации экологического мониторинга загрязнения атмосферы.....	95
<i>Кожеевникова Н. Ю.</i>	
Влияние Болонского процесса на преобразование учебных бизнес-процессов и распределение ресурсов вуза.....	98
<i>Лаптев В. В., Сербин В. И.</i>	
Учет времени при оценивании результатов автоматизированного контроля....	102
<i>Литовкин Д. В., Сычев О. А.</i>	
Библиотека модульного тестирования, используемая при обучении программированию.....	106
<i>Ужва А. Ю., Жукова И. Г., Кульцова М. Б.</i>	
Онтологическая модель предметной области, обеспечивающая поддержку рассуждений по прецедентам для поиска образовательных ресурсов.....	110
<i>Харитонов И. М., Скрипченко Е. Н.</i>	
Применение контент-анализа для оценки терминологической связи между учебными дисциплинами.....	114
<i>Шабалина О. А., Катаев А. В., Воробкалов П. Н.</i>	
Обучение разработчиков программного обеспечения: применение компьютерных игр и процесса их разработки.....	117
IV. АВТОМАТИЗИРОВАННОЕ ПРОЕКТИРОВАНИЕ	
<i>Голубятникова М. В.</i>	
Применение алгоритма выбора оптимального технологического режима при нечетко выраженных экспертных оценках.....	125
<i>Гопта Е. А., Фоменков С. А., Карачунова Г. А.</i>	
Автоматизация процесса линейного синтеза физического принципа действия	129
<i>Гулаков В. К., Трубаков А. О.</i>	
Проблема большого объема векторов характеристик в задаче многомерного индексирования графической информации.....	133
<i>Орлова Ю. А.</i>	
Анализ моделей и методов повышения эффективности проектирования программного обеспечения.....	137
V. ПОСТРОЕНИЕ СЕТЕЙ И ЗАЩИТА ИНФОРМАЦИИ	
<i>Ишкина Е. Г., Щербинина О. В.</i>	
Архитектура адаптивного сервисно-ориентированного промежуточного программного обеспечения.....	142
<i>Кирносенко С. И., Лукьянов В. С.</i>	
Идентификация исправляющих ревизий в системах контроля версий.....	146
<i>Лукьянов В. С., Степанченко О. В., Кузнецова Е. С.</i>	
Имитационное моделирование каналов передачи данных в процессе изучения студентами вопросов по надежности информационных систем.....	149
<i>Македонский С. А., Лукьянов В. С.</i>	
Анализ систем проведения электронного голосования.....	152
<i>Македонский С. А., Лукьянов В. С.</i>	
Универсальный протокол защищенного электронного голосования.....	156

МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ И ЧИСЛЕННЫЕ МЕТОДЫ

УДК 532.527

И. П. Вершинина

МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ДИНАМИКИ НЕНЬЮТОНОВСКОЙ ЖИДКОСТИ ВО ВРАЩАЮЩЕМСЯ ЦИЛИНДРИЧЕСКОМ КАНАЛЕ

Волгоградский государственный технический университет
dtm@vstu.ru

На основе уравнений динамики сплошной среды разработана математическая модель развития течения неньютоновской жидкости, реологические свойства которой описываются степенным законом Оствальда-де Вилла, во вращающемся цилиндрическом канале. Рассматриваются условия различного поведения жидкости вблизи твердой границы: прилипания и пристенного скольжения.

Ключевые слова: динамика сплошной среды, математическая модель, неньютоновская жидкость, вращающийся цилиндрический канал, степенной закон Оствальда-де Вилла, граничные условия, профиль скорости, давление жидкости.

I. P. Vershinina

A MATHEMATICAL MODEL OF NON-NEWTONIAN LIQUID'S DYNAMIC IN THE ROTATING CYLINDRICAL CHANNEL

Volgograd State Technical University

On the basis of continuum dynamic's equations, a mathematical model of non-newtonian flow, the rheological properties are described by the Ostwald-de Wale's power law, in a rotating cylindrical channel is developed. Conditions of various behavior of a liquid near to firm border are considered: stickings and wall slippages.

Key words: continuum dynamic, mathematical model, non-newtonian liquid, rotating cylindrical channel, Ostwald-de Wale's power law, boundary conditions, velocity profile, fluid pressure.

Вращающиеся вокруг своей оси цилиндрические каналы служат для подвода перерабатываемых жидкотекучих сред к ряду центробежных аппаратов, таких как центробежные грануляторы, распылители жидкостей, экстракторы, смесители и т. д. [1]. Внедрение центробежных аппаратов в широкую инженерную практику сдерживается отсутствием систематизированных исследований закономерностей течения неньютоновской жидкости в элементах этих аппаратов.

Цель работы состоит в создании математической модели динамики неньютоновской жидкости, реологические свойства которой описываются степенным законом Оствальда-де Вилла, во вращающемся с постоянной угловой скоростью ω цилиндрическом канале радиуса R . Рассматриваются два условия поведения жидкости вблизи твердой границы: прилипания и пристенного скольжения.

Задача решается в цилиндрической системе координат (рис. 1).

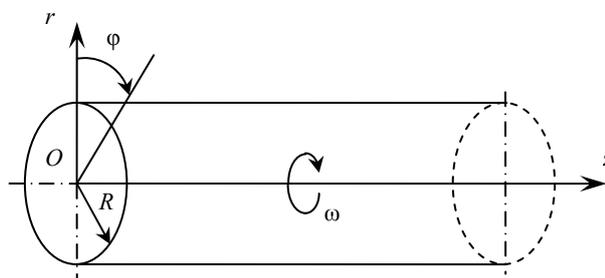


Рис. 1. Цилиндрическая система координат

Используя уравнения динамики сплошной среды в цилиндрической системе координат [2] для стационарного и осесимметричного течения, получаем для неньютоновской жидкости, реологические свойства которой описываются

степенным законом Оствальда-де Вилла, следующие уравнения:

$$V_r \frac{\partial V_r}{\partial r} + V_z \frac{\partial V_r}{\partial z} - \frac{V_\varphi^2}{r} = -\frac{1}{\rho} \frac{\partial P}{\partial r} +$$

$$+ \frac{k}{\rho} \left[A^{n-1} \left(\frac{\partial^2 V_r}{\partial r^2} + \frac{\partial^2 V_r}{\partial z^2} + \frac{\partial V_r}{r \partial r} - \frac{V_r}{r^2} \right) + \right. \\ \left. + 2 \frac{\partial V_r}{\partial r} \frac{\partial}{\partial r} (A^{n-1}) + \left(\frac{\partial V_z}{\partial r} + \frac{\partial V_r}{\partial z} \right) \frac{\partial}{\partial z} (A^{n-1}) \right], \quad (1)$$

$$V_r \frac{\partial V_\varphi}{\partial r} + V_z \frac{\partial V_\varphi}{\partial z} + \frac{V_r V_\varphi}{r} =$$

$$= \frac{k}{\rho} \left[A^{n-1} \left(\frac{\partial^2 V_\varphi}{\partial r^2} + \frac{\partial^2 V_\varphi}{\partial z^2} + \frac{\partial V_\varphi}{r \partial r} - \frac{V_\varphi}{r^2} \right) + \right. \\ \left. + \left(\frac{\partial V_\varphi}{\partial r} - \frac{V_\varphi}{r} \right) \frac{\partial}{\partial r} (A^{n-1}) + \frac{\partial V_\varphi}{\partial z} \frac{\partial}{\partial z} (A^{n-1}) \right], \quad (2)$$

$$V_r \frac{\partial V_z}{\partial r} + V_z \frac{\partial V_z}{\partial z} = -\frac{1}{\rho} \frac{\partial P}{\partial z} +$$

$$+ \frac{k}{\rho} \left[A^{n-1} \left(\frac{\partial^2 V_z}{\partial r^2} + \frac{\partial^2 V_z}{\partial z^2} + \frac{\partial V_z}{r \partial r} \right) + \right. \\ \left. + 2 \frac{\partial V_z}{\partial z} \frac{\partial}{\partial z} (A^{n-1}) + \left(\frac{\partial V_z}{\partial r} + \frac{\partial V_r}{\partial z} \right) \frac{\partial}{\partial r} (A^{n-1}) \right], \quad (3)$$

Для определения давления получим отдельное уравнение. Для этого уравнение (1) умножаем на r , дифференцируем по r , затем делим на r . Уравнение (3) дифференцируем по z . Сложение данных уравнений приведет к зависимости:

$$\frac{1}{\rho} \left(\frac{\partial P}{r \partial r} + \frac{\partial^2 P}{\partial r^2} + \frac{\partial^2 P}{\partial z^2} \right) =$$

$$= 2 \left(\frac{\partial V_r}{\partial r} \frac{\partial V_z}{\partial z} - \frac{V_r^2}{r^2} - \frac{\partial V_r}{\partial r} \frac{\partial V_r}{\partial z} + \frac{V_\varphi}{r} \frac{\partial V_\varphi}{\partial r} \right) +$$

$$+ \frac{2k}{\rho} \left[\left(\frac{\partial V_r}{r \partial r} + \frac{\partial^2 V_r}{\partial r^2} + \frac{\partial^2 V_r}{\partial z^2} \right) \frac{\partial}{\partial r} (A^{n-1}) + \frac{\partial V_r}{\partial r} \frac{\partial^2}{\partial r^2} (A^{n-1}) + \right. \\ \left. + \left(\frac{\partial V_z}{\partial r} + \frac{\partial V_r}{\partial z} \right) \frac{\partial^2}{\partial z \partial r} (A^{n-1}) + \frac{\partial V_z}{\partial z} \frac{\partial^2}{\partial z^2} (A^{n-1}) \right], \quad (4)$$

Здесь: V_r , V_φ , V_z – радиальная, тангенциальная и осевая компоненты скорости; P – давление жидкости; ρ – плотность среды; A – интенсивность скоростей деформаций, k – характеристика consistency, n – индекс течения.

$$A = \left[2 \left(\frac{\partial V_r}{\partial r} \right)^2 + 2 \left(\frac{V_r}{r} \right)^2 + 2 \left(\frac{\partial V_z}{\partial z} \right)^2 + \right. \\ \left. + \left(\frac{\partial V_\varphi}{\partial r} - \frac{V_\varphi}{r} \right)^2 + \left(\frac{\partial V_\varphi}{\partial z} \right)^2 + \left(\frac{\partial V_z}{\partial r} + \frac{\partial V_r}{\partial z} \right)^2 \right]^{1/2}.$$

Система уравнений (1–4) должна решаться при следующих граничных условиях:

$$r = 0; \quad V_r = 0; \quad V_\varphi = 0; \quad \frac{\partial V_z}{\partial r} = 0; \quad \frac{\partial P}{\partial r} = 0;$$

$$r = R; \quad V_r = 0; \quad V_\varphi = \omega R; \quad a) V_z = 0; \quad b) V_z = u_{ck};$$

$$z = 0; \quad V_r = 0; \quad V_\varphi = 0; \quad V_z = u_0;$$

$$z \rightarrow \infty; \quad V_r = 0; \quad V_\varphi = \omega r; \quad \frac{\partial V_z}{\partial z} = 0;$$

$$z \rightarrow \infty; \quad r = 0; \quad P \rightarrow 0;$$

$$z \rightarrow \infty; \quad r = R; \quad P = \frac{\rho \omega^2 R^2}{2}, \quad (5)$$

где $u_{ck} = \beta \tau_{zr}^\alpha$ – скорость скольжения.

Система уравнений (1–4) с граничными условиями (5) представляет собой математическую модель динамики неньютоновской жидкости во вращающемся вокруг своей оси цилиндрическом канале.

Для исключения бесконечного предела интегрирования решение уравнений (1–4) будем искать в виде:

$$V_r = u_0 u(r^*; z^*); \quad V_z = u_0 w(r^*; z^*);$$

$$V_\varphi = u_0 v(r^*; z^*); \quad P = \rho u_0^2 p(r^*; z^*);$$

$$r^* = \frac{r}{R}; \quad z^* = 1 - \frac{1}{1 + c \left(\frac{z}{R} \right)^m},$$

где: $c = 0,171$; $m = 0,254$ – коэффициенты, которые определялись из условия сходимости численного интегрирования.

С учетом этого уравнения (1–4) с граничными условиями (5) преобразуются к виду:

$$\frac{1}{r^*} \frac{\partial}{\partial r^*} (r^* u^2) + B \frac{\partial}{\partial z^*} (uw) - \frac{v^2}{r^*} + \frac{\partial p}{\partial r^*} =$$

$$= \frac{1}{\text{Re}^*} \times \left[A_0^{n-1} \times \left(\frac{\partial^2 u}{\partial r^{*2}} + \frac{\partial^2 u}{\partial z^{*2}} B^2 + \right. \right. \\ \left. \left. + \frac{\partial u}{\partial z^*} \frac{\partial B}{\partial z^*} B + \right. \right. \\ \left. \left. + \frac{\partial u}{\partial r^*} \frac{\partial B}{\partial z^*} - \frac{u}{r^{*2}} \right) + \right. \\ \left. + 2 \frac{\partial u}{\partial r^*} \frac{\partial}{\partial r^*} (A_0^{n-1}) + \right. \\ \left. + \left(\frac{\partial w}{\partial r^*} + \frac{\partial u}{\partial z^*} B \right) \frac{\partial}{\partial z^*} (A_0^{n-1}) B \right], \quad (6)$$

$$\frac{1}{r^{*2}} \frac{\partial}{\partial r^*} (r^{*2} v u) + B \frac{\partial}{\partial z^*} (w v) = \frac{1}{\text{Re}^*} \times \left[A_0^{n-1} \times \left(\frac{\partial^2 v}{\partial r^{*2}} + \frac{\partial^2 v}{\partial z^{*2}} B^2 + \frac{\partial v}{\partial z^*} \frac{\partial B}{\partial z^*} B + \frac{\partial v}{r^* \partial r^*} - \frac{v}{r^{*2}} \right) + \left(\frac{\partial v}{\partial r^*} - \frac{v}{r^*} \right) \frac{\partial}{\partial r^*} (A_0^{n-1}) + \frac{\partial v}{\partial z^*} B^2 \frac{\partial}{\partial z^*} (A_0^{n-1}) \right], \quad (7)$$

$$\frac{1}{r^*} \frac{\partial}{\partial r^*} (r^* u w) + B \frac{\partial}{\partial z^*} (w^2) + B \frac{\partial p}{\partial z^*} = \frac{1}{\text{Re}^*} \left[A_0^{n-1} \times \left(\frac{\partial^2 w}{\partial r^{*2}} + \frac{\partial^2 w}{\partial z^{*2}} B^2 + \frac{\partial w}{\partial z^*} \frac{\partial B}{\partial z^*} B + \frac{\partial w}{r^* \partial r^*} \right) + 2 \frac{\partial w}{\partial z^*} B^2 \frac{\partial}{\partial z^*} (A_0^{n-1}) + \left(\frac{\partial w}{\partial r^*} + \frac{\partial u}{\partial z^*} B \right) \frac{\partial}{\partial r^*} (A_0^{n-1}) \right], \quad (8)$$

$$\frac{1}{r^*} \frac{\partial}{\partial r^*} \left(r^* \frac{\partial p}{\partial r^*} \right) + \frac{\partial^2 p}{\partial z^{*2}} B^2 + \frac{\partial p}{\partial z^*} \frac{\partial B}{\partial z^*} B = 2 \left(\frac{\partial u}{\partial r^*} \frac{\partial w}{\partial z^*} B - \frac{u^2}{r^{*2}} - \frac{\partial w}{\partial r^*} \frac{\partial u}{\partial z^*} B + \frac{v}{r^*} \frac{\partial v}{\partial r^*} \right) + \left[\left(\frac{\partial u}{r^* \partial r^*} + \frac{\partial^2 u}{\partial r^{*2}} + \frac{\partial^2 u}{\partial z^{*2}} B^2 + \frac{\partial u}{\partial z^*} \frac{\partial B}{\partial z^*} B \right) \frac{\partial}{\partial r^*} (A_0^{n-1}) + \frac{\partial u}{\partial r^*} \frac{\partial^2}{\partial r^{*2}} (A_0^{n-1}) + \frac{2}{\text{Re}^*} \left(\frac{\partial w}{r^* \partial r^*} + \frac{\partial^2 w}{\partial r^{*2}} + \frac{\partial^2 w}{\partial z^{*2}} B^2 + \frac{\partial w}{\partial z^*} \frac{\partial B}{\partial z^*} B \right) B \frac{\partial}{\partial z^*} (A_0^{n-1}) + \left(\frac{\partial w}{\partial r^*} + \frac{\partial u}{\partial z^*} B \right) \frac{\partial^2}{\partial z^* \partial r^*} (A_0^{n-1}) + \frac{\partial w}{\partial z^*} \left(\frac{\partial^2}{\partial z^{*2}} (A_0^{n-1}) B^2 + \frac{\partial}{\partial z^*} (A_0^{n-1}) \frac{\partial B}{\partial z^*} B \right) \right]. \quad (9)$$

$$r^* = 0; u = 0; v = 0; \frac{\partial w}{\partial r^*} = 0; \frac{\partial p}{\partial r^*} = 0; \quad z^* = 1; r^* = 0; p = 0; z^* = 1; r^* = 1; p = \frac{R_\omega^2}{2},$$

$$r^* = 1; u = 0; v = R_\omega; a) w = 0; b) w = \frac{u_{ck}}{u_o} = w_s;$$

$$z^* = 0; u = 0; v = 0; w = 1; \quad (10)$$

$$z^* = 1; u = 0; v = R_\omega r^*; \frac{\partial w}{\partial z^*} = 0;$$

Здесь:

$$A_0 = \left[2 \left(\frac{\partial u}{\partial r^*} \right)^2 + 2 \left(\frac{u}{r^*} \right)^2 + 2 \left(\frac{\partial w}{\partial z^*} B \right)^2 + \left(\frac{\partial v}{\partial r^*} - \frac{v}{r^*} \right)^2 + \left(\frac{\partial v}{\partial z^*} B \right)^2 + \left(\frac{\partial w}{\partial r^*} + \frac{\partial u}{\partial z^*} B \right)^2 \right]^{1/2},$$

$$B = \frac{c^{1/m} m}{(1 - z^*)^{(m+1)/m} + z^{*(m-1)/m}},$$

$$\text{Re}^* = \frac{\rho u_0^{2-n} R^n}{k}, \quad R_\omega = \frac{\omega R}{u_0} \text{ – число закрутки.}$$

Для численного решения задачи применяем метод конечных разностей.

Некоторые результаты численного интегрирования приведены в табл. 1 и 2.

Таблица 1

Профили безразмерных осевых скоростей $(w = V_z / u_0)$ для $n = 0,46$
при условии прилипания жидкости к стенке канала

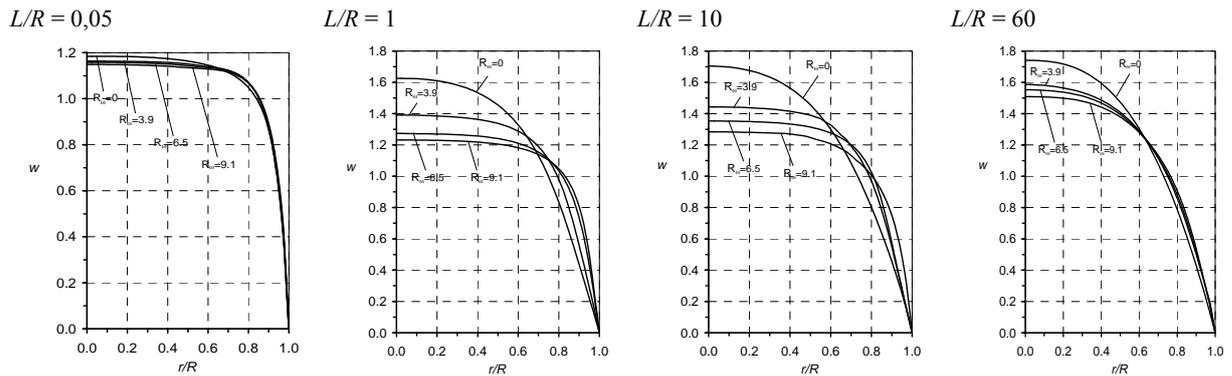
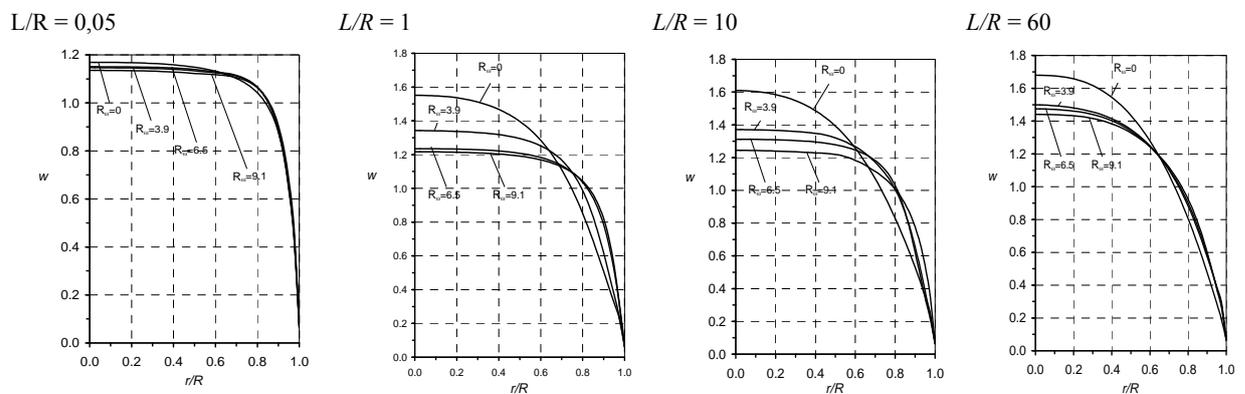


Таблица 2

Профили безразмерных осевых скоростей $(w = V_z / u_0)$ для $n = 0,46$
при условии скольжения жидкости у стенки канала



БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Холин, Б. Г. Центробежные и вибрационные грануляторы и распылители жидкости. – М. Машиностроение, 1977, 132 с.

2. Шульман, З. П. Конвективный теплоперенос реологически сложных жидкостей. – М. Энергия, 1978, 352 с.

УДК 519.612: 624.074.5.06

*В. В. Галишникова***ВЫЧИСЛИТЕЛЬНЫЕ МЕТОДЫ РЕШЕНИЯ ПОЧТИ СИНГУЛЯРНЫХ СИСТЕМ
ЛИНЕЙНЫХ АЛГЕБРАИЧЕСКИХ УРАВНЕНИЙ В АЛГОРИТМАХ ГЕОМЕТРИЧЕСКИ
НЕЛИНЕЙНОГО АНАЛИЗА УСТОЙЧИВОСТИ СТЕРЖНЕВЫХ СИСТЕМ****Российский университет дружбы народов**
galishni@gmail.ru

В работе приведен анализ двух численных методов решения почти сингулярных систем алгебраических уравнений: метода дефляции и метода стабилизации. Дано заключение о возможности их применения в алгоритмах продолжения решения в сингулярных точках пространственных стержневых конструктивных систем.

Ключевые слова: линейные алгебраические уравнения, сингулярные матрицы жесткости, численные методы, алгоритмы и программы.

*V. V. Galishnikova***COMPUTATIONAL METHODS FOR SOLVING NEARLY SINGULAR SYSTEMS
OF LINEAR ALGEBRAIC EQUATIONS IN ALGORITHMS OF GEOMETRICALLY
NONLINEAR STABILITY ANALYSIS OF SPACE TRUSSES****People's Friendship University of Russia**

The two numerical methods for solving nearly singular systems of linear equations, namely deflation and stabilization methods are presented in the paper. A conclusion is made on the possibility of application of these methods in algorithms for continuation of load paths of space trusses beyond singular points.

Key words: linear algebraic equations, singular stiffness matrices, numerical methods, algorithms and programs.

Потеря устойчивости является неотъемлемой проблемой нелинейного поведения конструктивных систем. Необходимым условием неустойчивости мгновенного состояния конструкции является сингулярность матрицы ее касательной жесткости. Однако это условие не является достаточным. Конструкция в точке сингулярности является неустойчивой если при бесконечно малом перемещении в направлении собственного вектора, соответствующего нулевому собственному значению матрицы касательной жесткости, нагрузка уменьшается.

Вычисление точек бифуркации гибких, подверженных большим деформациям конструкций, а также продолжений траекторий их нагружения после прохождения этих точек остается одной из сложнейших задач вычислительной механики. В окрестностях сингулярных точек матриц касательной жесткости произведение собственных значений содержит, по крайней мере, одно значение, почти равное нулю. Определитель матрицы почти равен нулю, и система разрешающих уравнений становится плохо определенной. Вследствие этого, методы решения нелинейных уравнений в окрестностях сингулярных точек становятся ненадежными.

Решения почти сингулярных систем уравнений могут быть получены при помощи специальных методов. В работе Вагнера [1], предлагаются несколько методов решения почти

сингулярных систем разрешающих уравнений, которые, как считает автор, позволяют вычислять траектории продолжения решения в случае, когда известно состояние системы с наименьшим собственным значением.

В данной работе выполнен анализ двух методов: метод дефляции и метод стабилизации. Демонстрируется, что метод дефляции сходится к решению в предельной точке, тогда как метод стабилизации такой сходимости не имеет.

Метод дефляции

Рассмотрим почти сингулярную систему линейных инкрементальных уравнений с симметричной матрицей коэффициентов \mathbf{K} , вектором инкрементов перемещений $\Delta \mathbf{d}$ и вектором инкрементов нагрузок $\Delta \mathbf{p}$:

$$\mathbf{K} \Delta \mathbf{d} = \Delta \mathbf{p}. \quad (1)$$

Обозначим множество ортонормальных собственных состояний матрицы \mathbf{K} как (ω_i, \mathbf{x}_i) , $i = 1, \dots, n-1$, где n – размерность матрицы:

$$\mathbf{K} = \sum_{i=0}^{n-1} \omega_i \mathbf{x}_i \mathbf{x}_i^T, \quad \mathbf{x}_i^T \mathbf{x}_m = \delta_{im}. \quad (2)$$

Здесь ω_i – собственное значение матрицы \mathbf{K} , \mathbf{x}_i – собственный вектор матрицы \mathbf{K} .

Все собственные значения почти сингулярной матрицы \mathbf{K} отличны от нуля. Представим инкременты перемещений и нагрузок в виде линейных комбинаций собственных векторов:

$$\Delta \mathbf{d} = \sum_{k=0}^{n-1} c_k \mathbf{x}_k, \quad (3)$$

$$\Delta \mathbf{p} = \sum_{m=0}^{n-1} \theta_m \mathbf{x}_m. \quad (4)$$

Подставим выражения (2)–(4) в уравнение (1):

$$\sum_{i=0}^{n-1} \omega_i \mathbf{x}_i \mathbf{x}_i^T \sum_{k=0}^{n-1} c_k \mathbf{x}_k = \sum_{m=0}^{n-1} \theta_m \mathbf{x}_m. \quad (5)$$

Так как собственные векторы ортонормальны, выражение (5) сводится к следующему:

$$\sum_{i=0}^{n-1} \omega_i c_i \mathbf{x}_i = \sum_{m=0}^{n-1} \theta_m \mathbf{x}_m. \quad (6)$$

Сравнение коэффициентов при собственном векторе в правой и левой частях уравнения (6) позволяет записать выражение, связывающее коэффициенты данного собственного состояния:

$$\omega_i c_i = \theta_i, \quad i \in \{0, \dots, n-1\}. \quad (7)$$

Пусть известно собственное состояние (ω_0, \mathbf{x}_0) с наименьшим собственным значением $|\omega_0|$. Коэффициенты c_0 и ω_0 для этого собственного состояния могут быть вычислены независимо от того, известны ли другие собственные состояния матрицы \mathbf{K} . Найдем коэффициент θ_0 , умножив уравнение (4) слева на собственный вектор \mathbf{x}_0^T :

$$\theta_0 = \mathbf{x}_0^T \Delta \mathbf{p}. \quad (8)$$

Коэффициент c_0 следует из выражения (7):

$$c_0 = \frac{\theta_0}{\omega_0}. \quad (9)$$

Если собственное состояние (ω_0, \mathbf{x}_0) оказывает преобладающее влияние на решение уравнения (1), необходимо устранить известный вклад этого собственного состояния в решение, вычтя вектор $c_0 \mathbf{x}_0$ из основного уравнения (1). Для этого используем следующие равенства:

$$c_0 \mathbf{K} \mathbf{x}_0 = c_0 \omega_0 \mathbf{x}_0 = \theta_0 \mathbf{x}_0. \quad (10)$$

Вычтем уравнение (10) из уравнения (5):

$$\sum_{i=1}^{n-1} \omega_i \mathbf{x}_i \mathbf{x}_i^T \sum_{k=1}^{n-1} c_k \mathbf{x}_k = \sum_{m=1}^{n-1} \theta_m \mathbf{x}_m. \quad (11)$$

Оставшиеся в уравнении (11) суммы обозначим, как $\Delta \mathbf{z}_0$ и $\Delta \mathbf{p}_0$:

$$\Delta \mathbf{z}_0 = \sum_{k=1}^{n-1} c_k \mathbf{x}_k = \Delta \mathbf{d} - c_0 \mathbf{x}_0, \quad (12)$$

$$\Delta \mathbf{p}_0 = \sum_{m=1}^{n-1} \theta_m \mathbf{x}_m = \Delta \mathbf{p} - \theta_0 \mathbf{x}_0. \quad (13)$$

Подставим выражения (12) и (13) в уравнение (11):

$$\mathbf{K} \Delta \mathbf{z}_0 = \Delta \mathbf{p}_0. \quad (14)$$

Так как матрица \mathbf{K} не является сингулярной, уравнение (14) может быть решено относительно дефлированного (“deflated”) вектора инкрементов перемещений $\Delta \mathbf{z}_0$. Подставив инкремент $\Delta \mathbf{z}_0$ в уравнение (12), получим решение $\Delta \mathbf{d}$ уравнения (1).

Алгоритм решения почти сингулярных уравнений методом дефляции может быть кратко описан следующим образом:

1. Вычисляются собственные значения матрицы коэффициентов системы уравнений, выбирается наименьшее по модулю собственное значение и вычисляется соответствующий ему собственный вектор.

2. Вычисляется коэффициент θ_0 по формуле (8). Вектор инкрементов нагрузок дефлируется в соответствии с уравнением (13).

3. Уравнение (12) решается относительно дефлированного вектора инкрементов перемещений $\Delta \mathbf{z}_0$.

4. Вычисляется коэффициент c_0 по формуле (9). Из уравнения (12) находится полный вектор инкрементов перемещений $\Delta \mathbf{d}$.

Из уравнения (9) можно предположить, что решение уравнения (1) сходится к собственному вектору \mathbf{x}_0 по мере того, как траектория нагружения приближается к сингулярной точке и матрица \mathbf{K} становится сингулярной. Коэффициент c_0 при этом возрастает и становится доминирующим в уравнении (3).

Пример дефляции

Рассмотрим следующую систему инкрементальных разрешающих уравнений, которая является почти сингулярной и плохо обусловленной:

$$\begin{array}{|c|c|c|} \hline 0.147431000 & 14.469023000 & \Delta u_0 \\ \hline 14.469023000 & 1327.394235000 & \Delta u_1 \\ \hline \end{array} = \begin{array}{|c|} \hline -0.078864000 \\ \hline -3.617205000 \\ \hline \end{array}.$$

Вычисленные собственные состояния матрицы коэффициентов:

$$\omega_0 = -0.010284771; \quad \mathbf{x}_0 = \begin{array}{|c|} \hline -0.999940598 \\ \hline 0.010899589 \\ \hline \end{array};$$

$$\omega_1 = 1327.551950770; \mathbf{x}_1 = \begin{bmatrix} 0.010899589 \\ 0.999940598 \end{bmatrix}.$$

Сингулярная матрица в окрестности заданной матрицы будет равна:

$$\mathbf{K}_0 = \begin{bmatrix} 0.157714549 & 14.468910907 \\ 14.468910907 & 1327.394236222 \end{bmatrix}.$$

Вычислим коэффициент θ_0 , после чего выполним дефляцию правой части системы уравнений:

$$\theta_0^{(0)} = \mathbf{x}_0^T \Delta \mathbf{p} = -0.0394332680,$$

$$\Delta \mathbf{p} - \theta_0^{(0)} \mathbf{x}_0 = \begin{bmatrix} -0.039433074 \\ -3.617634806 \end{bmatrix}.$$

Решение системы с поправленной правой частью дает дефлированный вектор инкрементов перемещений:

$$\Delta \mathbf{d}_0^{(1)} = \begin{bmatrix} -0.000029704 \\ -0.002725042 \end{bmatrix}.$$

Вычислим коэффициент

$$c_0 = \frac{-0.0394332680}{-0.010284771} = 3.834141567.$$

Подставив значения c_0 и $\Delta \mathbf{d}_0^{(1)}$ в уравнение (3), вычислим искомый вектор инкрементов перемещений:

$$\begin{aligned} \Delta \mathbf{d} &= \begin{bmatrix} -0.000029704 \\ -0.002725042 \end{bmatrix} = \\ &+ 3.834141567 \begin{bmatrix} -0.999940598 \\ 0.010899589 \end{bmatrix} = \\ &= \begin{bmatrix} 3.833884301 \\ -0.044515611 \end{bmatrix} \end{aligned}$$

Проверка показывает, что при подстановке вектора $\Delta \mathbf{d}$ в исходную систему уравнений, погрешность вычисленного вектора нагрузок относительно заданного имеет порядок $O[10^{-9}]$.

Применение дефляции к решению систем уравнений МКЭ

Рассмотрим почти сингулярную систему линейных уравнений вида $\mathbf{Kd} = \mathbf{q} + \mathbf{r}$, которая включает свободные перемещения \mathbf{d}_1 , заданные перемещения \mathbf{d}_2 , соответствующие им нагрузки \mathbf{q}_1 и \mathbf{q}_2 , и реакции \mathbf{r}_2 . Уравнения в системе сгруппированы следующим образом:

$$\begin{bmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{K}_{21} & \mathbf{K}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{d}_1 \\ \mathbf{d}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{q}_1 \\ \mathbf{q}_2 \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{r}_2 \end{bmatrix}. \quad (15)$$

Пусть матрица-блок \mathbf{K}_{11} почти сингулярна, а пара $(\omega_0, \mathbf{x}_{10})$ представляет собой ее собственное состояние с наименьшим по абсолютной величине собственным значением ω_0 .

$$\mathbf{K}_{11} \mathbf{x}_{10} = \omega_0 \mathbf{x}_{10} \wedge \omega_0 \neq 0 \wedge \mathbf{x}_{10}^T \mathbf{x}_{10} = 1. \quad (16)$$

Инкремент перемещений \mathbf{x}_{10} вызывает инкремент реакций \mathbf{r}_{20} , который равен произведению блока матрицы касательной жесткости \mathbf{K}_{21} и собственного вектора \mathbf{x}_{10} . Собственный вектор \mathbf{x}_{10} занимает верхние строки вектора \mathbf{x}_0 , размерность которого равна размерности матрицы \mathbf{K} . Вектор реакций \mathbf{r}_{20} занимает нижнюю часть вектора \mathbf{r}_0 , размерность которого также равна размерности матрицы \mathbf{K} .

$$\mathbf{K} \mathbf{x}_0 = \omega_0 \mathbf{x}_0 + \mathbf{r}_0, \text{ или}$$

$$\begin{bmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{K}_{21} & \mathbf{K}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{x}_{10} \\ \mathbf{0} \end{bmatrix} = \omega_0 \begin{bmatrix} \mathbf{x}_{10} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{r}_{20} \end{bmatrix}, \quad (17)$$

где $\mathbf{r}_{20} = \mathbf{K}_{21} \mathbf{x}_{10}$.

Здесь \mathbf{x}_0 – собственный вектор системы, \mathbf{r}_0 – собственный вектор реакций системы, \mathbf{r}_{20} – реакции, вызванные перемещениями \mathbf{r}_{10} .

В системе уравнений (15) дефляции подвергается блок \mathbf{q}_1 вектора нагрузок, после чего, следуя процедуре, описанной в предыдущем разделе, вычисляется дефлированный блок вектора перемещений:

$$\begin{aligned} \bar{\mathbf{q}}_1 &= \mathbf{q}_1 - \theta_0 \mathbf{x}_{10}, \quad \text{где } \theta_0 = \mathbf{x}_{10}^T \mathbf{q}_1; \\ \bar{\mathbf{d}}_1 &= \mathbf{d}_1 - c_0 \mathbf{x}_0, \quad \text{где } c_0 = \theta_0 / \omega_0. \end{aligned} \quad (18)$$

Вектор нагрузок системы \mathbf{q} и вектор перемещений системы \mathbf{d} дефлируются путем замены блоков \mathbf{q}_1 и \mathbf{d}_1 на их дефлированные версии:

$$\bar{\mathbf{q}} = \mathbf{q} - \theta_0 \mathbf{x}_0, \quad \bar{\mathbf{d}} = \mathbf{d} - c_0 \mathbf{x}_0. \quad (19)$$

Умножив уравнение (17) на коэффициент c_0 и вычтя его из уравнения (15), получим:

$$\mathbf{K}(\mathbf{d} - c_0 \mathbf{x}_0) = (\mathbf{q} - \theta_0 \mathbf{x}_0) + (\mathbf{r} - c_0 \mathbf{r}_0), \quad (20)$$

$$\text{или } \mathbf{K} \bar{\mathbf{d}} = \bar{\mathbf{q}} + \bar{\mathbf{r}}, \quad (21)$$

где $\bar{\mathbf{r}} = \mathbf{r} - c_0 \mathbf{r}_0$ – дефлированный вектор реакций (22).

В результате решения этого уравнения будут получены дефлированные векторы перемещений и реакций. Полные векторы перемещений и реакций находятся из уравнений (19) и (22):

$$\mathbf{d} = \bar{\mathbf{d}} + c_0 \mathbf{x}_0, \quad \mathbf{r} = \bar{\mathbf{r}} + c_0 \mathbf{r}_0. \quad (23)$$

Метод стабилизации

Рассмотрим почти сингулярную систему линейных инкрементальных уравнений (1).

Пусть при разложении матрицы \mathbf{K} в форме $\mathbf{K} = \mathbf{L}\mathbf{D}\mathbf{L}^T$ коэффициент, стоящий в i -й строке диагональной матрицы \mathbf{D} обращается в ноль. Матрицу \mathbf{K}_a будем называть стабилизированной матрицей, связанной с \mathbf{K} , если их диагональные коэффициенты отличаются на некоторую заданную константу c :

$$\mathbf{K}_a = \mathbf{K} + c\mathbf{e}_i\mathbf{e}_i^T, \quad (24)$$

где \mathbf{e}_i – единичный вектор с коэффициентом 1 в строке i .

Член $c\mathbf{e}_i\mathbf{e}_i^T\Delta\mathbf{d}$ добавляется к обеим частям уравнения (1):

$$\mathbf{K}_a\Delta\mathbf{d} = \Delta\mathbf{p} + c\mathbf{e}_i\mathbf{e}_i^T\Delta\mathbf{d}, \quad (25)$$

Параметр c выбирается таким образом, чтобы матрица \mathbf{K}_a не была сингулярной. Умножим уравнение (25) слева на матрицу, обратную матрице \mathbf{K}_a :

$$\Delta\mathbf{d} = \mathbf{K}_a^{-1}\Delta\mathbf{p} + c(\mathbf{e}_i^T\Delta\mathbf{d})\mathbf{K}_a^{-1}\mathbf{e}_i. \quad (26)$$

Векторы правой части уравнения (26), обозначим $\Delta\mathbf{v}_1$ и $\Delta\mathbf{v}_2$:

$$\mathbf{K}_a\Delta\mathbf{v}_1 = \Delta\mathbf{p}, \quad \Delta\mathbf{v}_1 = \mathbf{K}_a^{-1}\Delta\mathbf{p};$$

$$\mathbf{K}_a\Delta\mathbf{v}_2 = \mathbf{e}_i, \quad \Delta\mathbf{v}_2 = \mathbf{K}_a^{-1}\mathbf{e}_i.$$

Тогда уравнение (26) запишется следующим образом:

$$\Delta\mathbf{d} = \Delta\mathbf{v}_1 + c(\mathbf{e}_i^T\Delta\mathbf{d})\Delta\mathbf{v}_2. \quad (27)$$

Умножим уравнение (27) слева на транспонированный единичный вектор \mathbf{e}_i^T и решим получившееся уравнение относительно скалярного произведения $\mathbf{e}_i^T\Delta\mathbf{d}$:

$$\mathbf{e}_i^T\Delta\mathbf{d} = \frac{\mathbf{e}_i^T\Delta\mathbf{v}_1}{1 - c(\mathbf{e}_i^T\Delta\mathbf{v}_2)}. \quad (28)$$

Подставив выражение (28) в уравнение (26), получим инкремент перемещений в уравнении (25):

$$\Delta\mathbf{d} = \Delta\mathbf{v}_1 + \frac{c\mathbf{e}_i^T\Delta\mathbf{v}_1}{1 - c(\mathbf{e}_i^T\Delta\mathbf{v}_2)}\Delta\mathbf{v}_2. \quad (29)$$

Из уравнения (29) видно, что когда параметр c стремится к нулю, стабилизированная матрица \mathbf{K}_a стремится к почти сингулярной матрице \mathbf{K} , а стабилизированное решение $\Delta\mathbf{d}$ стремится к значению

$$\Delta\mathbf{v}_1 = \mathbf{K}^{-1}\Delta\mathbf{p}. \quad (30)$$

Приведенный вывод уравнений метода стабилизации демонстрирует, что получаемое решение при приближении к сингулярной точке не сходится к величине, кратной собственному вектору, соответствующему наименьшему собственному значению матрицы касательной жесткости.

Выводы

В результате выполненного анализа методов дефляции и стабилизации были сделаны следующие выводы:

1. Метод дефляции обладает высокой точностью решения систем плохо обусловленных систем линейных алгебраических уравнений и может быть применен к решению задач нелинейного деформирования и устойчивости конструктивных систем.

2. Метод стабилизации не дает сходимости решения к величине, кратной собственному вектору, соответствующему наименьшему собственному значению матрицы касательной жесткости в сингулярной точке. Поэтому метод стабилизации не может быть использован для продолжения траектории нагружения в сингулярных точках.

Заключение

В соответствии с результатами исследования, в алгоритме продолжения решения, разработанном автором данной статьи для анализа нелинейного деформирования и устойчивости пространственных стержневых систем, использован метод дефляции. Алгоритм реализован в составе программного приложения SpaceTruss, предназначенного для расчета пространственных стержневых конструкций на деформации и устойчивость в геометрически нелинейной постановке. Выполненные тестовые примеры, описанные в работах автора [2–4] продемонстрировали надежность и устойчивую работу алгоритма, а также высокую точность получаемых результатов.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Wagner, W.: "Zur Behandlung von Stabilitätsproblemen der Elastostatik mit der Methode der Finiten Elemente", Forschungs – und Seminarberichte aus dem Bereich der Mechanik der Universität Hannover, F91/1 (1991).
2. Галишникова В. В. Аналитическое решение нелинейной задачи устойчивости и исследование закритического поведения трехстержневой фермы // Вестник Волгогр. гос. архит.-строит. ун-та. сер.: Естеств. науки. 2006. вып. 6(23). – С. 53–64.
3. Galishnikova V. V. Stability Analysis of Space Trusses // International Journal for Computational Civil and Structural Engineering. 2009. Volume 5, Issue 1&2. Pp. 35–44.
4. Galishnikova V. V. Solving the Unsolvable: Unusual Formulations in Computational Mechanics // Proceedings of EG-ICE Conference "Computing in Engineering". Berlin, 2009. Pp. 113–123.

УДК 004.492

*М. В. Голубятникова***МОДЕЛИРОВАНИЕ ПРОЦЕССА ЭКСТРАКЦИИ
НА ОСНОВЕ МОДИФИЦИРОВАННОГО ИНТЕРВАЛЬНОГО МЕТОДА РАСЧЕТА****Астраханский государственный технический университет
m.golubyatnikova@astu.org**

В работе описываются особенности применения интервального метода при моделировании технологических процессов с учетом количественной и качественной информации на основе применения диаграммы взаимного влияния факторов и аппарата нечетких множеств. Полученная математическая модель процесса является основой для получения достаточной информации при принятии решений на более высоких уровнях.

Ключевые слова: процесс экстракции, растительное сырье, диаграмма взаимного влияния факторов, нечеткие множества.

*M. V. Golubyatnikova***MODELLING OF EXTRACTION PROCESS
ON THE BASIS OF THE MODIFIED INTERVAL METHOD OF CALCULATION****Astrakhan State Technical University**

In work it is described features of application of an interval method at modeling technological processes taking into account the quantitative and qualitative information on the basis of application of diagrams of mutual influence of factors and the device of indistinct sets. The received mathematical model of process is a basis for reception of the sufficient information at decision-making at higher levels.

Key words: extraction process, vegetative raw materials, the diagram of mutual influence of factors, indistinct sets.

Процессы экстрагирования получили широкое распространение для извлечения целевых компонентов из минерального и растительного сырья. Теория диффузионного экстрагирования минерального сырья, свойства которого остаются постоянными во время обработки, была разработана Г. А. Аксельрудом, В. М. Лысянским [1] и другими. Для растительного сырья характерно изменение механических, теплофизических и диффузионных свойств в процессе экстрагирования, поэтому наиболее целесообразно использовать для таких процессов интервальные методы расчета [1].

В работе [2] показано, что традиционный интервальный метод не учитывают целый ряд факторов, влияющих на скорость процесса экстракции. Вследствие этого расчетные экстракционные кривые отличаются от получаемых экспериментально. Это приводит к ошибкам в расчете, неполному извлечению полезного вещества из сырья и неоправданным затратам энергетических ресурсов.

Таким образом, разработка метода моделирования процесса экстракции, позволяющего учитывать весь объем имеющейся информации о процессе, является актуальной научно-практической задачей.

При этом основной проблемой является учет факторов, которые не могут быть представлены в формально-количественной форме. Вместе

с тем, оператор может определять влияние этих факторов и представлять информацию в естественно-лингвистической форме. К ним относятся: качество исходного, промежуточного и конечного продукта (внешний вид, особенности внутреннего строения, цвет, запах, вкус), а также влияние этих факторов на ход процесса. Построение математической модели процесса экстракции с учетом такой информации традиционными методами с использованием только аналитических или регрессионных моделей невозможно.

В качестве основы для построения математической модели процесса экстракции используется диаграмма взаимного влияния факторов [2]. Для формализации информации в естественно-лингвистической форме используются математический аппарат нечетких множеств.

В качестве примера покажем формализацию информации о влиянии анатомических особенностей растительного сырья γ на величину пористости сырья PR. Входной параметр «анатомические особенности растительного сырья» γ представлен в виде лингвистической переменной, а выходной «пористость» PR – в виде численного значения. Для лингвистических переменных введем предопределенные первичные термы: γ {невыраженные, умеренно выраженные, средне выраженные, крупно выраженные, очень выраженные}; PR {малая, ниже средней, средняя, выше средней, большая}.

Следуя принятому математическому аппарату, каждый из первичных термов характеризуется нечетким множеством X . В качестве порождающей функции используем нормальное распределение с соответствующим сдвигом среднего значения, отображаемое на универсальное множество U и приведенное к диапазону $[0...1]$:

$$\mu/U = \frac{K_m}{\sqrt{2\pi\sigma}} \int_{-\infty}^x e^{-\frac{(t-a)^2}{2\sigma^2}} dt, \quad (1)$$

где K_m – масштабный коэффициент, σ – среднеквадратичное отклонение, a – среднее значение.

Тогда первичные термы для каждой лингвистической переменной выглядят следующим образом:

$$\gamma: \{\gamma_1, \gamma_2, \gamma_3, \gamma_4, \gamma_5\},$$

$$\text{где: } \gamma_1 = \frac{2,5}{\sqrt{2\pi}} \int_{-\infty}^x e^{-\frac{t^2}{50}} dt, \quad \gamma_2 = \frac{2,5}{\sqrt{2\pi}} \int_{-\infty}^x e^{-\frac{(t-2,5)^2}{50}} dt,$$

$$\gamma_3 = \frac{2,5}{\sqrt{2\pi}} \int_{-\infty}^x e^{-\frac{(t-5)^2}{50}} dt, \quad \gamma_4 = \frac{2,5}{\sqrt{2\pi}} \int_{-\infty}^x e^{-\frac{(t-7,5)^2}{50}} dt,$$

$$\gamma_5 = \frac{2,5}{\sqrt{2\pi}} \int_{-\infty}^x e^{-\frac{(t-10)^2}{50}} dt,$$

или при отображении на дискретное множество U :

$$\gamma_1 = 1/0 + 0,605/1 + 0,135/2 + 0,011/3 + 0/4 + 0/5 + 0/6 + 0/7 + 0/8 + 0/9 + 0/10$$

$$\gamma_2 = 0,044/0 + 0,324/1 + 0,880/2 + 0,880/3 + 0,324/4 + 0,044/5 + 0,002/6 + 0/7 + 0/8 + 0/9 + 0/10$$

$$\gamma_3 = 0/0 + 0/1 + 0,011/2 + 0,135/3 + 0,605/4 + 1/5 + 0,605/6 + 0,135/7 + 0,011/8 + 0/9 + 0/10$$

$$\gamma_4 = 0/0 + 0/1 + 0/2 + 0/3 + 0,002/4 + 0,044/5 + 0,324/6 + 0,880/7 + 0,880/8 + 0,324/9 + 0,044/10$$

$$\gamma_5 = 0/0 + 0/1 + 0/2 + 0/3 + 0/4 + 0/5 + 0/6 + 0,011/7 + 0,135/8 + 0,605/9 + 1/10.$$

Аналогично для параметра «пористость»

$$PR: \{pr_1, pr_2, pr_3, pr_4, pr_5\}, \text{ где:}$$

$$pr_1 = 1/0 + 0,605/1 + 0,135/2 + 0,011/3 + 0/4 + 0/5 + 0/6 + 0/7 + 0/8 + 0/9 + 0/10$$

$$pr_2 = 0,044/0 + 0,324/1 + 0,880/2 + 0,880/3 + 0,324/4 + 0,044/5 + 0,002/6 + 0/7 + 0/8 + 0/9 + 0/10$$

$$pr_3 = 0/0 + 0/1 + 0,011/2 + 0,135/3 + 0,605/4 + 1/5 + 0,605/6 + 0,135/7 + 0,011/8 + 0/9 + 0/10$$

$$pr_4 = 0/0 + 0/1 + 0/2 + 0/3 + 0,002/4 + 0,044/5 + 0,324/6 + 0,880/7 + 0,880/8 + 0,324/9 + 0,044/10$$

$$pr_5 = 0/0 + 0/1 + 0/2 + 0/3 + 0/4 + 0/5 + 0/6 + 0,011/7 + 0,135/8 + 0,605/9 + 1/10.$$

Малое количество предопределенных термов может быть скомпенсировано использованием набора порождающих термов [3], что позволяет обрабатывать лингвистические данные в широком диапазоне с достаточной точностью.

В результате предварительного анализа выявлено, что чем меньше выражены анатомические особенности сырья, тем меньше коэффициент пористости. Эта взаимосвязь может быть выражена с помощью нечеткого отношения. Построение нечеткого отношения осуществляется на основе набора лингвистических правил:

Если анатомические особенности растительного сырья невыраженные, то пористость малая, иначе если анатомические особенности растительного сырья умеренно выраженные, то пористость ниже средней, иначе если анатомические особенности растительного сырья средне выраженные, то пористость средняя, иначе если анатомические особенности растительного сырья крупно выраженные, то пористость выше средней, иначе если анатомические особенности растительного сырья очень выраженные, то пористость большая.

Формализуя, получим исходное выражение для нечеткого отношения:

Если $\gamma = \gamma_1$, то $PR = pr_1$, иначе если $\gamma = \gamma_2$, то $PR = pr_2$, иначе если $\gamma = \gamma_3$, то $PR = pr_3$, иначе если $\gamma = \gamma_4$, то $PR = pr_4$, иначе $\gamma = \gamma_5$, то $PR = pr_5$.

Далее на основе полученного выражения вычисляется нечеткое отношение FR_1 :

$$FR_1 = (\gamma_1 \cdot pr_1) + (\gamma_2 \cdot pr_2) + (\gamma_3 \cdot pr_3) + (\gamma_4 \cdot pr_4) + (\gamma_5 \cdot pr_5)$$

	1	0,605	0,135	0,044	0,044	0,044	0,002	0	0	0	0
	0,605	0,605	0,324	0,324	0,324	0,044	0,002	0	0	0	0
	0,135	0,324	0,88	0,88	0,324	0,044	0,011	0,011	0,011	0	0
	0,044	0,324	0,88	0,88	0,324	0,135	0,135	0,135	0,011	0	0
	0,044	0,324	0,324	0,324	0,605	0,605	0,605	0,135	0,011	0	0
FR1=	0,044	0,044	0,044	0,135	0,605	1	0,605	0,135	0,011	0	0
	0,002	0,002	0,011	0,135	0,605	0,605	0,605	0,135	0,011	0	0
	0	0	0,011	0,135	0,135	0,135	0,135	0,135	0,011	0,011	0,011
	0	0	0,011	0,011	0,011	0,011	0,011	0,011	0,135	0,135	0,135
	0	0	0	0	0	0	0	0,011	0,135	0,605	0,605
	0	0	0	0	0	0	0	0,011	0,135	0,605	1

Так как параметр «пористость» в дальнейшем используется в расчете коэффициента внутренней диффузии и коэффициента массопередачи как числовой параметр, то необходимо произвести переход от представления в виде нечеткого множества к численному представлению. Для этого используем метод средней площади [3].

Остальные параметры, выражаемые в естественно-лингвистической форме, обрабатываются аналогично. Параметры в численной форме обрабатываются традиционным способом. Сочетание описанных подходов позволяет получить полную математическую модель процесса экстракции с учетом всех влияющих параметров в виде

$$C_{\text{кон}} = f(TS, CL, SM, BE, CR, BD, \tau, IM, p, T, r, \gamma, C_{\text{нас}}, Q), \quad (3)$$

где: TS – вкус, CL – цвет, SM – запах исходного сырья, BE – подвод энергии в процессе экстракции, CR – измельченность сырья, BD – добавка поверхностно-активных веществ, τ – длительность процесса экстракции, IM – интенсивность перемешивания, p – давление в аппарате, T – температура экстрагента, r – радиус диффундирующей частицы, γ – анатомические особенности сырья, $C_{\text{нас}}$ – концентрация насыщения экстрагента целевым компонентом, Q – соотношение масс сырья и экстрагента, $C_{\text{кон}}$ – концентрация готового экстракта.

Следуя интервальному методу, вся длительность процесса τ разбивается на n равных интервалов. По математической модели (3) осуществляется расчет конечной концентрации целевого продукта в экстрагенте в конце i -го интервала $C_{\text{кон}}^i$. Затем производятся вычисления начальных значений входных параметров для интервала $i + 1$, после чего процесс повторяется пока $i \leq n$. $C_{\text{кон}}^i$ на интервале $i = n$ является искомой концентрацией готового экстракта.

Применение нечеткого отношения в качестве основы для интервального расчета наталкивается на ряд трудностей. Наличие верхней и нижней граней матрицы нечеткого отношения приводит к тому, что концентрационные кривые на границах выбранных интервалов не совпадают между собой. Это не позволяет использовать для дальнейшего расчета аффинные преобразования, используемые в интервальном методе. Кроме того, становится невозможным использование различных градиентных методов оптимизации процесса.

Для устранения этого затруднения предлагается провести обработку концентрационных кривых и получение сжатого формального описания, которое может быть использовано в алгоритмах оптимального управления решением. При этом [2] показано, что на каждом интервале участок концентрационной кривой может быть легко аппроксимирован полиномиальным уравнением не выше второй степени.

После обработки концентрационных кривых и их представления в виде совокупности полиномов второго порядка решается задача оптимизации процесса. В качестве критерия оптимальности используется время проведения процесса при получении концентрации целевого компонента в экстрагенте не ниже заданной:

$$\tau \rightarrow \min, \text{ при } C_{\text{кон}} \geq C_{\text{зад}}$$

и наложенных ограничений в виде математической модели вида (3).

Эта задача условной оптимизации, относящаяся к классу задач нелинейного программирования. Методы решения подобных задач в общем виде хорошо известны.

Тем не менее, основной проблемой является монотонный характер убывания совокупности полиномиальных функций. При этом известные способы решения задачи оптимизации «сваливаются» к краю разрешенного диапазона, что приводит к резкому увеличению времени ведения процесса без получения существенного улучшения целевой функции.

В соответствии с этим предлагается решение на основе преобразования задачи с использованием штрафных функций. В качестве штрафной функции используется штраф типа квадрата срезки:

$$\Omega = R \cdot \langle \alpha \rangle^2,$$

$$\text{где } R > 0, \langle \alpha \rangle = \begin{cases} 0, & \tau \leq 0, \\ \tau, & \tau > 0. \end{cases}$$

Этот вид штрафа вводит мягкую искусственную дискриминацию точек допустимой области перед граничными и отражает постоянный рост энергетических затрат при увеличении длительности процесса, что соответствует реальным условиям. После этого возможно применение известных методов условной оптимизации. Получаемое решение является оптимальным с точки зрения длительности процесса.

Применение предложенной модификации интервального метода для расчета процесса экстракции растительного сырья позволяет ус-

тановить математические соотношения между переменными, описывающими состояние процесса экстракции, и представить связи между входами и выходами в аналитической форме. Учет качественной информации о сырье и ходе процесса позволяет рассмотреть все факторы, влияющие на процесс, не отказываясь от использования известных количественных закономерностей. Разработанный алгоритм может быть использован в различных системах автоматического оптимального управления процессами экстракции и массообмена в целом.

УДК 621.83

Ал-др Аф. Гончаров, Ант. Ал-др. Гончаров

МОДЕЛИРОВАНИЕ ПРОЦЕССА САМОТОРМОЖЕНИЯ КЛИНОВЫХ МЕХАНИЗМОВ СВОБОДНОГО ХОДА

Волгоградский государственный технический университет
dtm@vstu.ru

Рассматривается процесс самоторможения клиновых механизмов свободного хода в условиях активного нагружения. Приводятся результаты решения двух контактных задач.

Ключевые слова: механизм свободного хода, контактное взаимодействие, закон Амонтона, самоторможение, неконсервативная модель.

Alexander Af. Goncharov, Ant. Al. Goncharov

MODELLING OF SELF-RETARDATION PROCESS OF FREE-WHEEL WEDGE MECHANISM

Volgograd state technical university

Self-retardation process of free wheel wedge mechanism in the conditions of active loading is considered. Results of the solution of two contact problems are resulted.

Key words: the free wheel mechanism, contact interaction, the law of Amonton, self-retardation, nonconservative model.

Геометрический расчет является наиболее важным этапом проектного расчета всех существующих самозаклинивающихся фрикционных механизмов свободного хода (МСХ). Выбор конструктивных параметров должен обеспечивать работоспособность МСХ и их эксплуатационные характеристики: высокое быстродействие, прочность и жесткость. Под работоспособностью в узком смысле этого слова следует понимать способность механизмов включаться (заклиниваться) при действии расчетной нагрузки. В сложившейся практике проектирования ее оценивают при помощи упрощенных расчетных моделей, и условия работоспособности МСХ имеют вид соотношений, связывающих в состоянии предельного равновесия конструкции геометрические параметры контактных поверхностей элементов и реализуемые на них коэффициенты трения.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Аксельруд, Г. А., Лысянский, В. М. Экстрагирование. Система твердое тело – жидкость. – Л.: Химия, 1974. – 356 с.
2. Голубятникова, М. В., Антонов, О. В. Системный подход в задачах моделирования массообменных процессов. Труды Конгресса по интеллектуальным системам и информационным технологиям «AIS-IT'09». Научное издание в 4-х томах. – М.: Физматлит, 2009, Т. 1, 588 с. – С. 458–463.
3. Проталинский, О. М. Применение методов искусственного интеллекта при автоматизации технологических процессов: Моногр. / Астрахан. гос. техн. ун-т. – Астрахань: Изд-во АГТУ, 2004. – 184 с.

Клиновые МСХ с дополнительной кинематической связью [1] по принципу действия относятся к категории комбинированных механизмов, схемно-конструктивное решение которых обеспечивает теоретическую возможность передачи нагрузки двумя параллельными силовыми потоками – силами трения через контакт клина с обоймами и через кинематическую связь между ними. Экспериментально установлено, что при реализации кинематической связи в виде зубчатого зацепления или муфты Ольдгейма основная часть нагрузки передается фрикционным силовым потоком. В этой связи условия включения таких механизмов, как и чисто фрикционных МСХ, зависят от соотношения геометрических параметров элементов. В частности, в работе [1] удовлетворительно согласующееся с практикой условие включения МСХ было построено при использовании геометрическо-

го критерия самоторможения. Оно имеет следующий вид:

$$f > \frac{e}{r_2 + r_3}, \quad (1)$$

где e – эксцентриситет МСХ, r_2, r_3 – радиусы контактных поверхностей клина (ведущей и ведомой обойм), f – реализуемый в областях контакта коэффициент трения скольжения.

Очевидно, что определяющее конфигурацию элементов МСХ соотношение совершенно не зависит от распределения внутренних сил в областях контакта, его нельзя использовать для нахождения абсолютных размеров механизма, выбор которых при проектном расчете связан с условиями прочности и жесткости конструкции, а также величиной реализуемого механизмом момента трения. Кроме этого, полученное соотношение совершенно не учитывает условия нагружения, что, как показали результаты моделирования [2], не является верным. Поэтому дальнейшее развитие теории клиновых МСХ должно базироваться на более совершенных расчетных моделях, учитывающих как реальное распределение сил на контактных поверхностях элементов, так и историю нагружения. В этом случае нет альтернативы использованию теории контактных задач для анализа силового взаимодействия элементов механизма в процессе его нагружения и применению в качестве теоретических критериев, определяющих кинематическое состояние МСХ, общепринятых оценок фрикционного контакта тел.

При оценке явления самозаклинивания МСХ следует различать два режима эксплуатации: режим стопорного механизма (останова) и механизма пульсирующего действия (выпрямителя механических колебаний). Они соответствуют условиям простого и сложного нагружения. В первом случае происходит остановка ведущего звена механизма, во втором – вращательное движение передается к нагруженному ведомому звену. В силу существенного различия указанных режимов создание самозаклинивающейся конструкции следует начинать с определения параметров, обеспечивающих самоторможение механизма при заданной исходной величине коэффициента трения. Реализация заранее заданного расчетного крутящего момента трения в режиме выпрямителя является акту-

альной задачей более позднего этапа исследований. Поэтому в данной публикации рассмотрены некоторые проблемы геометрического расчета и приводятся результаты гранично-элементного моделирования поведения клинового МСХ в стоповом режиме. Исследования выполнены при помощи разработанной гранично-элементной модели [2].

1. Рассмотрим процесс самозаклинивания клинового механизма в режиме останова. На рис. 1 представлена плоская расчетная схема МСХ. Механизм состоит из следующих конструктивных элементов: вала-эксцентрика 1, ведущей и ведомой обойм 3 и 5, клина 4. Постоянный контакт клина с обоймами обеспечивает прижимное устройство, состоящее из пружины 6 и упора 7. Для снижения потерь на трения в зоне контакта с эксцентриком ведущая обойма имеет подшипник скольжения 2. Кинематическая связь обойм осуществляется посредством зубчатого зацепления 8. Основными геометрическими параметрами МСХ являются: эксцентриситет e , радиусы вала, подшипника и обойм (клина) соответственно r_o, r_n, r, R, R_1 и углы радиальных срезов клина φ_1, φ_2 . Из условий сборки МСХ линейные параметры должны удовлетворять равенству: $R - r - e - \Delta = 0$, где Δ – конструктивный зазор. Практически целесообразным является наложение ограничений на величину зазора ($\Delta \leq 2$ мм) и угловых параметров, ($\varphi_1, \varphi_2 \geq 45^\circ$).

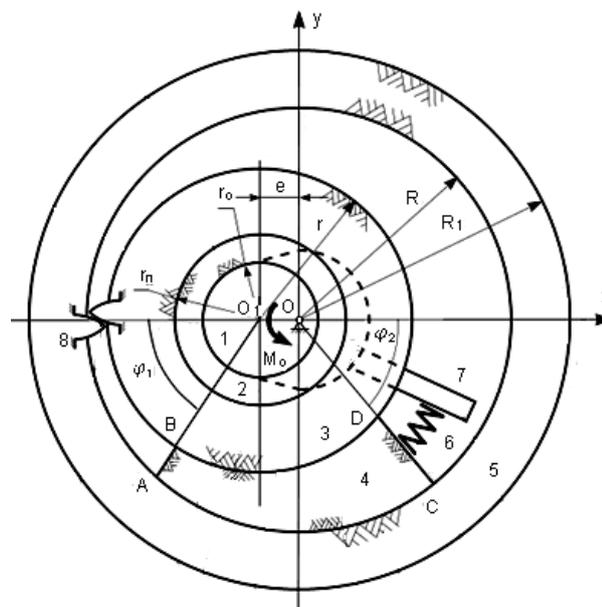


Рис. 1. Клиновой МСХ: конструктивные элементы, геометрические параметры, условия нагружения

В стоповом режиме осуществляется простое нагружение МСХ крутящим моментом M_o , приложенным к ведущему валу-эксцентрику. Момент M_o создает нормальное давление на элементы МСХ и одновременно инициирует их относительное скольжение относительно жестко закрепленной ведомой обоймы. При малых углах поворота эксцентрика данный случай эквивалентен поэтапному возрастанию касательной и нормальной нагрузок, отношение которых сохраняется постоянным. По этой причине процессы самозаклинивания и самоторможения МСХ будут идентичными, то есть реализация включения механизма в условиях простого нагружения не зависит от уровня прилагаемого момента M_o . В этом случае основные характеристики трения определяются только значениями реализуемых в областях контакта коэффициентов трения и геометрическими параметрами элементов МСХ.

Краткое описание модели

Разработанная неконсервативная модель клиновых МСХ [2] воспроизводит условия нагружения, конфигурацию элементов, упругие характеристики конструкционных материалов, наложенные внутренние и внешние связи триботехнической системы. Она учитывает особые условия деформирования, которые при скользящем контакте тел связаны с изменяемостью нагружаемой структуры, а также возможности трансформирования ее механического состояния (статуса) при действии функциональной нагрузки.

Модель использует допущения, позволяющие свести трехмерную контактную задачу к двумерной. Считалось, что механическая нагрузка передается только через фрикционный контакт клина с обоймами и по ширине тел распределена равномерно; элементы МСХ находятся в условиях плоского напряженного состояния, они имеют идеально сопрягаемые контактные поверхности; вал-эксцентрик является абсолютно жестким телом, подшипник, обоймы и клин – идеально упругими; в области контакта вала-эксцентрика с жестко закрепленным в ведущей обойме радиальным подшипником отсутствует трение; в областях контакта клина с обоймами силы трения подчиняются закону Амонтона.

Модель построена на прямом методе граничных интегралов [5]. Гранично-элементная аппроксимация контактной задачи приводит

к формируемой для узловых значений нормальных и касательных перемещений (u_n, u_s), и поверхностных усилий (σ_n, σ_s) системе алгебраических уравнений следующего вида:

$$\begin{cases} Y_s^i = \sum_{j=1}^N C_{ss}^{ij} X_s^j + \sum_{j=1}^N C_{sn}^{ij} X_n^j \\ Y_n^i = \sum_{j=1}^N C_{ns}^{ij} X_s^j + \sum_{j=1}^N C_{nn}^{ij} X_n^j \end{cases}, \quad (i = 1, \dots, N), \quad (2)$$

где: N – общее число элементов в гранично-элементной модели, Y_s^i, Y_n^i – линейные комбинации известных параметров, задаваемых как граничные условия, $C_{ss}^{ij}, C_{sn}^{ij}, C_{ns}^{ij}, C_{nn}^{ij}$ – коэффициенты влияния, связанные с неизвестными граничными параметрами X_s^j, X_n^j .

Система уравнений (2) дополняется имитирующими действие внешней нагрузки граничными условиями на «свободной части» контуров элементов МСХ и условиями их контакта в зонах сопряжений вала-эксцентрика с ведущей обоймой, а также обойм с клином.

Моделирование нелинейного поведения конструкции при наличии неидеальных внутренних связей обеспечивал контактный алгоритм [4], построенный на основе критериев, устанавливающих взаимосвязь между основными конструктивными параметрами механизма, условиями нагружения и кинематическим состоянием МСХ.

Критерии механического состояния МСХ

Критерии состояния МСХ формулировались на основе общепринятых оценок фрикционного контакта твердых тел. В рамках закона Амонтона на основе феноменологических соотношений между нормальными и касательными усилиями, действующими в области контакта тел, определены два состояния их фрикционного взаимодействия – жесткое сцепление и относительное проскальзывание соответственно [6]:

$$|\sigma_s| < f |\sigma_n|, \quad (3)$$

$$|\sigma_s| \geq f |\sigma_n|, \quad (4)$$

где σ_n, σ_s – нормальные и касательные поверхностные усилия, f – реализуемый коэффициент трения скольжения.

Соотношения (3) и (4) использовались при построении вычислительных процедур нелинейного контактного алгоритма в качестве универсальных критериев, определяющих кинематические состояния МСХ. Связь между статусом

конструкции и соответствующим распределением контактных напряжений устанавливалась при рассмотрении явлений микро и макроскольжения тел. Микроскольжение рассматривалось как локальное явление, протекающее в состоянии предварительного смещения (трения покоя) на отдельных участках контактных поверхностей клина с обоймами при выполнении условия (4). Макроскольжение связывалось с состоянием взаимодействующих тел, при котором происходит их относительное движение. Оно осуществляется при выполнении условия (4) во всех точках контактных поверхностей.

В заклиненном состоянии МСХ отсутствует макроскольжение ведущих и ведомых элементов. Его характерным признаком является наличие на каждой границе клина зоны жесткого сцепления или хотя бы одной точки, в которых тела деформируются совместно [2]. Такое состояние элементов по обе стороны контакта определяет соотношение (3). Полное отсутствие зон жесткого сцепления или их вырождение в процессе нагружения хотя бы на одной границе клина означает нарушение статического равновесия МСХ, переход системы в другое кинематическое состояние – относительного скольжения.

Контактный алгоритм

Описание итерационного контактного алгоритма приведено в работах [2–4]. Поведение механизма в условиях активного нагружения определялось при решении системы линейных алгебраических уравнений (2) с ограничениями вида (3), (4). Нелинейный анализ выполнялся в процессе инкрементального нагружения при помощи двух итерационных процедур. Первая процедура связывалась с моделированием процесса нагружения механизма, вторая – с определением его кинематического состояния. На первой итерации решение задачи соответствовало условиям жесткого сцепления тел.

В качестве параметра, управляющего процессом деформирования в рамках принятой истории нагружения, использовался малый угол поворота жесткого эксцентрика относительно общего центра механизма. На каждом этапе нагружения величина момента M_0 пропорциональна значению этого угла, определяющего как размеры области контакта, так и интегральное значение момента. Критерием сходимости равновесия механической системы в пределах заданного допуска являлась оценка баланса ве-

личины задаваемого крутящего момента и интегрального значения момента восстанавливающих упругих сил (контактных напряжений) в зоне контакта эксцентрика с ведущей обоймой.

Определение кинематических состояний МСХ связывалось с непосредственной реализацией контактных ограничений (3) и (4), установлением наличия зон жесткого сцепления и относительного проскальзывания на границах клина с обоймами. На каждой итерации статус граничных элементов и значения контактных функций соответствовали условиям равновесия механической системы и совместности деформаций. При наличии на контактных поверхностях клина зон жесткого сцепления итерации равновесия продолжают до окончательного уточнения их размеров. Полное отсутствие зон жесткого сцепления или их вырождение в процессе нагружения хотя бы на одной границе клина определяло нарушение статического равновесия МСХ, переход системы в состояние относительного скольжения и остановку процедуры решения задачи.

2. Возможности разработанной математической модели использовались для изучения влияния конструктивных параметров на процесс самоторможения клиновых МСХ. Ниже приводятся результаты решения двух представляющих практический интерес контактных задач. Расчеты выполнены на расчетной сетке из 640 граничных элементов при значениях модуля Юнга и коэффициента Пуассона: $E_2 = 1,05 \cdot 10^5$ МПа, $E_3 = E_4 = E_5 = 2,1 \cdot 10^5$ МПа; $\nu_2 = 0,3$, $\nu_3 = \nu_4 = \nu_5 = 0,27$. Обозначения материальных констант соответствуют позициям элементов на рис. 1. Для вычисления полей напряжений и перемещений использовалось $2 \cdot 10^5$ внутренних точек.

Задача 1. Определение величины минимального значения коэффициента трения, обеспечивающего самоторможение МСХ с заданными геометрическими параметрами элементов.

В разработанных теориях клиновых МСХ центральным вопросом геометрического расчета является выбор угла, образованного касательными к контактными поверхностям клина. Значение так называемого угла заклинивания посредством статической расчетной модели связывается с задаваемым коэффициентом трения – обобщенной характеристикой трения, определяющей сопротивление относительному перемещению тел. Оптимальный

угол должен одновременно удовлетворять противоречащим друг другу условиям заклинивания (включения) и расклинивания (выключения) механизма.

Контактные поверхности клина, образованные эксцентрично расположенными дугами окружностей, имеют переменный угол заклинивания, его максимальное значение пропорционально эксцентриситету и не зависит от радиуса ведущей обоймы [1]: $\alpha_{\max} = \arcsin(e/R)$. В рамках сформулированной задачи моделировался процесс самоторможения МСХ с различными углами заклинивания α_{\max} при изменении реализуемого коэффициента трения, и устанавливалась зависимость кинематического состояния моделей механизма от его величины. Модифицирование геометрии производилось за счет величины эксцентриситета. Расчеты выполнялись при значениях e , равных 3,0, 4,0, 4,5 и 5,0 мм, при которых величина угла заклинивания α_{\max} соответственно составила 4,9, 6,56, 7,38 и 8,2°. Модели имели фиксированные параметры: $r_0 = 13$ мм, $r_{II} = 16$ мм, $r = 29$ мм, $R = 35$ мм, $R_1 = 50$ мм, $\varphi_1 = \varphi_2 = 60^\circ$. Минимальное значение коэффициента, обеспечивающего самоторможение механизма, соответствовало моменту образования зон жесткого сцепления клина с ведомой обоймой. Результаты компьютерного моделирования, представлены на рис. 2. Они показывают линейный характер распределения значений указанного коэффициента. Графическая зависимость 1 соответствует коэффициентам трения, определенным для каждого варианта модели по соотношению (1), зависимость 2 построена по результатам численного моделирования. Зависимость 3 показывает линейное увеличение абсолютной разности значений коэффициентов трения ($\Delta f = f_2 - f_1$), определенных при использовании гранично-элементной и упрощенной модели МСХ, с ростом α_{\max} . Так изменение эксцентриситета с 3 до 5 мм приводит к возрастанию Δf на 16 %.

В пределах рассмотренного диапазона выполняется соотношение $f_2 > f_1$. Из этого следует, что при геометрических параметрах, удовлетворяющих условию (1), механизмы не являются самотормозящимися. Реализация процесса самоторможения исследуемых вариантов МСХ является возможной при более высоких значениях коэффициента трения (выделенная область).

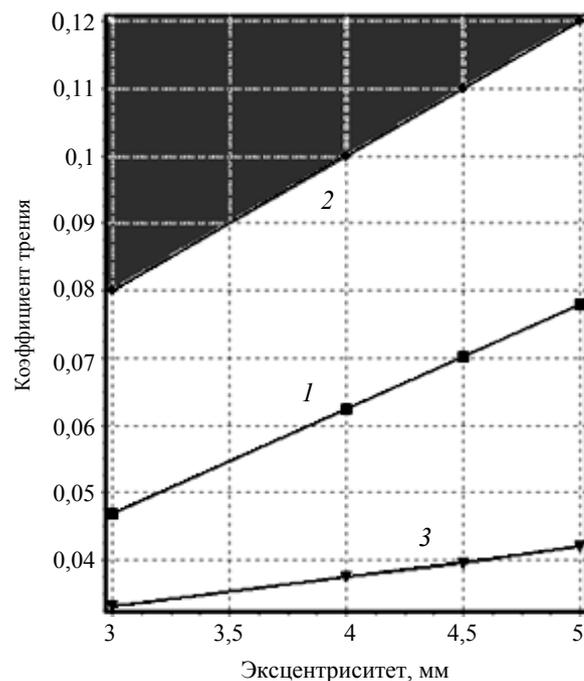


Рис. 2. Определение минимального значения коэффициента трения, обеспечивающего самоторможение МСХ

Решение подобной задачи позволяет выполнять геометрические расчеты МСХ, исходя из заданной величины коэффициента трения, реализуемого, по мнению проектировщиков, в МСХ. Необходимо отметить, что при использовании МСХ в режиме выпрямителя механических колебаний крутящий момент полезной нагрузки, приложенный к ведомой обойме, в области контакта с клином создает тангенциальные усилия. Поэтому выбор геометрических параметров механизма (α_{\max}) должен обеспечивать необходимый запас по сцеплению элементов МСХ. Другой подход связан с использованием для реализации заклинивания механизма усилия поджимающих клин устройств. Моделирование такой ситуации показано при решении следующей контактной задачи.

Задача 2. Определение усилия пружины, обеспечивающего самоторможение механизма.

Включение самозаклинивающегося механизма осуществляется исключительно за счет внутренних сил трения на контактных поверхностях клина с обоймами, создающих сопротивление относительно движению элементов. Необходимо также отметить, что для обеспечения условий постоянного контакта с обоймами, исключения ударов в МСХ используется поджимающее устройство. В самозаклинивающих конструкциях включение реализуется

без учета усилия пружины. Оно может быть незначительным. В МСХ с принудительным заклиниванием величина воздействия на клин со стороны дополнительных устройств является соизмеримой с силами, возникающими при действии функциональной нагрузки. Возможность создания МСХ обоих типов обуславливает причины, в соответствии с которыми усилие пружины может и должно рассматриваться в качестве внутреннего силового фактора.

Первая причина заключается в известной нестабильности реализуемых коэффициентов трения. Перспектива создания высокоэффективных самозаклинивающихся конструкций МСХ во многом связана с применением конструктивных материалов, обладающих стабильно высокими фрикционными характеристиками. При их отсутствии проектный расчет МСХ должен осуществляться с учетом минимального уровня указанных коэффициентов. Создаваемое усилием пружины дополнительное давление на клин может компенсировать нестабильность фактических значений коэффициентов трения, обеспечивая надежность включения МСХ и стабильность величины передаваемого момента. Вторая причина связана с необходимостью определения минимального усилия пружины при проектировании МСХ с принудительным заклиниванием.

В процессе решения задачи исследовалось самоторможение двух различных геометрических моделей МСХ с поджимающими устройствами при заданной величине коэффициента трения $f = 0,1$. Геометрические параметры элементов первого варианта при $\alpha_{\max} = 7,38^\circ$ практически обеспечивали самоторможение конструкции. При решении первой задачи было установлено, что для самозаклинивания данного механизма должен реализоваться коэффициент трения $f_{\min} = 0,11$. Геометрия второй модели соответствовала изначально несамотормозящемуся механизму с углом заклинивания $\alpha_{\max} = 9,87^\circ$. Модели строились при одинаковых значениях параметров: $r_0 = 13$ мм, $r_n = 16$ мм, $R = 35$ мм, $R_1 = 50$ мм, $\varphi_1 = \varphi_2 = 60^\circ$, отличающихся величинах эксцентриситетов ($e_1 = 4,5$ мм, $e_2 = 6$ мм) и радиусов ведущей обоймы ($r_1 = 29$ мм, $r_2 = 28$ мм).

Рассматривалось последовательное увеличение силы упругости пружины от нуля и определялось ее значение, при котором прекращалось макроскольжение элементов МСХ. На рис. 3 показано эволюционное изменение кинематических состояний нагружаемых моделей МСХ

с ростом усилия пружины. С помощью средств графической визуализации вычислительной системы эти состояния представлены картинками суммарных перемещений $u_{\text{sum}}^2 = (u_x^2 + u_y^2)$

с ценой полосы 0,0025 мм. Распределение изоповерхностей перемещений позволяет установить наличие, локализацию и протяженность зон сцепления и относительного проскальзывания элементов, и тем самым определить кинематическое состояние механизма [3]. Полное совпадение порядков полос на границах обойм с клином означает совместную деформацию в образованных зонах жесткого сцепления тел.

При анализе картин было установлено, что увеличение эксцентриситета на 25 % привело к существенному ухудшению условий фрикционного сцепления ведущей обоймы с клином. Наиболее существенные отличия имеют картины начального деформированного состояния моделей ($F = 0$). Изоповерхности перемещений первой модели ($e_1 = 4,5$ мм) имеют деформационный характер (рис. 3, а). Следует отметить совпадение порядков полос на ведущей обойме и клине и их разрывы на границе с ведомой обоймой. Несовпадение порядков полос свидетельствует о том, что деформируемые тела имеют различные перемещения, и происходит относительное скольжение клина с ведущей обоймой относительно ведомой обоймы.

Изоповерхности второй модели ($e_2 = 6$ мм) имеют кинематический характер (рис. 3, б). Их высокий градиент на ведущей обойме и клине показывает, что по сравнению с первой моделью происходит более интенсивное проскальзывание механизма. По этой причине самоторможение первой модели обеспечивает усилие пружины $F = 30$ Н. Из-за большего угла α_{\max} проскальзывание второй модели происходит и при значительном увеличении поджимающего усилия. Принудительное самоторможение происходит только под действием усилия $F = 200$ Н, которое создает относительно центра МСХ момент $M_o(F) = 6,6$ Нм. Его величина приблизительно составляет 31 % от нагружающего механизма крутящего момента $M_o = 21,2$ Нм. В связи с этим можно заключить, что увеличение эксцентриситета свыше величины $e > 0,143R$ создает значительные проблемы для реализации самоторможения механизма при заданной величине коэффициента трения. Данный вывод, безусловно, соответствует только принятым условиям моделирования.

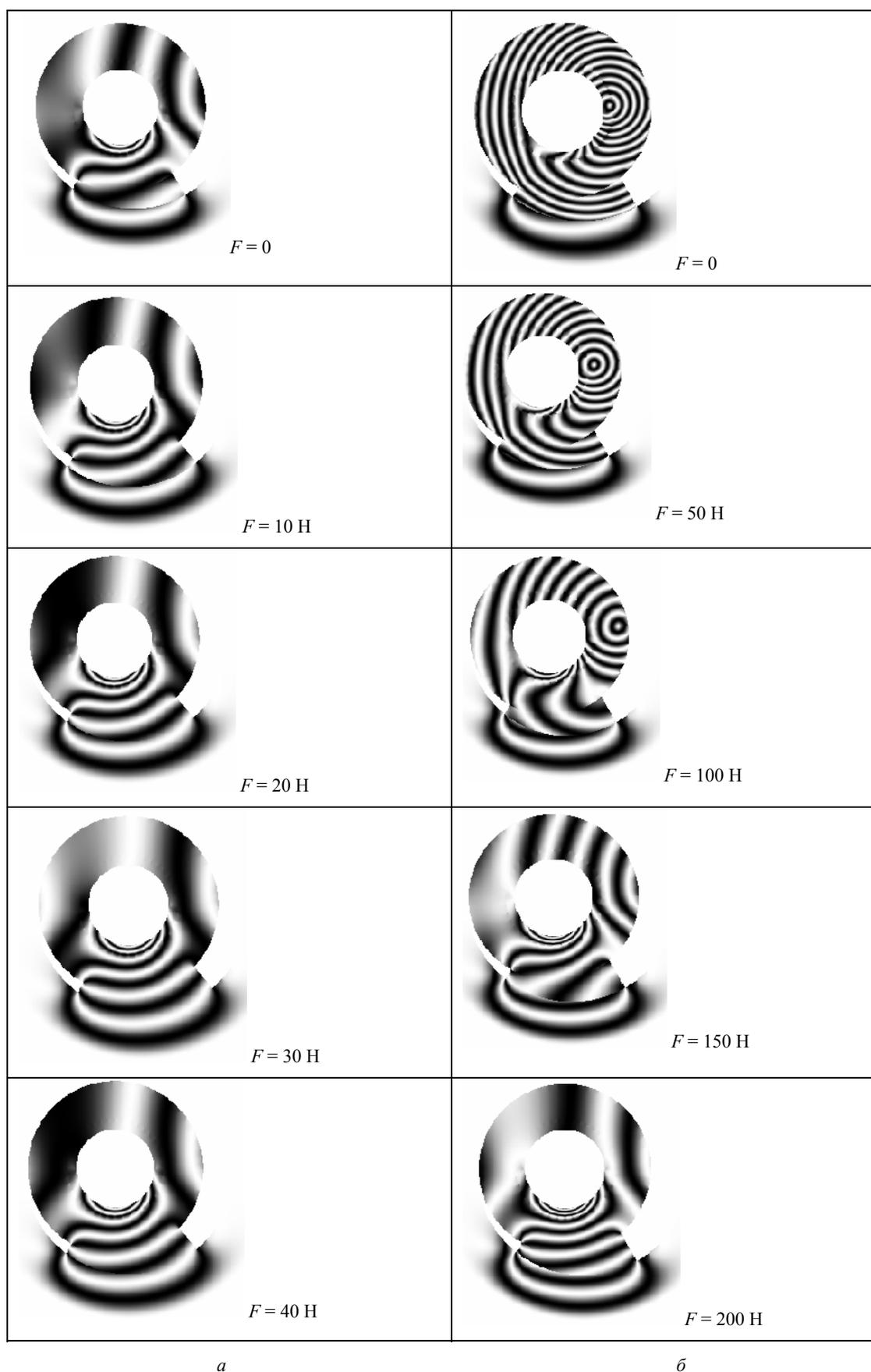


Рис. 3. Трансформация кинематического состояния МСХ при увеличении усилия пружины

Представленные в публикации результаты показывают, что разработанная гранично-элементная модель по сравнению с известными упрощенными моделями на более высоком научном уровне описывает процессы силового взаимодействия элементов клиновых МСХ в условиях скользящего контакта. Она может эффективно использоваться при моделировании наиболее важных процессов самозаклинивания и нагружения клиновых МСХ, для определения их наиболее важных триботехнических характеристик, а также анализа прочности и жесткости конструкции.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Благонравов А. А.* Механические бесступенчатые передачи нефрикционного типа. – М.: Машиностроение. 1977. – 145 с.

2. *Гончаров А. А.* Неконсервативная гранично-элементная модель клиновых механизмов свободного хода // Проблемы машиностроения и надежности машин. РАН. 2008. № 2. – С. 28–36.

3. *Гончаров А. А.* Численное моделирование кинематических состояний клинового механизма свободного хода в фазе активного нагружения // Вестник машиностроения. – 2010. – № 2. – С. 10–16.

4. *Гончаров А. А.* Реализация контактного алгоритма нелинейного конструкционного анализа клинового механизма свободного хода // Известия ВолгГТУ. – 2009. – № 12. – С. 14–21.

5. *Крауч С., Старфилд А.* Методы граничных элементов в механике твердого тела. – М.: Мир. 1987. – 328 с.

6. *Джонсон К.* Механика контактного взаимодействия. – М.: Мир. 1986. – 510 с.

УДК 075.8

В. А. Иванюк, И. Е. Егорова

РАЗРАБОТКА ПРОГРАММНОЙ РЕАЛИЗАЦИИ ГЕНЕТИЧЕСКОГО АЛГОРИТМА ДЛЯ РЕШЕНИЯ ЗАДАЧ МНОГОМЕРНОЙ ОПТИМИЗАЦИИ

Волгоградский государственный технический университет
Hedin99@mail.ru

Раскрывается сущность генетического алгоритма, рассматривается терминология, описываются достоинства и недостатки генетического алгоритма. Приводится пример использования генетического алгоритма на примере решения задачи многомерной оптимизации.

Ключевые слова: генетический алгоритм, оптимизация, система.

V. A. Ivanyuk, I. E. Egorova

DEVELOPMENT OF PROGRAM IMPLEMENTATION OF GENETIC ALGORITHM FOR SOLVING MULTIVARIATE OPTIMIZATION

Volgograd State Technical University

The essence of the genetic algorithm, we consider the terminology, describes the advantages and disadvantages of genetic algorithm. Is an example of using a genetic algorithm on the example of solving the problem of multivariate optimization.

Key words: genetic algorithm, optimization, system.

Генетический алгоритм чаще всего используют в физике при большом количестве экспериментов, он позволяет ответить на вопрос существует ли скрытая зависимость между показателями или нет. Генетический алгоритм дает нам наиболее точную функциональную зависимость между показателями при ограниченном количестве по времени. Например, физики открыли квантовую запутанную пару испускаемую рубином благодаря генетическому алгоритму. Квантовая запутанная пара позволяет повысить скорость передачи данных вдвое. Открыли частицу «бозон», проведя миллионы экспериментов на коллайдере, сделав миллиард фотографий и обработав их с помощью генетического алгоритма. Генетический алгоритм по-

зволяет за один проход (цикл) просчитать 1000 уравнений. Таким образом в классическом понимании генетический алгоритм отвечает на вопрос есть скрытая зависимость между показателями в эксперименте или ее нет.

Сущность генетического алгоритма заключается в имитации процесса эволюции особей одного вида животных в замкнутом пространстве, в результате выживают наиболее приспособленные. Достоинства генетического алгоритма:

1. Высокая скорость нахождения решения в сравнении с другими стохастическими методами.

2. Достаточно низкая вероятность нахождения неверного решения (западание в локальный оптимум).

3. Четкая ограниченность точности по времени в зависимости от необходимой точности вычислений.

Далее будем рассматривать генетический алгоритм как типовой метод многофакторной оптимизации функции с несколькими переменными. Создадим алгоритм решения задач методом генетического алгоритма. На первом этапе введем основную терминологию, используемую при реализации генетического алгоритма.

Рассмотрим простейшую многофакторную функцию вида $F(x_1, x_2, \dots, x_n) = f_1(x_1) * f_2(x_2) \dots f_n(x_n)$, где:

а) ген – соответствует одному из аргументов функции – один аргумент (x_1) или (x_2) или (x_3) и т. д.

б) особь – это полный набор аргументов с помощью которой выражается функция ($x_1, x_2 \dots x_n$), причем значения аргументов выбираются случайно.

в) популяция – это набор особей

г) критерий оптимизации функции (Π) – это величина обратная расстоянию до искомого оптимума.

$$\Pi = \frac{1}{|F_{\text{ИСКОМАЯ}} - F_{\text{особи}}|} \sim \frac{1}{\Delta F},$$

где ΔF расстояние до искомого оптимума.

В случае оптимизации функции неизвестному значению критерия F , $F_{\text{искомое}}$ будет являться наилучшим случайно найденным min или max функции.

д) отсев или вымирание – это удаление из популяции с наименьшим критерием выживания (Π).

е) мутация – это случайные изменения случайно выбранного гена у случайно выбранной особи.

ж) кроссовер – операция, при которой из генов двух случайно выбранных родителей создается особь – потомок, помещающийся на место, освободившейся после смерти особи.

Ограничение времени работы генетического алгоритма может быть реализована тремя различными способами: 1 способ – временное ограничение; 2 способ – количественное ограничение циклов эволюции; 3 способ – способ естественной убыли – смертность превышает рождаемость.

Приведем блок схему работы генетического алгоритма (рис. 1).

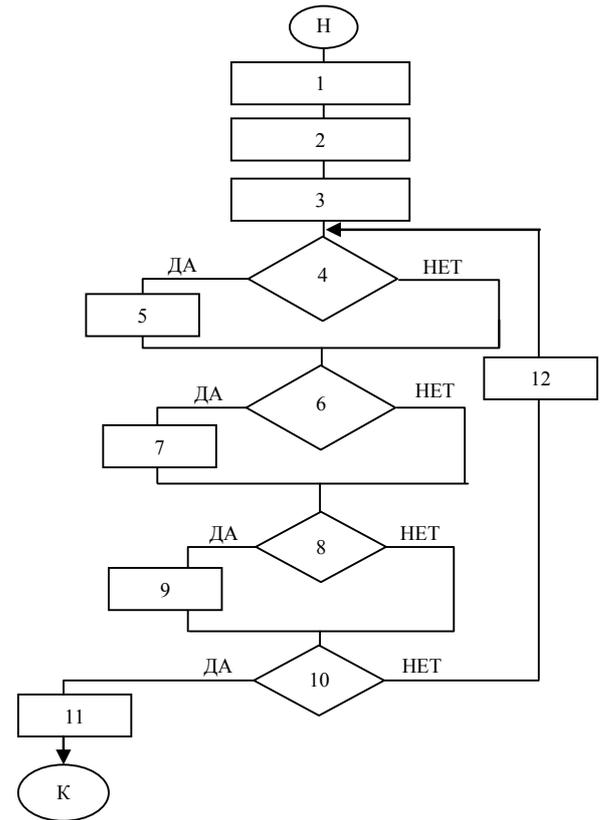


Рис. 1. Блок-схема работы генетического алгоритма

Пункт 1. Вводим в программу основные критерии работы алгоритма: N – начальное число особей в популяции; S – количество эволюционных циклов, через которое происходит отсев; C – количество эволюционных циклов, через которое происходит кроссовер; M – количество эволюционных циклов, через которое происходит мутация. Необходимо, чтобы $S < C$, значения S, C, M варьируем. *Пункт 2.* Создаем в памяти массив из $N \cdot \Gamma$ переменных, где N – количество особей, Γ – количество генов. Заполняем их случайными вещественными числами $A \in R$. *Пункт 3.* Создаем массив из $N_{\text{величин}}$ $\Pi_{\text{приспособленности}}$. Используя критерий оптимизации и значения аргументов особей, рассчитываем для каждой особи ее приспособленность. *Пункт 4.* Проверяем не наступило ли время отсева. Не кратен ли счетчик циклов значению отсева. *Пункт 5.* Если да, находим особь с min значением Π (критерий оптимизации функции). Помечаем занимаемое ею место как свободное. *Пункт 6.* Проверяем не кратно ли значение C счетчику цикла. *Пункт 7.* Если да (кратно), случайным образом выбираем двух родителей на свободное место в таблице особей, помещаем их значения генов (аргументов).

Пункт 8. Проверяем не наступило ли время мутации. Не кратен ли счетчик циклов числу M (мутации). Пункт 9. Получаем три случайных числа особь $a = 1 - N$, ген $b = 1 - \Gamma$, ceR . Особи с номером a , ген с номером b изменяем на c . Пересчитываем ее Π (приспособленность). Пункт 10. Проверяем не осталась ли в живых одна особь. Пункт 11. Выводим как результат ее гены (аргументы). Пункт 12. Увеличиваем счетчик циклов.

Основным недостатком генетического алгоритма является то, что при программирова-

нии задачи невозможно создать универсальный код, описывающий функцию и критерий оптимизации, поскольку начальные условия задач всегда различны.

Рассмотрим пример использования генетического алгоритма. Задача заключается в отыскании параметров регрессии a , Y_{pot} и b из уравнения

$$P = a \ln(Y_{pot} - Y) + b, \quad (1)$$

где Y – динамический ряд реального ВВП, P – динамический ряд дефлятора (общий уровень цен) (таблица).

Таблица

Исходные данные

	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010
ВВП реальный	372,30	373,7892	373,0416	373,7877	376,0304	379,0387	381,3129	387,4139	392,0629	394,4153	394,0208	399,5371
дефляторы	100,8 %	100,4 %	99,8 %	100,2 %	100,6 %	100,8 %	100,6 %	101,6 %	101,2 %	100,6 %	99,9 %	101,4 %

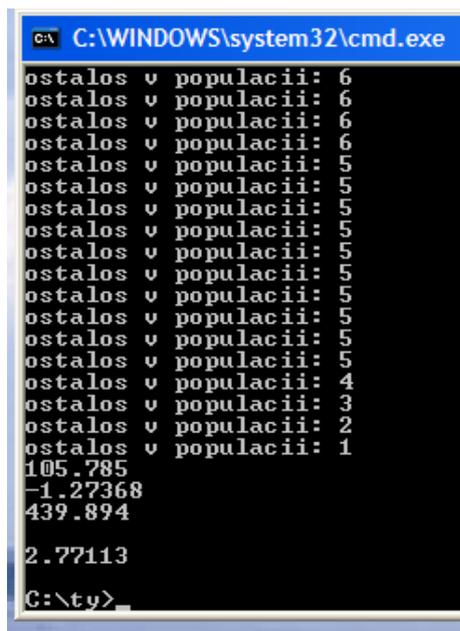


Рис. 2. Программная реализация генетического алгоритма

По условию задачи задано, что: a должно быть отрицательно, Y_{pot} (потенциальный ВВП) должен быть больше всех значений Y , b любым числом.

Данная задача реализована программно в среде разработки dev c++ на языке программирования C++ (рис. 2).

Таким образом имеем следующие результаты: коэффициент a равен -1.27368 , коэффициент b равен 105.785 , коэффициент Y_{pot} равен 439.894 . При этом сумма квадратов отклонений в каждой точке равна 2.77113 .

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Тейво Кохонен, Гвидо Дебок. Анализ финансовых данных с помощью самоорганизующихся карт. Москва, издательский дом. Альпина., 2001.
2. Уоссерман. Нейрокомпьютерная техника., Москва, издательство Мир., 1992.
3. Шумский С. А. Нейрокомпьютинг и его применение в экономике и бизнесе., Москва, издательство МИФИ, 1998.

УДК 681.5:682.3 (075.8)

Е. Г. Крушель, И. В. Степанченко, О. В. Степанченко

ОБ ОРГАНИЗАЦИИ ВЫЧИСЛЕНИЙ В ЦИФРОВЫХ УПРАВЛЯЮЩИХ СИСТЕМАХ С ДВОЙНОЙ ШКАЛОЙ ВРЕМЕНИ

Камышинский технологический институт (филиал)
Волгоградского государственного технического университета

helen@kti.ru, stilvi@mail.ru, ovste@mail.ru

Рассматриваются особенности задач оценки состояния и оптимизации (в терминах аналитического конструирования оптимальных регуляторов, АКОР), возникающих при создании алгоритмического обеспечения цифровых систем управления объектами с непрерывным временем, характерной особенностью процессов в которых является наличие двух субпроцессов с существенными различиями в инерционности.

Ключевые слова: дискретная система управления; разнотемповые системы управления; двухшкальный субоптимальный регулятор.

E. G. Krushel, I. V. Stepanchenko, O. V. Stepanchenko

ABOUT ONE METHOD USING PARALLEL COMPUTING FOR TWO TIME DISCRETE CONTROL SYSTEMS

Kamyshin Technological Institute (branch of) Volgograd State Technical University

The issue deals with the applying of two-time scale methods in the problems of two classes: (a) the simulation of two-time scale linear discrete systems and (b) the control quality decreasing due to the separate control of slow and fast components of process space variables. The simulation of two-time scale control systems was carried out on the optimal linear-quadratic control law.

Key words: Discrete-time control system; two time scales control systems; double scale optimal linear-quadratic control law.

1. Основные положения метода «двойной шкалы времени»

Главная идея метода «двойной шкалы времени» (ДШВ) состоит в декомпозиции общего вектора состояния динамического объекта управления на медленную и быструю составляющую, благодаря чему задача синтеза алгоритмов управления может быть решена для этих составляющих раздельно, после чего вектор управляющих воздействий определяется сложением расчетных значений управлений быстрой и медленной составляющими [1, 2]. Общая схема метода ДШВ такова:

в шкале медленного времени быстрая составляющая трактуется как безынерционная (в частности, для ее описания используются не дифференциальные и не разностные, а алгебраические соотношения);

для периодов резких изменений режима работы объекта, на которых динамика быстрой составляющей существенна, вводится более дробная временная единица (шкала быстрого времени); уравнения, описывающие быструю составляющую, трактуются как динамические, а значения медленной составляющей во время переходных процессов быстрой составляющей приближенно считаются постоянными.

Управляющие системы с ДШВ, при проектировании технологического процесса АСУТП, реализуются на двух уровнях [3].

Верхний уровень управляющей системы ответствен за управление медленной динамикой (стратегией управления). Ее составляющими является динамика инерционного субпроцесса и вынужденная составляющая динамики быстрого субпроцесса, причиной которой является взаимосвязь между медленным и быстрым субпроцессами. Частота управления определяется свойствами динамики медленной составляющей процесса.

Нижний уровень ответствен за подавление собственной динамики быстрого субпроцесса (тактикой управления), вызываемой изменениями начальных условий в моменты резких изменений задающих воздействий.

Шкалы времени функционирования верхнего и нижнего уровней различны, причем величина этих различий прямым образом сказывается на эффективности введения подобной управляющей системы.

К настоящему времени основные результаты по управлению разнотемповыми процессами на базе ДШВ получены для систем управления с непрерывным временем [2]. Для целей использования метода ДШВ для разработки цифровых систем управления разнотемповыми процессами более пригодны модели с дискретным временем. Далее будут приводиться результаты, полученные в этом направлении.

2. Исследование возможности реализации алгоритма конструирования оптимальных регуляторов (АКОР) в двухуровневой управляющей системе с разными шкалами времени

По результату решения АКОР–задачи [4] на модели объекта в быстром масштабе времени без декомпозиции динамики процесса на суб-процессы мы получим эталонное качество управления. Однако расчет этого управления производится в быстром времени максимально возможное количество раз и потому загруженность вычислительной ЭВМ будет наибольшей, что экономически не выгодно.

Поэтому рассмотрим вариант отказа выдачи управляющего воздействия АКОР–алгоритма в каждом такте быстрого времени. Вариации этого приема:

рассчитывать управляющие воздействия, постоянные на интервале дискретизации медленного времени на верхнем уровне управляющей системы и передавать ее на нижний уровень для непосредственного применения;

рассчитывать на верхнем уровне параметры этого алгоритма, постоянные для всей текущей дискреты медленного времени, и затем передавать их на нижний уровень для самостоятельного формирования управляющего воздействия.

Заметим, что расчет оптимального управления АКОР–регулятора $u_{opt}[s]$ в конечный момент дискретного времени s ($s = 0, 1, \dots, N - 1$, N – задано) зависит от значения вектора состояния модели объекта в этот момент времени $x[s]$:

$$u_{opt}[s] = -C[s]x[s] + G[s] \quad (1)$$

где $C[s]$ и $G[s]$ – оптимальные параметры регулятора.

Следовательно, при использовании второй вариации, когда передаются параметры рассчитанного оптимального управления, мы получим более сглаженные управляющие воздействия по сравнению с управлением в вариации 1, что приведет к улучшению критерия оптимальности. Это, несомненно, достоинство вариации 2. Однако в этом случае на локальные регуляторы нижнего уровня управляющей системы ложится функция расчета оптимального управления, что скажется на загрузке вычислительных ресурсов – это недостаток.

Приведенные доводы противоположны доводам в сторону использования первой вариации. Отсюда появляется задача оптимального

выбора вариации, исходя из требований к качеству управления и технической загруженности.

3. Постановка задачи исследования эффективности составного управления в цифровой системе с ДШВ

Поскольку для нахождения составного управления (точнее, его компонент) решается две независимые друг от друга задачи АКОР, то для каждой из них можно задать такие штрафы, при которых результирующее управление в целом получилось бы наилучшим. Таким образом, исследование составного управления начинается с исследования влияния параметров матриц штрафов $K[N]$ и $Q[s]$ на качество управления (например, значение критерия оптимальности).

В данном случае исследуется влияние не всей матрицы $K[N]$ или $Q[s]$, а ее элементов, влияющих на поведение быстрого subprocessa, $-K_{22}[N]$ и $Q_{22}[s]$. Их размерность одинакова – $n_2 \times n_2$.

Предполагается, что эти матрицы симметричные и имеют вид:

$$K_{22}[N] = \begin{bmatrix} \alpha_1 & 0 \\ 0 & \beta_1 \end{bmatrix}, \quad Q_{22}[s] = \begin{bmatrix} \alpha_2 & 0 \\ 0 & \beta_2 \end{bmatrix}.$$

где $\alpha_1, \beta_1, \alpha_2, \beta_2$ – неотрицательные действительные числа.

Следовательно, изменяя значения $\alpha_1, \beta_1, \alpha_2, \beta_2$ можно достичь лучшего качества управления. Примем упрощение, что $\alpha_1 = \alpha_2 = \alpha, \beta_1 = \beta_2 = \beta$, то есть

$$K_{22}[N] = \begin{bmatrix} \alpha & 0 \\ 0 & \beta \end{bmatrix}, \quad Q_{22}[s] = \begin{bmatrix} \alpha & 0 \\ 0 & \beta \end{bmatrix} \quad (2)$$

где α, β – неотрицательные действительные числа.

Таким образом, задача исследования эффективности составного управления в управляющей системе с ДШВ сводится к выбору таких значений α и β , что функция зависимости показателя качества от этих двух переменных достигала своего минимума

$$I(\alpha, \beta) \rightarrow \min. \quad (3)$$

Если число β представить зависимостью от числа α , например, $\beta = Z \times \alpha$, где Z – действительное положительное число, и фиксировать при этом Z в заданном значении, то зависимость (3) примет вид:

$$I(\alpha) \rightarrow \min,$$

то есть придем к функции зависимости от единственной переменной.

Имеет место вариант, когда эффективность составного управления определяется из сравнения с показателем качества управления, например, в случае передачи управляющих воздействий, рассчитанных с использованием ДШВ на медленном верхнем уровне управляющей системы:

$$\text{Эффективность} = \frac{I_{fs_M} - I_{sost}}{I_{sost}} 100\%, \quad (4)$$

где I_{fs_M} – значение показателя качества управления при варианте передаче управляющих воздействий, рассчитанных на основе теории ДШВ, с верхнего уровня управляющей системы на нижний, I_{sost} – качество составного управления.

4. Результаты реализации составного управления в управляющей системе

Как было сказано ранее, составное управление складывается из управления медленным и быстрым subprocessами:

$$u[s, \tau] = u_M[s] + u_B[s, \tau],$$

где τ – такты быстрого времени.

На рис. 1 изображены результаты применения только медленного управления с помощью ДШВ и составного. Качество воспроизведения задания при использовании только медленного управления оказалось лучше, чем при использовании составного управления. Это объясняется известным недостатком метода АКОР: критерий – компромисс между отклонениями вектора состояния от задания и вектора управления от номинала. Поэтому вывод об эффективности составного управления можно сделать только на основании сравнения критериев.

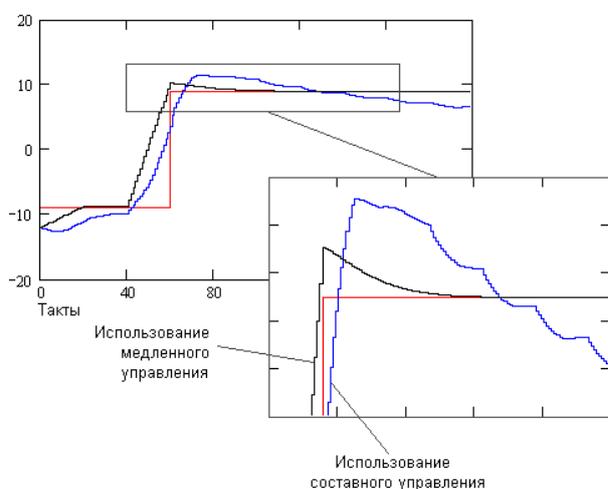


Рис. 1. Обработка задания при использовании медленного и составного управлений

В момент резкого изменения задания быстрая составляющая управления, подавляя собственную динамику быстрого subprocessа, увеличивает быстродействие обработки задания (фрагмент траектории первого компонента вектора состояния на рис. 1).

Для улучшения качества управления была проведена оптимизация настройки параметров контура расчета быстрого компонента составного управления. Степень улучшения качества управления определялась по величине уменьшения критерия оптимальности в системе с составным управлением I_{sost} по сравнению с критерием I_{fs_M} , достижимым при использовании только медленного управления (4). Параметрами контура расчета быстрого компонента составного управления являются параметры матриц штрафов (2).

Варьируя значения α от 0 до 0,15 и Z от 0 до 0,1 с одинаковым для всех шагом 0,01 была получена нелинейная зависимость при $\tau_{\max} = 12$, графически изображенная на рис. 2.

Максимальное значение эффективности составного управления, полученное в результате моделирования, равно 84,53%. Поведение модели объекта при таком значении эффективности составного управления приведено на рис. 1.

Степень участия быстрого компонента в составном управлении имеет экстремум: чрезмерная величина этого компонента приводит к тому, что процесс подавления собственной динамики быстрого subprocessа становится колебательным и поэтому затягивается, ухудшая качество управления вплоть до потери преимуществ составного управления (рис. 3).

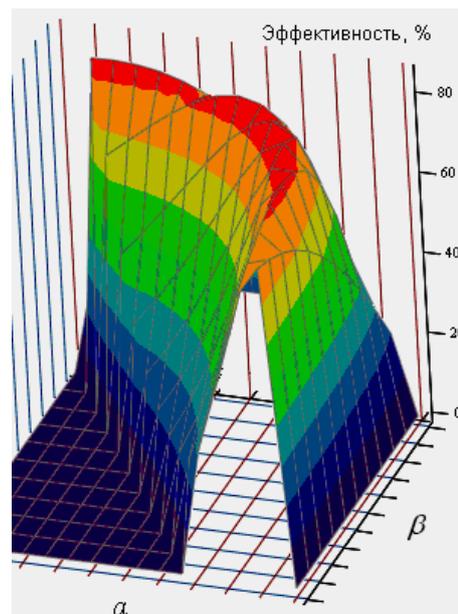


Рис. 2. Зависимость качества управления от параметров матрицы штрафов

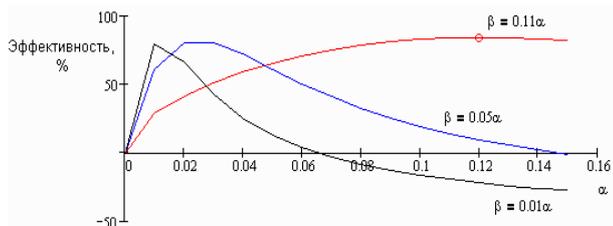


Рис. 3. Зависимость эффективности составного управления от параметра α при фиксировании значения Z

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Крушель, Е. Г. Синтез и моделирование цифровых управляющих систем с двойной шкалой времени: монография / Е. Г. Крушель, О. В. Степанченко; ВолгГТУ. – М.: Машиностроение-1, 2006. – 96 с.

2. Гайцгори, В. Г. Управление системами с быстрыми и медленными движениями / В. Г. Гайцгори. – М.: Наука, 1991. – 224 с.

3. Панфилов, А. Э. Методика формального описания объектов технологического процесса для автоматизированной системы конструирования АСУТП / А. Э. Панфилов, В. А. Камаев // Известия Волгоградского государственного технического университета: межвуз. сб. науч. ст. Сер: Актуальные проблемы управления, вычислительной техники и информатики в технических системах. Вып. 3. – Волгоград: РПК «Политехник», 2007. – № 9(35). – С. 94–97.

4. Степанченко, И. В. Оценка вычислительной трудоемкости вариантов реализации алгоритма АКОР / И. В. Степанченко // Известия Волгоградского государственного технического университета: межвуз. сб. науч. ст. Сер: Актуальные проблемы управления, вычислительной техники и информатики в технических системах. Вып. 1. – Волгоград: РПК «Политехник», 2007. – № 1(27). – С. 100–103.

УДК 004.94

К. В. Кумунжиев, В. Е. Черненко, А. А. Малыханов

**АЛГОРИТМ ПОСТРОЕНИЯ ТРАЕКТОРИИ ДВИЖЕНИЯ АГЕНТА
В МОДЕЛИ ТРАНСПОРТНОЙ СИСТЕМЫ**

Ульяновский государственный университет
kvk@kvk.ulsu.ru, ChernenkoVE@gmail.com, malykhanov@yandex.ru

Предложен алгоритм построения пути следования агента в пространстве с набором препятствий. Результат работы алгоритма – траектория, отстоящая от любого из препятствий не менее, чем на заданное значение.

Ключевые слова: агентное моделирование, построение траектории движения, объезд препятствий, нахождение пути в двумерном пространстве.

K. V. Kumunjiev, V. E. Chernenko, A. A. Malykhanov

**ALGORITHM OF BUILDING AGENT’S TRAJECTORY
IN TRAFFIC MICROSIMULATION MODEL**

Ulyanovsk State University

An algorithm of finding way in two-dimensional space with polygonal obstacles is proposed. The algorithm builds a trajectory with distance not less than given value to any obstacle. Proposed algorithm can be used in agent-based models with special relations between agents and environment.

Key words: agent-based simulation, trajectory building, way finding, barriers in 2D space.

Потребность в создании имитационных моделей транспортных систем возрастает с увеличением транспортных потоков. В основе большинства таких моделей лежат алгоритмы, учитывающие пространственные положения агентов и их передвижение, при этом адекватность модели во многом определяется алгоритмом поведения агентов. Более того, целесообразным является перенесение алгоритмической сложности модели именно на логику функционирования агента. В последнее время модели со сложной логикой поведения агента выделяются в особый класс агентных моделей – модели с «умными» агентами. Использование таких моделей оправдано, в частно-

сти, при моделировании систем движения автомобилей на мелком и среднем уровне. Важнейшей частью алгоритма функционирования агента в таких моделях является построение траектории движения агента.

Пусть имеется двумерное непрерывное пространство, а агент представляет собой прямоугольник заданных размеров, находящийся в исходной точке движения S . В пространстве определены конечная точка движения агента D и препятствия, представляющие собой замкнутые многоугольники произвольной формы без самопересечений. Конечное (N штук) множество препятствий обозначим $P = \{p_i | i = \dots N\}$. Точки S и D находятся вне любого из препятствий p_i .

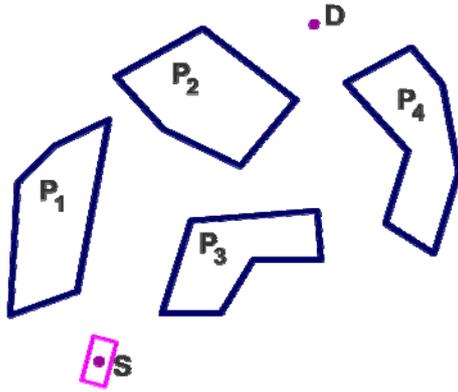


Рис. 1. Иллюстрация к формальной постановке задачи

Необходимо построить ломаную W , начальной точкой которой является точка S , конечной – точка D , удовлетворяющую условию: все точки W должны отстоять от любой из точек препятствий не менее, чем на заданное расстояние d . Это условие объясняется спецификой применения алгоритма – моделирование траектории движения некоторого объекта, при котором необходимо выдерживать безопасную дистанцию до препятствий (например, объезд автомобилем препятствий на дороге). Ломаную W назовем искомой траекторией движения агента.

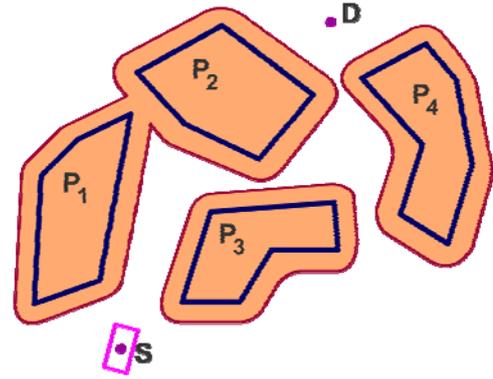
Существует несколько алгоритмов нахождения пути между двумя точками в двумерном пространстве с барьерами [2, 3]. Некоторые из них [2] основаны на отыскании кривых второго и высших порядков, что является неприемлемым при реализации в AnyLogic, так как агенты в AnyLogic могут перемещаться только по прямолинейным траекториям. Другие алгоритмы находят путь в виде ломанной, однако, она может касаться препятствий, что противоречит поставленной задаче. В данной статье предлагается алгоритм построения траектории движения агента в двумерном пространстве с учетом текущего положения агента, целевой точки его движения и имеющихся на пути препятствий. Алгоритм позволяет найти траекторию движения агента, отстоящую на заданном расстоянии от любого из препятствий.

Среди всех точек пространства можно выделить множество точек F , с которыми не может пересекаться искомая траектория W . Построим множество F , основываясь на следующих соображениях:

любая внутренняя точка любого препятствия p_i принадлежит F ;

любая точка, отстоящая от любого препятствия p_i менее, чем на d , принадлежит F .

Схематично множество F показано на рис. 2.

Рис. 2. Схематичное изображение множества F

Множество F имеет сложную структуру: его границы состоят не только из прямых линий, но и дуг окружностей, описанных вокруг точек изгиба ломаных. Учитывать такие формы границ множества F сложно, так как их математическое описание будет содержать уравнения второго порядка, и операции с такими уравнениями внесут в алгоритм дополнительную сложность, что приведет к излишним вычислениям при построении пути объезда препятствий. Следовательно, для исключения из алгоритма нахождения траектории W операций с кривыми второго порядка необходимо произвести аппроксимацию криволинейных участков ломаными. Рис. 3 иллюстрирует необходимые построения для некоторого препятствия p_i .

Возьмем d^* , большее чем d , построим внешнюю границу l_i множества точек, отстоящих от барьера p_i менее, чем на d^* . Рассмотрим кривую l_i вблизи некоторой точки p_i^j изгиба ломаной p_i .

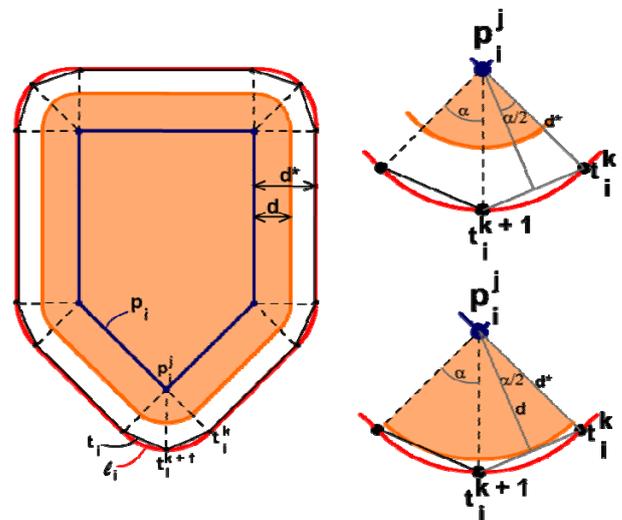


Рис. 3. Дополнительные построения

Для того, чтобы избавиться от дуги окружности, можно приблизить ее некоторой ломаной, причем все ее точки должны находиться на расстоянии не менее чем d от точки p_i^j . Построение ломаной будем производить следующим образом: в качестве первой точки ломаной берется точка пересечения кривой l_i и луча, проведенного из p_i^j перпендикулярно одному из отрезков препятствия p_i ; все следующие точки ломаной лежат на пересечении окружности с лучами, исходящими из p_i^j под углами, кратными некоторому значению α . Таким образом, необходимо выбрать d^* и α так, чтобы:

d^* не намного превышало d (чтобы не допустить неточности при построении искомой траектории);

α не было бы слишком малым (во избежание возникновения большого количества точек изгиба ломаной).

Положим $d^* = \beta d$, тогда $\beta = \frac{1}{\cos \frac{\alpha}{2}}$. Снача

чала выберем угол α , а затем определим соответствующее ему значение β . Как отмечалось, угол α не должен быть слишком малым, однако при больших значениях α увеличивается и β , что приводит к нежелательному увеличению d^* . В результате экспериментов выявлено, что приемлемый результат достигается при значении $\alpha = \frac{\pi}{3}$, тогда $\beta = \frac{2}{\sqrt{3}} \approx 1,155$ и $d^* = \frac{2d}{\sqrt{3}}$.

Определив α и d^* , можно построить ломаную, окружающую препятствие p_i и не приближающуюся к p_i ближе, чем на d . Построенную ломаную t_i , состоящую из точек t_i^k , назовем огибающей для препятствия p_i .

Точки ломаной-препятствия p_i обозначим $\{p_i^1, \dots, p_i^{m_i}\}$. Аналогично точки огибающей t_i обозначим $\{t_i^1, \dots, t_i^{k_i}\}$. Тогда каждой точке t_i^k будет соответствовать некоторая точка p_i^j .

Сформируем множество C из отрезков вида $[p_i^j, t_i^k]$, причем отрезок добавляем в множество, если расстояние от точки t_i^k до любого из препятствий не меньше d^* . Сформируем множество V из конечных точек t_i^k отрезков множества C . Добавим в множество V точки S и D . Сформируем множество отрезков E следующим образом: для всех пар точек $v_i, v_j \in V$ от-

резок $[v_i, v_j]$ включаем в E , если он не пересекается ни с одним из отрезков ломаных-препятствий p_i и не пересекается ни с одним из отрезков множества C .

Представим множества V и E в виде графа $G = \langle V^*, E^* \rangle$, где каждой точке $v \in V$ соответствует вершина графа $v^* \in V^*$, а каждому отрезку $e \in E$ – ребро графа $e^* \in E^*$. Построенный граф изображен на рис. 4.

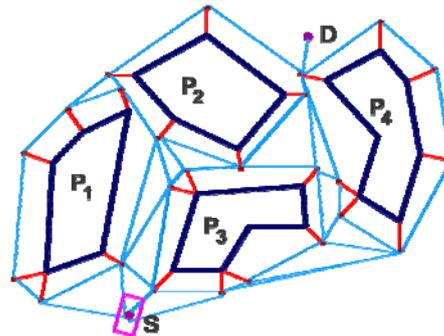


Рис. 4. Граф поиска пути

Причем, $(V \cup E) \cap F = \emptyset$ то есть все точки, соответствующие ребрам и вершинам графа не принадлежат F . Значит, ни одна из точек траектории не приближается к препятствиям ближе, чем на d .

Теперь задача нахождения пути W состоит в поиске кратчайшего пути на графе G , из вершины, соответствующей точке S , в вершину, соответствующую точке D . Найдем такой путь, применив алгоритм Дейкстры [1]. Результатом работы этого алгоритма может стать последовательность $(S^*, v_{q_1}^*, \dots, v_{q_k}^*, D^*)$, либо сообщение о том, что путь не найден. В случае положительного результата работы алгоритма путь W строится как объединение отрезков, соответствующих ребрам, связывающим последовательно вершины $(S^*, v_{q_1}^*, \dots, v_{q_k}^*, D^*)$.

Предложенный в статье алгоритм положен в основу логики поведения агента в моделях транспортных систем.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Dijkstra E. W. A note on two problems in connection with graphs, Numerical Mathematics, Vol. 1, Pp. 269–271, Oct 1959.
2. Kapoor S., Maheshwari S.N. Efficient algorithms for Euclidean shortest path and visibility problems with polygonal obstacles, Proceedings of 4th Annual ACM Symposium on Computational Geometry, 1988, Pp. 172–182.
3. Klippell, K.-F. Richter, S. Hansen. Wayfinding Choreme Maps, Visual Information and Information Systems. 8th International Conference, VISUAL 2005, Springer, Berlin.

УДК 621.865.8

В. В. Чернышев

МОДЕЛИРОВАНИЕ ДИНАМИКИ ШАГАЮЩЕЙ МАШИНЫ С ЦИКЛОВЫМИ ДВИЖИТЕЛЯМИ КАК СИСТЕМЫ ТВЕРДЫХ ТЕЛ С УПРУГОДИССИПАТИВНЫМИ СВЯЗЯМИ*

Волгоградский государственный технический университет
dtm@vstu.ru

Рассматривается динамическая модель шагающей машины в виде системы твердых тел – корпуса и шагающих опор (шагающих модулей) правого и левого борта, имеющих упругодиссипативные связи. Модель позволяет решать задачи повышения энергетической эффективности движения шагающих машин методами динамического управления колебаниями.

Ключевые слова: динамическое моделирование, многоногие шагающие машины, механическая система, упругодиссипативные связи.

V. V. Chernyshev

MODELING THE DYNAMICS OF THE WALKING MACHINE WITH THE CYCLIC PROPULSORS AS SYSTEM SOLIDS WITH ELASTIC AND DAMPING RELATIONS

Volgograd State Technical University

The dynamic model of a walking machine as a system of solid – the machine body and walking modules right and left sides with elastic and damping constraints is considered. Model allows us to solve problems to improve energy efficiency of motion of walking machines by the dynamic control of vibrations.

Key words: dynamic modeling, multilegged walking machines, mechanical system, the elastic-dissipative coupling.

Опыт разработки и испытаний шагающих машин с цикловыми движителями показывает, что они обладают высокой проходимостью и отличаются простотой и надежностью [1]. Вместе с тем, скорость их передвижения существенно ограничена. Это обусловлено высокими энергозатратами на преодоление сил инерции в каждом цикле (шаге) движения.

В работе рассматривается динамическая модель шагающей машины с цикловыми движителями в виде системы твердых тел – корпуса и шагающих опор (шагающих модулей) правого и левого борта, соединенных упругодиссипативными связями (подвеской). Такая модель, в частности, позволяет определять параметры подвески, обеспечивающей помимо традиционных для нее функций, взаимную компенсацию сил инерции корпуса и шагающих опор [2]. Известные математические модели движения шагающих аппаратов, например, в виде шарнирно-соединенной совокупности абсолютно твердых тел – корпуса и звеньев ног [3, 4], не могут в полной мере удовлетворять поставленной задаче.

При описании движения корпуса машины положение его центра масс (рис. 1) определялось относительно неподвижной системы от-

счета $O\xi\eta\zeta$ радиус-вектором ρ_C . Оси $C\xi'\eta'\zeta'$ параллельны неподвижной системе отсчета $O\xi\eta\zeta$, связаны с корпусом в центре масс и перемещаются поступательно. Оси $Sx_1y_1z_1$ жестко связаны с корпусом и совпадают с его главными осями инерции. Их положение относительно координатных осей $C\xi'\eta'\zeta'$ определяется корабельными углами: крена θ , дифферента ψ и рысканья φ .

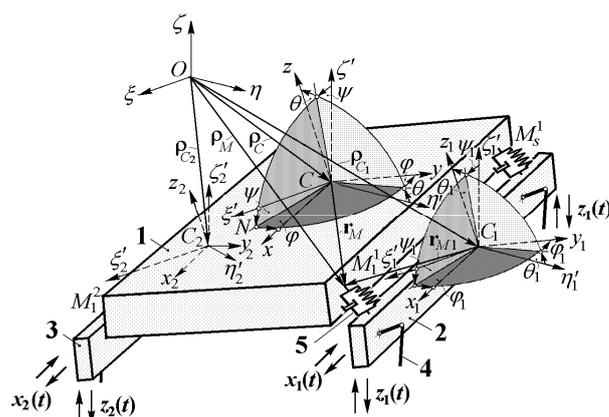


Рис. 1. Расчетная схема шагающей машины:
1 – корпус машины; 2, 3 – шагающие опоры левого и правого борта; 4 – механизмы шагания (ноги); 5 – упругодиссипативные связи (подвеска)

*Работа выполнена при поддержке РФФИ (проект 09-08-9701) и ФЦП «Научные и научно-педагогические кадры инновационной России» на 2009–2013 годы.

Положение центров масс шагающих опор определялось радиус-векторами \mathbf{p}_{C1} и \mathbf{p}_{C2} . Системы координат $C_k x_k y_k z_k$ жестко связаны с опорами в их центрах масс. Здесь и далее нижний индекс k показывает номер шагающей опоры ($k = 1, 2$), а верхний индекс k относится к k -му борту корпуса, к которому крепится опора с номером k . Координатные плоскости $C_k x_k y_k z_k$ совпадают с плоскостями шагания, оси $C_k x_k$ направлены вдоль несущих балок опор. Положение связанных осей $C_k x_k y_k z_k$, относительно неизменно ориентированных осей $C_k \xi_k \eta_k \zeta_k$, определяется углами θ_k, ψ_k, ϕ_k , которые образуются аналогично углам θ, ψ, ϕ .

Корпус соединен с каждой шагающей опорой посредством S связей с приведенными к точкам крепления осевыми коэффициентами жесткости $c_{sx}^k, c_{sy}^k, c_{sz}^k$ и вязкого сопротивления $\mu_{sx}^k, \mu_{sy}^k, \mu_{sz}^k$ ($s = 1, 2, \dots, S$). Частный случай связей – шарнирное крепление шагающих опор к корпусу.

Математическая модель движения корпуса машины в подвижной системе координат $Cxyz$ записывалась в форме Эйлера – Лагранжа

$$\begin{aligned} m_0 (\dot{\mathbf{v}}_C + \boldsymbol{\omega} \times \mathbf{v}_C) &= \mathbf{R}^{(e)}, \\ \Theta^C \cdot \dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \Theta^C \cdot \boldsymbol{\omega} &= \mathbf{L}^{(e)}, \end{aligned} \quad (1)$$

где: m_0 – масса корпуса шагающей машины; $\mathbf{v}_C, \boldsymbol{\omega}$ – вектора скорости центра масс и угловой скорости корпуса; Θ^C – тензор инерции корпуса в его центре масс; $\mathbf{R}^{(e)}, \mathbf{L}^{(e)}$ – главный вектор и главный момент внешних сил, которые, складываются из сил тяжести и реакций элементов подвески.

В общем случае реакции подвески можно подразделить на силы упругих элементов F_s^k , диссипативные силы R_s^k и реакции шарниров N_s^k в точках крепления корпуса к шагающим опорам.

Система (1) дополнялась кинематическими уравнениями, связывающими обобщенные скорости корпуса машины $\dot{\xi}_C, \dot{\eta}_C, \dot{\zeta}_C, \dot{\theta}, \dot{\psi}, \dot{\phi}$ с квазискоростями $v_x, v_y, v_z, \omega_x, \omega_y, \omega_z$ (проекциями векторов \mathbf{v}_C и $\boldsymbol{\omega}$ на подвижные оси $Cxyz$) [5]:

$$\begin{aligned} \dot{\xi}_C &= v_x \alpha_{11} + v_y \alpha_{21} + v_z \alpha_{31}, \quad \dot{\theta} = \omega_x \cos \varphi - \omega_y \sin \varphi, \\ \dot{\eta}_C &= v_x \alpha_{12} + v_y \alpha_{22} + v_z \alpha_{32}, \\ \dot{\phi} &= \omega_x \operatorname{tg} \theta \sin \varphi + \omega_y \operatorname{tg} \theta \cos \varphi + \omega_z, \\ \dot{\zeta}_C &= v_x \alpha_{13} + v_y \alpha_{23} + v_z \alpha_{33}, \\ \dot{\psi} &= \omega_x \sin \varphi / \cos \theta + \omega_y \cos \varphi / \cos \theta. \end{aligned} \quad (2)$$

где $\alpha_{11}, \alpha_{12}, \alpha_{13}, \dots, \alpha_{31}, \alpha_{32}, \alpha_{33}$ – направляющие косинусы.

Движение шагающих опор в плоскости шагания задавалось кинематическими уравнениями. Наибольшая динамическая нагруженность машины имеет место при недеформируемом грунте. В этом случае вектора скорости \mathbf{v}_k и ускорения \mathbf{a}_k k -й опоры в абсолютном движении равны:

$$\mathbf{v}_k = - \sum_{j=1}^J U_{jk} \mathbf{v}_{Cjk}^{(r)}, \quad \mathbf{a}_k = - \sum_{j=1}^J U_{jk} \mathbf{a}_{Cjk}^{(r)}, \quad (3)$$

где $\mathbf{v}_{Cjk}^{(r)}, \mathbf{a}_{Cjk}^{(r)}$ – относительные скорости и ускорения опорных точек j -й ноги движителя ($j = 1, 2, \dots, J$) k -го борта; $U_{jk} = U_{jk}(z_{Cjk})$ – единичная функция, описывающая состояние j -й ноги движителя, принимающая лишь два значения: 1 при опоре ноги на грунт и 0 при переносе (считалось, что в опоре находится тот механизм шагания, опорная точка которого имеет в системе отсчета $C_k x_k y_k z_k$ наименьшую вертикальную координату $z_{Cjk}^{(r)}$).

При определении $\mathbf{v}_{Cjk}^{(r)}$ механизмы шагания рассматривались как плоские I -звенные механизмы. При составлении дифференциальных уравнений движения звеньев их угловые скорости $\dot{\phi}_{ij}$ ($i = 1, 2, \dots, I$) выражались через скорости точек, на которые накладываются внешние связи. Выражения для скоростей узловых точек получались последовательным, от звена к звену, рассмотрением движений твердых тел. В итоге для механизма шагания получали систему уравнений вида:

$$\begin{aligned} \dot{\phi}_{2j} &= f_{2j}(\phi_{1j}, \phi_{2j}, \phi_{3j}, \dots, \phi_{(I-1)j}, \dot{\phi}_{1j}), \\ \dot{\phi}_{3j} &= f_{3j}(\phi_{1j}, \phi_{2j}, \phi_{3j}, \dots, \phi_{(I-1)j}, \dot{\phi}_{1j}, \dot{\phi}_{2j}), \\ &\dots \\ \dot{\phi}_{(I-1)j} &= f_{(I-1)j} \left(\phi_{1j}, \phi_{2j}, \phi_{3j}, \dots, \phi_{(I-1)j}, \dot{\phi}_{1j}, \dot{\phi}_{2j}, \dots, \dot{\phi}_{I-2} \right). \end{aligned} \quad (4)$$

Для задания движения шагающей опоры к уравнениям (3) и (4) добавлялись формулы для проекций относительных скоростей опорных точек

$$\mathbf{v}_{Cjx}^{(r)} = - \sum_{i=1}^q \dot{\phi}_{ij} l_i \sin \varphi_{ij}, \quad \mathbf{v}_{Cjz}^{(r)} = \sum_{i=1}^q \dot{\phi}_{ij} l_i \cos \varphi_{ij}, \quad (5)$$

являющихся узловыми точками опорного q -го звена, где l_i – длина i -го звена механизма шагания, а также формулы для относительных координат опорных точек, необходимых для определения момента смены стоп.

Уравнения (3)–(5) задают движение шагающих опор в плоскости шагания. Положение самой плоскости определяется динамикой всей системы.

Кинематические уравнения связывающие обобщенные скорости шагающих опор с квази-скоростями, аналогичны уравнениям (2). Они будут несколько проще, так как корпус шагающих опор, вследствие синхронного движения движителей одного борта, имеет нулевой дифферент ($\psi_k = 0$).

Движение корпуса машины происходит под действием кинематических возмущений $x_k(t)$, $z_k(t)$, создаваемых движителями и передающихся от шагающих опор через точки крепления подвески. Деформации связей и реакции в точках шарнирного закрепления корпуса заранее неизвестны.

При движении машины для каждой точки M_s^k крепления связей к корпусу, в частности для точки M_1^1 (рис. 1), справедливо векторное равенство:

$$\rho_{M_s}^k = \rho_C + \mathbf{r}_{M_s}^k = \rho_{Ck} + \mathbf{r}_{Msk}, \quad (6)$$

где $\rho_{M_s}^k$, $\mathbf{r}_{M_s}^k$, \mathbf{r}_{Msk} – радиус-вектора точки M_s^k в инерциальной системе отсчета $O\xi\eta\zeta$, в системе отсчета $Cxyz$, жестко связанной с корпусом и в системе отсчета $Cx_ky_kz_k$, связанной с k -й шагающей опорой, соответственно.

Для характеристики деформаций связей использовался вектор

$$\Delta \mathbf{r}_s^k = \mathbf{r}_{Msk} - \mathbf{r}_{Msk}^{cm}, \quad (7)$$

где \mathbf{r}_{Msk}^{cm} – радиус-вектор, определяющий в системе отсчета $Cx_ky_kz_k$ положение точки $M_s^k = M_{sk}^{cm}$ при статическом равновесии системы. Считалось, что \mathbf{r}_{Msk}^{cm} жестко связан с шагающей опорой и при движении остается в системе отсчета $Cx_ky_kz_k$ постоянным как по модулю, так и по направлению.

Диссипативные составляющие реакций подвески определяются относительными скоростями $\Delta \dot{\mathbf{r}}_s^k$ их точек крепления на корпусе машины.

Реакции в шарнирах крепления корпуса к шагающим опорам увеличивают число неизвестных в системе (1)–(7) и она дополнялась уравнениями связывающими скорости центра масс корпуса и скорости точек его шарнирного крепления, движение которых считалось заданным:

$$\mathbf{v}_{M_s}^k = \mathbf{v}_C + \omega \times \mathbf{r}_{M_s}^k, \quad (8)$$

где $\mathbf{v}_{M_s}^k = \mathbf{v}_{Msk}$ – вектор задаваемой скорости точки M_s^k крепления корпуса к k -й шагающей опоре; $\mathbf{r}_{M_s}^k$ – радиус-вектор, определяющий в подвижной системе отсчета $Cxyz$ положение точки крепления шарнирной связи.

На базе представленной динамической модели был проведен динамический анализ шагающего робота «Восьминог» предназначенного для работы на слабых грунтах. С учетом свойств подвески система уравнений математической модели в этом случае существенно упрощается. В работе использовалась полужесткая схема подвески (рис. 2) с независимым креплением шагающих опор. Упругие элементы, в рассматриваемой схеме, обладают жесткостью только в направлении оси Cz , специальных демпфирующих устройств в подвеске не предусмотрено.

В рассматриваемом случае в группе уравнений (1)–(8) независимой переменной является ω_y – проекция вектора ω на ось Cy . Проекции ω_x и ω_z на оси Cx и Cz определяются из векторного уравнения (8). Составляющие скоростей шарниров в плоскостях шагания известны – они равны соответствующим скоростям шагающих опор. Результаты расчета динамики некоторых режимов движения машины приведены на рис. 3.

При движении «Восьминога», вследствие малой курсовой скорости (около 1 км/ч) и незначительных сил инерции, динамичность процесса проявляется слабо. Характер движения корпуса определяется, в основном, кинематикой шагающих опор. Подвеска машины выполняет лишь традиционные для системы поддрессирования функции и не решает задачи взаимной компенсации энергозатрат на преодоление цикловых сил инерции корпуса и шагающих опор при колебаниях системы.

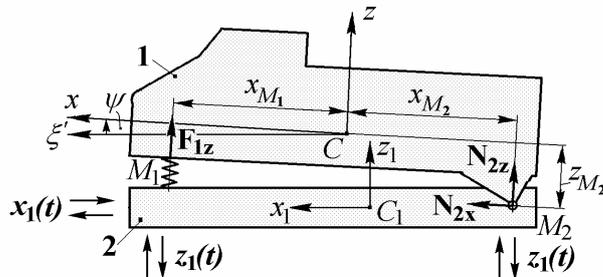


Рис. 2. Схема подвески полужесткого типа шагающего робота «Восьминог»:
1 – корпус; 2 – шагающая опора

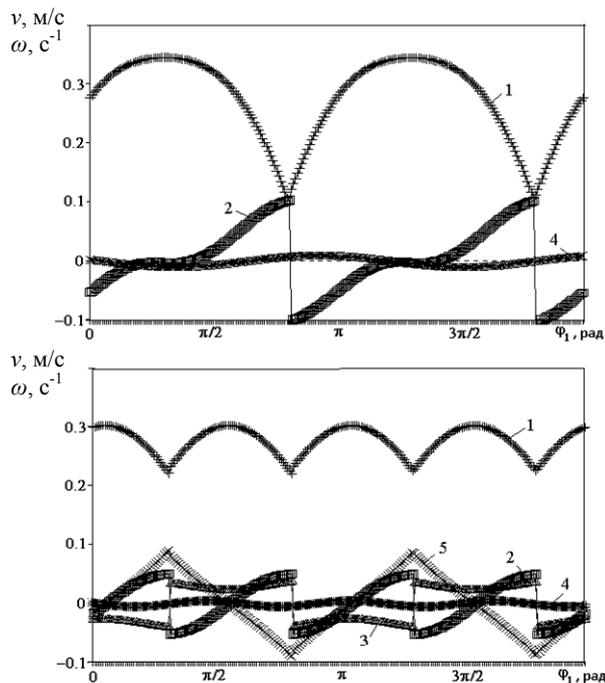


Рис. 3. Цикловые зависимости скоростей $v_x, v_z, \omega_x, \omega_y, \omega_z$ (кривые 1–5, соответственно) шагающего робота «Восьминог» при синхронном (а) и противофазном (б) движении движителей правого и левого борта

При более высоких скоростях движения, как показали результаты динамического моде-

лирования, существует принципиальная возможность частичной взаимной компенсации затрат мощности на преодоление цикловых сил инерции корпуса и шагающих опор без ухудшения показателей виброзащиты. Эффект взаимной компенсации инерционных затрат мощности шагающих машин аналогичной с 5-ти тонным «Восьминогом» весовой категории будет заметен уже при скорости порядка 4–5 км/ч.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Чернышев, В. В. Полевые исследования шагающих машин / Чернышев В.В. // Тракторы и сельскохозяйственные машины. – 2004. № 4. – С. 20–22.
2. Чернышев, В. В. Пассивное подрессоривание в мобильных робототехнических системах с цикловыми механизмами шагания / Чернышев В. В. // Известия вузов. Машиностроение. – 2003. – № 1. – С. 31–39.
3. Охоцимский, Д. Е., Механика и управление движением автоматического шагающего аппарата / Охоцимский Д. Е., Голубев Ю. Ф. – М.: Наука, 1984. – 310 с.
4. Брискин, Е. С. Курсовая устойчивость шагающей машины «Восьминог» / Брискин Е. С., Шурыгин В. А., Жога В. В., Чернышев В. В., Малолетов А. В. // Информационно-измерительные и управляющие системы. – 2006. – № 1–3. – С. 56–58.
5. Брискин, Е. С. Концепция проектирования, динамика и управление движением шагающих машин. Ч. 2. Динамика движения шагающих машин серии «Восьминог» / Е. С. Брискин, В. В. Чернышев, В. В. Жога, А. В. Малолетов, Н. Г. Шаронов, Н. Е. Фролова // Мехатроника, автоматизация, управление. – 2005. – № 6. – С. 19–26.

УДК 629.1.03

Н. Г. Шаронов

СИНТЕЗ АЛГОРИТМА УПРАВЛЕНИЯ ДВИЖЕНИЕМ ШАГАЮЩЕЙ МАШИНЫ «ВОСЬМИНОГ» В ЗАВИСИМОСТИ ОТ ПАРАМЕТРОВ ЛОКАЛЬНОГО ПРЕПЯТСТВИЯ*

Волгоградский государственный технический университет
dtm@vstu.ru

В статье сформулированы алгоритмы управления движением шагающей машины. Описывается автономная видеосистема определения относительного положения локальных препятствий. Представлены результаты исследования алгоритма управления движением машины «Восьминог» в зависимости от параметров локального препятствия.

Ключевые слова: управление движением, шагающая машина.

N. G. Sharonov

SYNTHESIZE CONTROL ALGORITHM OF MOTION OF WALKING MACHINE “VOSMINOG” DEPENDING ON LOCAL IRREGULARITY PARAMETRS

Volgograd State Technical University

The control algorithm of motion of walking machine is formulated in the article. The autonomous videosystem of positioning of local obstacles have been described. The results of research of algorithm of motion of machine “Vosminog” depending on local irregularity parameters.

Key words: control algorithm of motion, walking machine.

Шагающая машина «Восьминог» [1, 2], разработанная в ВолгГТУ, предназначена для перемещения по относительно ровной поверхно-

сти в маршевом режиме движения. Задача синтеза алгоритмов управления состоит в выборе законов управления приводами, и допускает

* Работа выполнена при финансовой поддержке Роснауки и РФФИ (09-08-97016).

различные постановки [3]. Рассматривается задача управления движением шагающей машины «Восьминог» с учетом расположения локальных препятствий, в которой заданы начальное и конечное положение центра масс машины.

Источником визуальной информации при управлении движением шагающей машины «Восьминог» являются две web-камеры, установленные на равном расстоянии от его горизонтальной оси симметрии (рис. 1).

Исходными данными для синтеза алгоритмов управления движением шагающей машины «Восьминог» являются координаты x_K, y_K ориентира относительно корпуса машины (рис. 2). Введение координат точек старта становится целесообразно только когда заданы абсолютные координаты ориентира. При математическом моделировании в качестве исходных данных принимаются координаты ориентира x'_K, y'_K , и координаты центра масс машины x'_C, y'_C в абсолютной системе координат (связанной с поверхностью движения):

$$\begin{cases} x'_K = x'_C + x_K \cos \vartheta - y_K \sin \vartheta \\ y'_K = y'_C + x_K \sin \vartheta + y_K \cos \vartheta \end{cases} \quad (1)$$

Координаты x_K, y_K ориентира в подвижной системе координат, связанной с корпусом машины:

$$\begin{cases} x_K = (x'_K - x'_C) \cos \vartheta + (y'_K - y'_C) \sin \vartheta \\ y_K = -(x'_K - x'_C) \sin \vartheta + (y'_K - y'_C) \cos \vartheta \end{cases} \quad (2)$$

Процесс движения от одного ориентира к другому разделяется на два этапа: движение до зоны точного подхода к ориентиру ($S > S_{кр}$) и точный подход к ориентиру ($S_{кр} > S > S_0$). Под критическим расстоянием до ориентира $S_{кр}$ понимается минимальное расстояние до преодо-

ваемого препятствия, обеспечивающее точный подход к ориентиру для его преодоления, или минимальное расстояние до объекта с особыми свойствами (маяк, ворота и т. п.), обеспечивающее точный подход для постановки машины и ее ног в определенное положение. В проведенных исследованиях ориентир считался достигнутым, если расстояние до объекта становилось меньше заданного расстояния S_0 .

В шагающей машине «Восьминог» закон изменения угловой скорости вращения приводов механизмов шагания имеет вид:

$$\begin{cases} \omega_1 = k_1 \omega_0 \\ \omega_2 = k_2 \omega_0 \end{cases} \quad (3)$$

где: k_1, k_2 – коэффициенты дискретного управления; ω_0 – номинальная скорость вращения двигателя. Управление приводом с выполнением дискретных команд «вперед», «стоп», «назад» соответствует значениям коэффициентов k_1, k_2 соответственно «1», «0», «-1». Тогда, с учетом жесткой кинематической связи движителей по бортам, возможными являются 9 различных режимов движения шагающей машины «Восьминог». Например, для поступательного движения вперед $k_1 = 1, k_2 = 1$, для разворота на месте $k_1 = -1, k_2 = 1$ или $k_1 = 1, k_2 = -1$, и т. д.

Синтез алгоритмов управления движением шагающей машины «Восьминог» основывается на разделении области возможного движения относительно машины на несколько зон или секторов. В зависимости от положения ориентира в той или иной зоне (секторе), определяются состояние машины (т. е. комбинации коэффициентов дискретного управления двигателями приводов движителей).

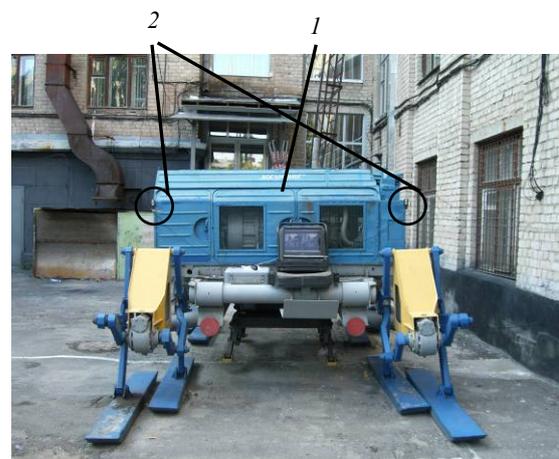


Рис. 1 – Расположение бортовой видеосистемы:
1 – шагающая машина; 2 – Web-камеры

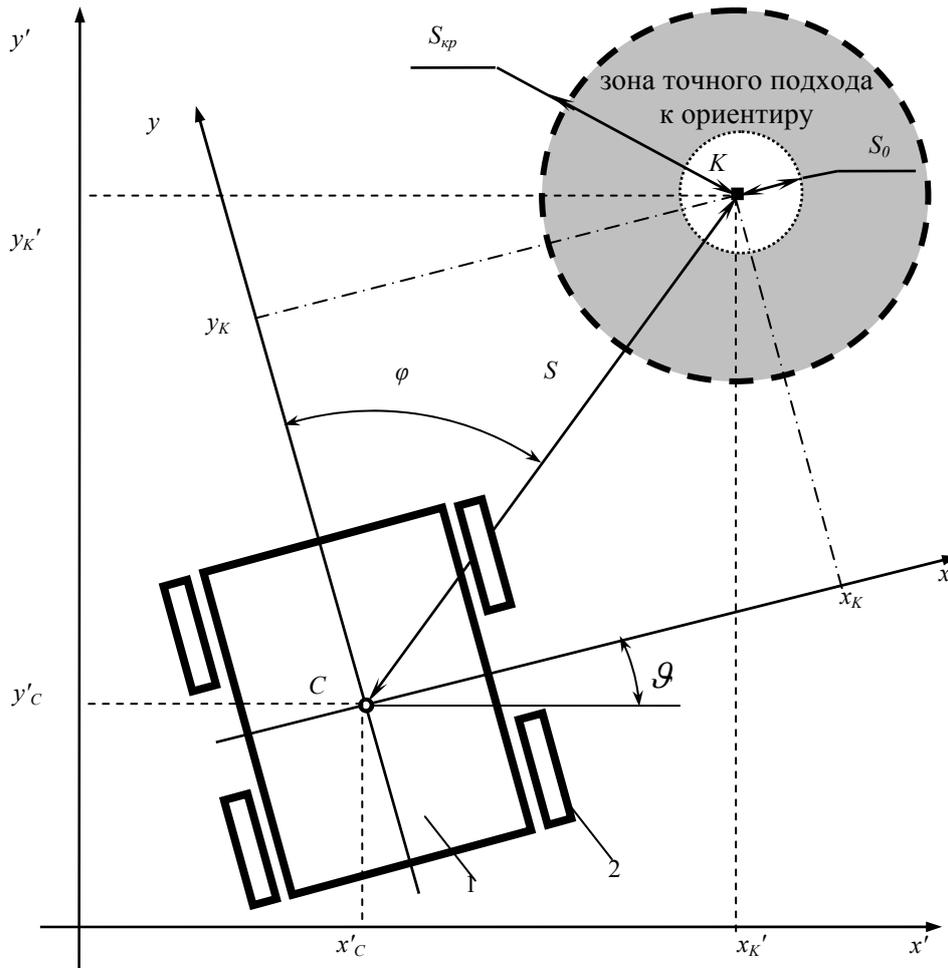


Рис. 2. Схема движения машины:
1 – корпус машины, 2 – движители (спаренные механизмы шагания)

В алгоритме «зона» плоскость движения разбивается относительно машины на пять зон (рис. 3), положение ориентира определяется его координатами относительно машины.

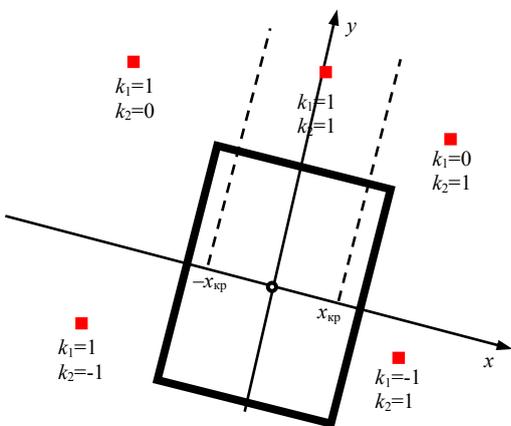


Рис. 3. Значения коэффициентов дискретного управления по алгоритму «зона» в зависимости от положения ориентира относительно машины

В алгоритме «сектор» плоскость движения разделяется на пять секторов (рис. 4), положение ориентира определяется углом между направлением на ориентир и продольной строительной осью машины).

Предложенные алгоритмы управления движением исследованы с помощью математической модели и апробированы на шагающей машине «Восьминог». Параметры математической модели динамики плоского движения [4] выбраны в соответствии с конструктивными параметрами машины «Восьминог». Анализ результатов моделирования (рис. 5) показывает, что энергетически менее затратные алгоритмы управления – это алгоритмы «зона» и «сектор» при $\beta = 180^\circ$. Однако, они более продолжительны по времени, а машина проходит больший путь. Это объясняется тем, что в этих алгоритмах большую часть времени работает только один привод.

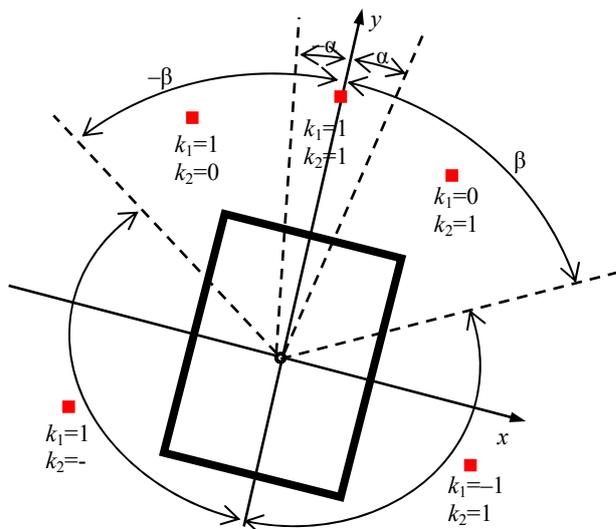


Рис. 4. Значения коэффициентов дискретного управления по алгоритму «сектор» в зависимости от положения ориентира относительно машины

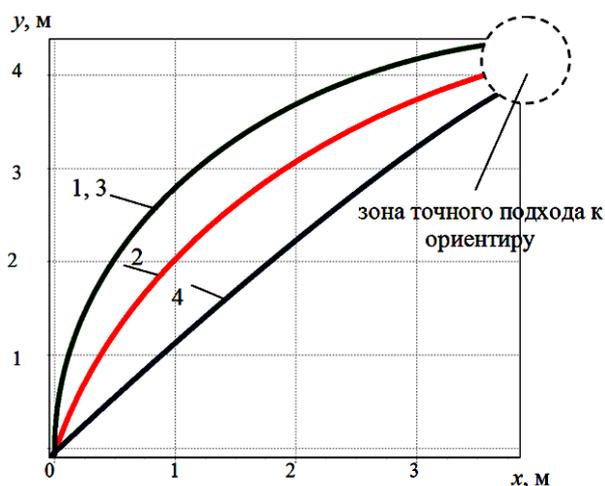
Поэтому, если решается задача быстрогодействия, то более выгодным является алгоритм «сектор» при $\beta = 0^\circ$, если задача энергосбережения, то алгоритм «зона» или «сектор» при $\beta = 180^\circ$.

В ходе эксперимента на реальной местности был реализован алгоритм «сектор» управления движением шагающей машины к «зоне точного подхода к ориентиру». При этом в процессе движения (рис. 6) для корректировки параметров управления движением с помощью бортовой видеосистемы [5] определялись координат неподвижных объектов (ориентиров) относительно шагающей

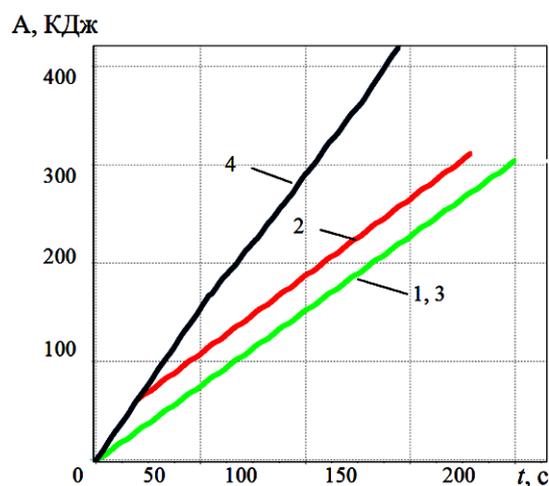
машины, вычисление расстояния до них и углов между продольной осью симметрии машины и объектами. Для обработки изображений использовалось разработанное программное обеспечение, позволяющее оператору определять необходимые параметры ориентиров относительно шагающей машины в реальном времени. Радиус «зоны точного подхода к ориентиру» был принят равным 12 м.

Во время движения при помощи бортовой видеосистемы измерялись координаты главного ориентира 1 через дискретные интервалы времени равные 30 с. Перемещение машины до «зоны точного подхода к ориентиру» составило 185 с. На рис. 6 в виде окон интерфейса программы оператора представлено несколько положений шагающей машины относительно главного ориентира 1 и примеры результатов измерений при ее движении к «зоне точного подхода».

Экспериментальные исследования показали, что предложенные алгоритмы управления движением машины позволяют осуществлять движение к «зоне точного подхода» с использованием бортовой видеосистемы и контролировать положение машины в режиме реального времени. При этом в процессе передвижения машины необходимо постоянно контролировать положение ориентира (препятствия) и проводить корректировку движения, управляя приводами механизмов шагания.



а) траектория движения центра масс машины



б) работа, затрачиваемая на перемещение

Рис. 5. Сравнение различных алгоритмов движения:

1 – алгоритм «зона», 2 – алгоритм «сектор» при $\beta = 45^\circ$, 3 – алгоритм «сектор» при $\beta = 180^\circ$, 4 – алгоритм «сектор» при $\beta = 0^\circ$



Время 16.04.30, угол на ориентир 14°, расстояние до ориентира 58 м.



Время 16.07.30, угол на ориентир 4°, расстояние до ориентира 27,2 м.



Время 16.10.00, угол на ориентир 0,4°, расстояние до ориентира 14,7 м.

Рис. 6 Интерфейс программы оператора при движении машины

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Концепция проектирования шагающих машин / Е. С. Брискин, В. В. Чернышев, В. В. Жога, А. В. Малолетов, Н. Г. Шаронов, Н. Е. Фролова // Наука – производство. – 2005. – № 1. – С. 33–38.
2. Шагающая машина «Восьминог» / Е. С. Брискин, В. В. Чернышев, А. В. Малолетов, В. В. Жога, Н. Г. Шаронов, К. В. Шаров, Н. Е. Фролова, Д. Н. Покровский // Мехатроника, автоматизация, управление. – 2004. – № 5. – С. 48–49.
3. Динамика управления роботами / Козлов В. В. [и др], под ред. Е. И. Юревича. М.: Наука, 1984. – С. 91–95.
4. Об общей динамике и повороте шагающих машин / Е. С. Брискин // Проблемы машиностроения и надежности машин. – 1997. – № 6. – С. 33–39.
5. Выбор алгоритма управления автономным движением шагающей машины с цикловыми движителями / Е. С. Брискин [и др] // Искусственный интеллект. 2007. № 3. С. 357–366.

УДК 629.1.03

*Н. Г. Шаронов, Я. В. Калинин***СИНТЕЗ АЛГОРИТМА УПРАВЛЕНИЯ ПРИВОДОМ ЦИКЛОВЫХ ДВИЖИТЕЛЕЙ ШАГАЮЩЕЙ МАШИНЫ В ОСОБЫХ УСЛОВИЯХ***Волгоградский государственный технический университет
dtm@vstu.ru

В статье сформулирован алгоритм выбора параметров походки шагающей машины с цикловыми движителями, которые позволяют осуществлять движение по заранее определенной следовой дорожке. Представлены результаты исследования предложенного алгоритма на опытной шагающей машине.

Ключевые слова: алгоритмы управления, цикловой движитель, шагающая машина.

*N. G. Sharonov, Y. V. Kalinin***SYNTHESIS OF CONTROL ALGORITHM OF ACTUATOR OF CYCLE MOVER OF WALKING MACHINE IN SPECIAL CONDITIONS**

Volgograd State Technical University

The control algorithm of parameters of gait allow to accomplish the motion of walking machine with cycle movers along previously determined road is formulated in the article. The results of research of the control algorithm by the experimental walking machine are submitted.

Key words: control algorithm, cycle mover, walking machine.

Одно из основных преимуществ шагающих машин перед машинами с традиционными типами движителей состоит в возможности выбора зон взаимодействия опорных стоп механизмов шагания с грунтом. Разработанная в ВолгГТУ шагающая машина «Восьминог» [1] предназначена для передвижения по относительно ровной поверхности. Поэтому ее кинематическая схема включает четыре пары цикловых, с одной степенью подвижности, механизмов шагания, и в каждой паре механизмы шагания работают в противофазе и образуют движитель. На каждом из бортов два движителя кинематически связаны общим приводом.

Шагающая машина «Восьминог М» [2] отличается наличием поворотного круга под днищем машины, на котором установлены две пары домкратов (подъемных движителей), что позволяет, вывешивая машину, осуществлять поворот без проскальзывания, а также изменять

положение стоп относительно грунта. Шагающая машина «Восьминог М2» [3] отличается наличием управляемого дифференциала (рис. 1) вместо раздаточного редуктора, что позволяет смещать походку одного движителя относительно другого на одном борту.

Алгоритм [3] управления приводами шагающей машиной «Восьминог М2» при преодолении запрещенных для наступания зон иллюстрируется схемой рис. 2.

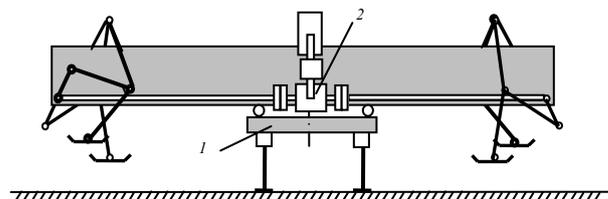


Рис. 1. Кинематическая схема шагающей машины «Восьминог М2»:

1 – поворотный круг; 2 – управляемый дифференциал

*Работа выполнена при финансовой поддержке РФФИ (09-08-97016).

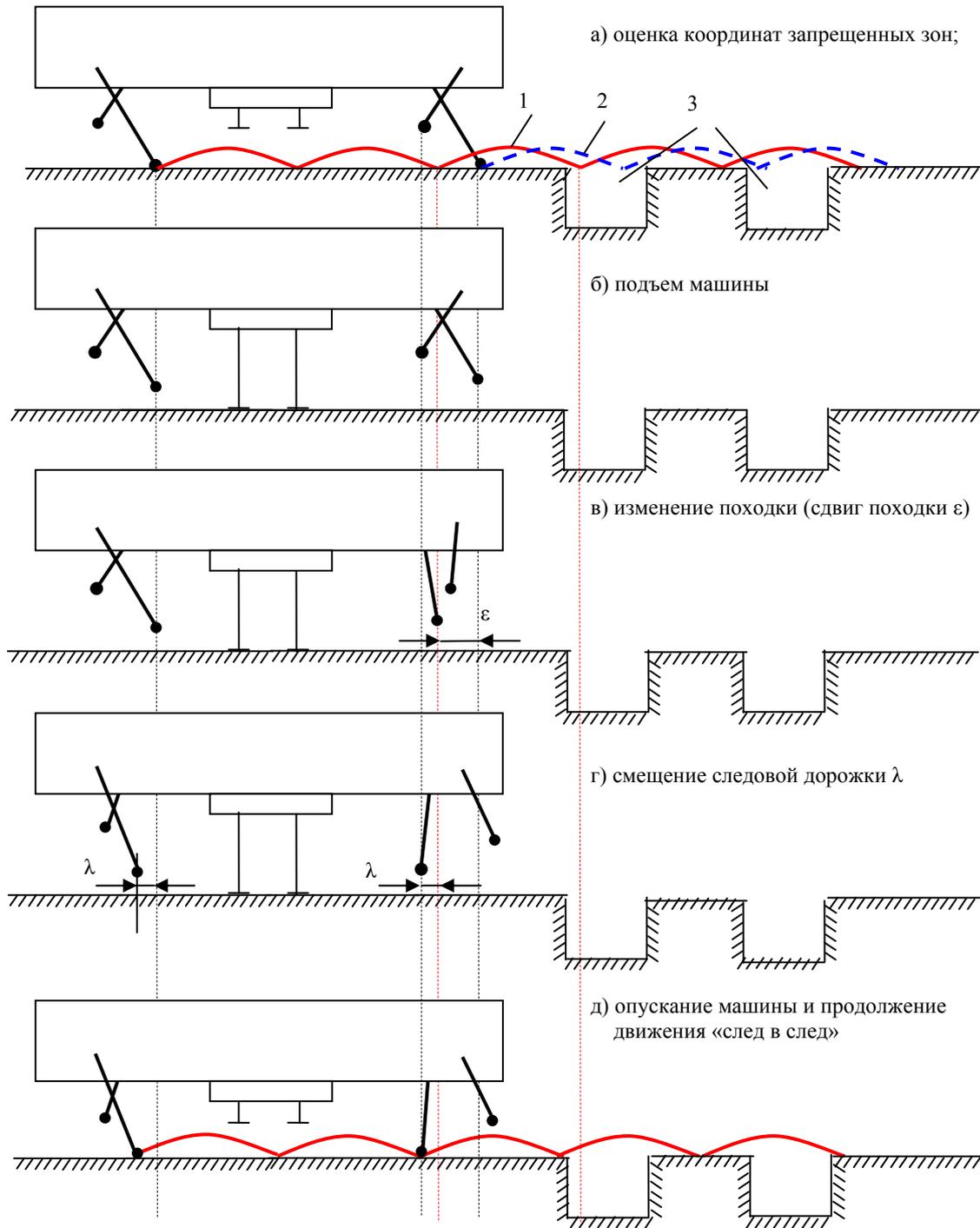


Рис. 2 – Алгоритм управления шагающей машиной «Восьминог М2» при преодолении запрещенных зон:
 1 – траектория опорных точек задних механизмов шагания; 2 – траектория опорных точек передних механизмов шагания; 3 – запрещенные зоны постановки опор

Для исследования алгоритма при переступании стопы через запрещенную зону (рис. 3) проведен эксперимент на опытном образце шагающей машины «Восьминог М2». Конструктивные параметры машины «Восьминог М2»: длина шага в относительном движении $l = 0,74$ м; минимальное рас-

стояние между траекториями опорных точек передних и задних движителей одного борта $\Delta = 2,8$ м; следовая дорожка правого S_{np} и левого S_l движителей (в метрах) описывается последовательностью

$$S_l = \{1,48 - \lambda, 4,28 - \lambda, 2,96 - \lambda, 5,76 - \lambda, \dots\}$$

$$S_{np} = \{0,74 - \lambda, 2,22 - \lambda, 5,02 - \lambda, 3,7 - \lambda, 6,5 - \lambda, \dots\}$$



Маркер стопы с уменьшенной длиной (0,2 м)

Маркер зоны запрещенной для наступания (длина 0,2 м)

Рис. 3. Установка маркеров стопы и препятствия

В процессе движения фиксируются координаты границ запрещенных зон (a_k, b_k) относительно системы координат, связанной с машиной. При достижении границы запрещенной зоны движение прекращается (рис. 2, а). Так как размер маркера запрещенной зоны $b_k - a_k = 0,2$ м меньше длины шага в относительном движении $l = 0,74$ м с учетом длины маркера стопы $l_{cm} = 0,2$ м, т. е. $b_k - a_k < l - l_{cm}$, то после изменения параметров походки запрещенную зону возможно преодолеть.

Для изменения параметров походки определяется величина сдвига передних движителей ε (относительного перемещения опорной точки переднего механизма шагания). После подъема машины домкратами (рис. 2, б), с помощью управляемого дифференциала включается привод переднего движителя курсового движения и перемещается при неподвижной машине (рис. 2, в). Изменение положения опорной точки механизма шагания показано на рис. 4. Определяется величина смещения следовой дорожки λ (относительного перемещения опорной точки механизма шагания) и соответствующее ей значение углов поворота кривошипов. На рис. 5 показано, как приводом курсового движения машины опорная точка перемещается на вывешенной машине (рис. 2, г). Корпус машины с помощью приводов дополнительных подъемных движителей опускается, далее включается привод движителей курсового движения и машина перемещается.

Экспериментальные исследования предлагаемого алгоритма управления параметрами походки, проведенные на натурном образце шагающей машины «Восьминог М», показали, что шагающие машины с цикловыми движителями, несмотря на достаточно простую конструкцию движителей и систему управления в состоянии обеспечить режимы движения, характерные для машин профильной проходимости, имеющих движители с тремя и более управляемыми степенями подвижности. При-

чем в маршевых режимах, при движении по сравнительно ровной поверхности без ограничений на место постановки стоп движение осуществляется с применением только двух приводов курсового движения [4].



неподвижные механизмы шагания заднего движителя подвижные механизмы шагания переднего движителя

Рис. 4. Изменение положения опорной точки механизма шагания



Рис. 5. Изменение положения опорной точки механизмов шагания

Методика подобных расчетов обосновывается тем, что структура кинетической энергии шагающей машины и переносимого движителя достаточно проста и при составлении уравнений Лагранжа I рода достаточно применять один неопределенный множитель Лагранжа [5, 6].

Также следует отметить, что при таком преодолении запрещенных зон нарушается оптимальное с точки зрения минимума тепловых потерь в приводном двигателе машины движение машины [7], так как происходит остановка и последующий разгон движителей, а также подъем-опускание корпуса машины. Все эти переходные и неустановившиеся процессы вызывают повышенный расход энергии, которая безвозвратно рассеивается в тепло.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Концепция проектирования шагающих машин / Е. С. Брискин, В. В. Чернышев, В. В. Жога, А. В. Малолетов, Н. Г. Шаронов, Н. Е. Фролова // Наука – производству. – 2005. – № 1. – С. 33–38.
2. Концепция проектирования, динамика и управление движением шагающих машин. Ч.2. Динамика движения шагающих машин серии «Восьминог» / Е. С. Брискин, В. В. Чернышев, В. В. Жога, А. В. Малолетов, Н. Г. Шаронов, Н. Е. Фролова // Мехатроника, автоматизация, управление. – 2005. – № 6. – С. 19–26.
3. Об управлении походкой шагающей машины «Восьминог» / Е. С. Брискин // Мехатроника, автоматизация, управление: приложение к журналу. – 2008. – № 5. – С. 6–10.
4. Выбор алгоритма управления автономным движением шагающей машины с цикловыми движителями / Е. С. Брискин [и др] // Искусственный интеллект. 2007. № 3. – С. 357–366.
5. Об общей динамике и повороте шагающих машин / Е. С. Брискин // Проблемы машиностроения и надежности машин. – 1997. – № 6. – С. 33–39.
6. О выборе обобщенных координат при исследовании движения механических систем / Е. С. Брискин // Известия РАН. Механика твердого тела. – 2000. – № 2. – С. 44–50.
7. Оценка энергетической эффективности шагающих машин с цикловыми движителями / Я. В. Калинин // Известия ВолгГТУ. Серия «Актуальные проблемы управления, вычислительной техники и информатики в технических системах». Вып. 7: межвуз. сб. науч. ст. / ВолгГТУ. – Волгоград, 2009. – № 12. – С. 40–44.

СИСТЕМНЫЙ АНАЛИЗ И ОБРАБОТКА ИНФОРМАЦИИ

УДК 004.021

Е. М. Артемьев, О. О. Привалов, И. В. Степанченко, В. О. Петров

ПРИМЕНЕНИЕ АЛГОРИТМОВ МАШИННОГО ЗРЕНИЯ ДЛЯ ПОЛУЧЕНИЯ ПАНОРАМНОГО ИЗОБРАЖЕНИЯ МЕДИКО-БИОЛОГИЧЕСКОГО ПРЕПАРАТА ПРИ МИКРОСКОПИИ

Камышинский технологический институт (филиал)
Государственного образовательного учреждения высшего профессионального образования
«Волгоградский государственный технический университет»
galva@mail.ru, stilvi@mail.ru, krabler@rambler.ru

Описывается метод создания панорамы изображений, захваченных с окуляра микроскопа с помощью алгоритмов машинного зрения. Входными данными метода является серия изображений соседних полей микроскопа, а выходными данными – комбинированное изображение, содержащее информацию со всех полей. Разработанный метод исследован на скорость работы и качество получаемого изображения.

Ключевые слова: микроскопия, панорама, соединение изображений, машинное зрение.

E. M. Artemev, O. O. Privalov, I. V. Stepanchenko, V. O. Petrov

APPLICATION OF COMPUTER VISION ALGORITHMS FOR CREATION PANORAMA OF MICROSCOPE FIELD

Kamyshin Technological Institute (branch of) Volgograd State Technical University

The method to make panorama of microscope field using computer vision algorithms is considered. This method takes a series images of neighbor microscopy field and makes combined image that contain information from all fields. There were used computer vision algorithms. The developed method is researched for speed of operation and quality of the resulting image.

Key words: microscopy, panorama, concatenated images, computer vision.

Принципиальным недостатком микроскопии медико-биологических препаратов является отсутствие возможности охвата больших полей наблюдения при увеличении детализации объектов интереса [1, 2].

Решение данной проблемы возможно за счет получения серии мелких полей изображения с большим увеличением и последующего их объединения в единое изображение, которое будет охватывать весь объект и при этом иметь достаточное для анализа увеличение.

Постановка задачи. Целью является реализация возможности получения крупномасштабного растрового изображения медико-биологического препарата требуемой детализацией путем объединения серии изображений полей малого размера в автоматическом режиме. Обязательным условием является наличие общих фрагментов на соседних изображе-

ниях серии. При этом анализ возможен только для препаратов, состоящих из биологических субстанций период модификации которых (движение объектов, изменение размера частиц и т. д.) значительно меньше времени сканирования.

Основная идея алгоритма, позволяющего собрать фрагменты единого поля наблюдения в результирующее изображение, представлена на рис. 1.

В качестве метода для поиска общих ключевых точек были исследованы такие алгоритмы поиска локальных особенностей как: SURF [3], SIFT [4], PCA-SIFT [5], GLOH [6]. Параметры алгоритмов были настроены так, чтобы алгоритмы нашли примерно равное общее количество точек. Сравнение алгоритмов осуществлялось по времени вычисления и проценту правильно распознанных точек на контрольных изображениях.



Рис. 1. Схема алгоритма составления панорамного изображения

Конечная реализация системы выполнена с использованием алгоритма SURF (Speeded Up Robust Features), так как эксперименты (табл. 1) показали, что он имеет наибольшее быстродействие и точность.

Таблица 1

Результаты сравнения алгоритмов

Алгоритм	Время вычисления (мс)	Кол-во точек	Правильных ответов
SURF	350	1418	82,6 %
SIFT	1037	1520	78,1 %
PCA-SIFT	1259	1515	72,3 %
GLOH	988	1478	78,3 %

Для установления параметрической зависимости положений ключевых точек между изображениями был применен алгоритм RANSAC [7] (Random Sample Consensus) – «согласование случайных выборок». Это – итерационный метод оценивания параметров математической модели по набору наблюдаемых данных, которые содержит выбросы. Целесообразность его применения объясняется тем, что ключевые точки могут содержать ошибочные соответствия, а простые методы параметрической идентификации, такие как метод наименьших квадратов, даже при наличии единичного выброса в исходных данных могут дать существенную ошибку в модели.

С помощью вычисленных на предыдущем этапе параметров, составляется матрица аффинного преобразования (1), приводящая координаты ключевых точек одного изображения в соответствие с координатами точек другого изображения.

$$T = \begin{pmatrix} a & b & 0 \\ -b & a & 0 \\ c & d & 1 \end{pmatrix}, \quad (1)$$

где: a – коэффициент масштаба, b – коэффициент поворота, c – горизонтальный сдвиг, d – вертикальный сдвиг.

Последним шагом является добавление текущего изображения на результирующее панорамное изображение с применением полученной матрицы аффинных преобразований.

Для проверки работы алгоритма были взяты четыре пересекающихся поля зрения с одного препарата крови (рис. 2).

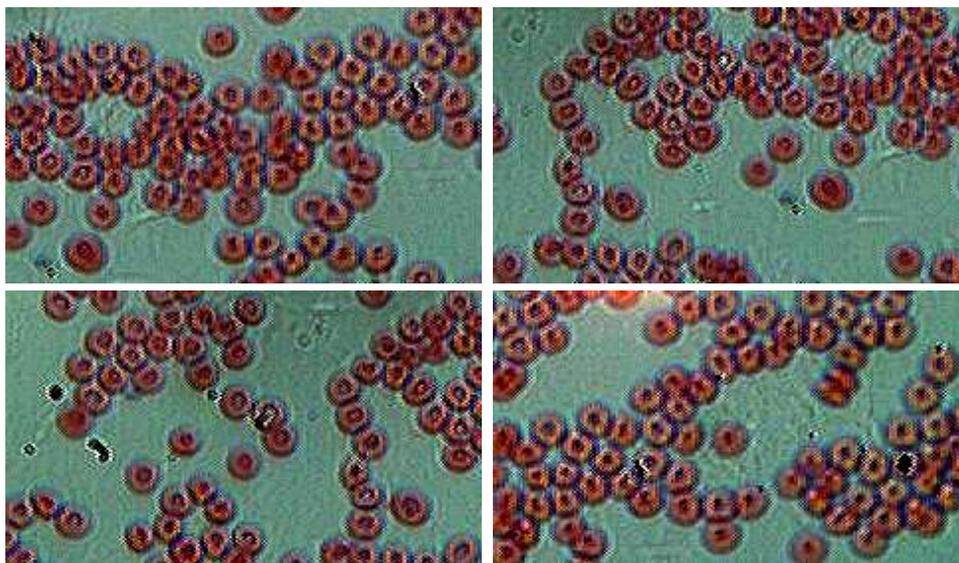


Рис. 2. Тестовые поля зрения

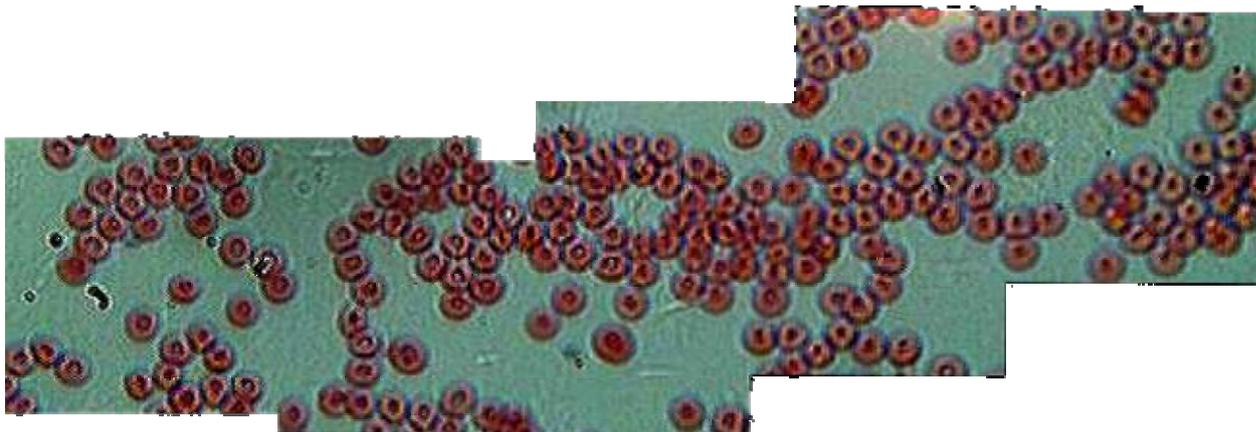


Рис. 3. Результат вычисления

Полученные поля зрения были обработаны разработанным алгоритмом, была выполнена генерация виртуального поля зрения, результат вычисления приведен на рис. 3.

Из рисунка видно, что результат вычисления представляет собой корректно объединенные части изображения препарата. Таким образом, можно сделать вывод, что реализованный алгоритм вполне справляется с поставленной задачей.

З а к л ю ч е н и е. Следует отметить, что данный алгоритм обладает устойчивостью к наличию артефактов на отдельных фрагментах анализируемых изображений, при условии, что артефакт носит локальный характер и не искажает большую часть наблюдаемой картины (доля искажений зависит от типа сканируемого препарата). Устойчивость обусловлена автоматическим исключением тех характеристических точек, которые не являются общими для группы изображений составляющих единое поле.

Разработанный алгоритм реализован в виде отдельной функции (модуля) в составе универсального программного комплекса для обработки информации, полученной в результате медико-биологической диагностики.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Камаев, В. А.* Многомодульная система медицинской диагностики / В. А. Камаев, Б. Н. Филатов, Д. П. Панченко // Вестник компьютерных и информационных технологий. – 2005. – № 1. – С. 17–19.
2. *Петров, В. О.* Методика автоматизированного анализа фации перито-неальной жидкости по растровому изображению при микроскопии / В. О. Петров, О. О. Привалов, И. В. Степанченко // Научно-технические ведомости СПбГПУ; сб. науч. статей / СПбГПУ. – Санкт-Петербург, 2009. – № 3. – С. 134–138.
3. *Bay, H.* SURF: Speeded Up Robust Features / H. Bay, A. Ess, T. Tuytelaars // Computer Vision and Image Understanding (CVIU) – 2008. – Vol. 110, – No. 3. – PP. 346–359.
4. *Lowe, D. G.* Object recognition from local scale-invariant features / D. G. Lowe // International Conference on Computer Vision – 1999. – September.
5. *Ke, Y.* PCA-SIFT: A More Distinctive Representation for Local Image Descriptors / Y. Ke, R. Sukthankar // Computer Vision and Pattern Recognition – 2004.
6. *Mikolajczyk, K. A.* Performance evaluation of local descriptors / K. Mikolajczyk, C. Schmid // IEEE Transactions on Pattern Analysis and Machine Intelligence – 2005. – Vol. 10, – No. 27. – PP. 1615–1630.
7. *Fischler, M. A.* RANdom SAMpling Consensus: a paradigm fro model fitting with application to image analysis and automated cartography / M. A. Fischler, R. C. Bolles, // Commun.Assoc.Comp. – 1981. – No. 24. – PP. 381–95.

УДК 7.011 681.51

*Дм. В. Бутенко, Л. Н. Бутенко, Я. С. Кошечкин, М. С. Козлов, А. Д. Прошин***КОНЦЕПТУАЛЬНЫЙ АНАЛИЗ СИСТЕМЫ АНТИКРИЗИСНОГО УПРАВЛЕНИЯ****Волгоградский государственный технический университет**
jasha88@list.ru

Проводится концептуальный анализ понятия антикризисное управление системой. Отдельно рассматривается каждая его составляющая и выявляется интенционал. Полученные результаты позволяют выявить задачи, которые необходимо решить, чтобы обеспечить антикризисное управление системой.

Ключевые слова: антикризисное управление системой, концептуальный анализ, гомеостаз, кризис.

*Dm. V. Butenko, L. N. Butenko, I. S. Koshechkin, M. S. Kozlov, A. D. Proshin***THE CONCEPTUAL ANALYSIS OF ANTI-RECESSIONARY MANAGEMENT SYSTEM****Volgograd State Technical University**

The conceptual analysis of the term anti-recessionary management of system is carried out. Its each component is separately considered and the essence comes to light. The received results allow to reveal problems which are necessary for deciding to provide anti-recessionary management of system.

Key words: anti-recessionary management of system, the conceptual analysis, a homeostasis, crisis.

Концептуальные методы, развиваемые в настоящее время школой С. П. Никанорова [1], А. Г. Теслиновым [2], В. И. Разумовым [3] предоставляют мощный аппарат для переноса знаний между различными, иногда далеко отстоящими друг от друга, предметными областями, что позволяет создать многоаспектное представление об изучаемой области и увидеть другие горизонты. Анализ по этим методам проводится посредством выделения наиболее общих понятий, относящихся к формулированию задачи. Предельно общие понятия называются категориями. Категории являются метаязыковыми когнитивными образованиями, к которым относятся дефиниции классов понятий, используются в задачах систематизации знаний и познавательного процесса. Категории фиксируют классы знания, этапы и факторы познавательных процессов, поэтому они входят в систему управления знаниями [4]. Начнем с определения категорий в нашей тематике. Проведем концептуальный анализ понятия «антикризисное управление системой» с помощью рассмотрения его составляющих.

Понятие «кризис» имеет много различных определений и используется в различных сферах, например, финансовой, политической, энергетической, психологической, экологической и т. д. Кризис:

потеря гомеостаза значительной частью экосистемы;

происходящие в психике процессы, связанные с необходимостью изменения модели поведения;

ситуация, когда нарушен баланс сил в стране или регионе и оппоненты обостряют конфликт, надеясь улучшить собственное положение;

нарушение равновесия между спросом и предложением на товары и услуги. В построенной на конкуренции экономике посредством кризиса происходит массовая селекция эффективных собственников;

ситуация, когда рост фиктивного капитала значительно опережает рост капитала реально. Переоценка стратегий сбережения и связанный с нею кризис ликвидности происходит лавинообразно, вынуждая прибегать к санации в масштабах всей экономики.

Из всех перечисленных определений кризиса выделяются следующие составляющие: гомеостаз, исчезновение, баланс сил, конфликт, нарушение равновесия, рост, лавина. Таким образом, можно сформировать определение, обобщающее вышеперечисленные определения. Кризис (греч. *krisis* – решение, поворотный пункт) – перелом, переворот или пора переходного состояния, когда система теряет свои гомеостатические свойства, теряется баланс сил или нарушается равновесие во взаимодействии между элементами системы, приводящие к изменению стратегии поведения системы и появлению новых видов взаимодействия и приобретению системой новых свойств. Процесс происходит лавинообразно и рождает непредсказуемые проблемы для управления им.

Анти (греческое *anti* – против, вместо), приставка, означающая: противоположность, вра-

ждебность чему-либо, направленность против чего-нибудь.

Понятие «управление» имеет также множество определений:

- это воздействия, направленные на достижение цели;

- управление укрупнено рассматривается в составе основных функций, замкнутых в цикл планирования и оперативного управления. Для понимания концепта управление необходимо дать определение:

планирование – определение желаемого состояния и пути его достижения;

учет – фиксация отклонения от запланированных целей;

контроль и анализ – определение проблемных ситуаций;

регулирование или оперативное управление – принятие и осуществление решения по ликвидации отклонения, как разрешение проблемной ситуации.

Соотношение между планированием и оперативным управлением может быть представлено на стягивающем графе.

Антикризисное управление также необходимо рассматривать в качестве системы. Приведем определение системы. Система (от др. греч. σύνστημα – «сочетание») – множество взаимосвязанных объектов и ресурсов, организованных процессом системогенеза в единое целое и, возможно, противопоставляемое среде или суперсистеме [5]. Целое существует в определенных границах в состоянии равновесия противодействующих сил [6]. Система в системном анализе – совокупность сущностей (объектов) и связей между ними, выделенных из среды на определенное время и с определенной целью.

Для построения интенционала понятия «антикризисное управление» необходимо построить интенционал понятия «антикризиса». Для этого необходимо инвертировать все понятия, сопряженные с понятием «кризиса». В этом случае «антикризис» также представляет собой переходное состояние, целью которого является восстановление гомеостатических свойств, баланса сил и равновесия во взаимодействии между элементами системы. Для этого необходимо определить параметры системы в состоянии «антикризис», т. е. фактически построить полюса гомеостата по кризисным параметрам системы, тогда система управления с «кризисным» и «антикризисным» полярными состоя-

ниями будет являться антикризисной системой управления. Следовательно, необходимо проводить анализ состояния системы, чтобы предупредить кризисное состояние, для этого нужно следить за тем, чтобы все ключевые состояния не выходили за границы допустимых значений. Тогда удержание параметров системы в состоянии равновесия как раз и будет антикризисным управлением [7]. Отметим также, что выход из кризиса может преследовать разные цели: восстановление параметров ранее достигнутых системой или переход системы в новое качественное состояние с другими параметрами. Система антикризисного управления должна быть дополнена подсистемой прогнозирования, которая должна определять параметры состояний с новыми качественными свойствами.

Результаты анализа концептов, входящих в понятие «антикризисное управление системой» позволяет сформировать его интенционал. Антикризисным управлением системы является система гомеостатического типа, обеспечивающая возврат исходной системы в состояние динамической устойчивости ее параметров при взаимодействии с внешней средой и удержание ее в состоянии ранее достигнутого равновесия (до кризиса), либо перевод системы в новое качественное состояние. Исходя из этого определения, можно выделить требование к системе антикризисного управления:

- 1) мониторинг текущих значений параметров системы;

- 2) построение подсистем управления гомеостатического типа по ключевым параметрам систем;

- 3) прогнозирование поведения системы в результате агрессивного воздействия окружающей среды. Для этого необходимо прокреативное видение внешнего воздействия (предвидение возможных негативных влияний и создание резерва для непредвиденных ситуаций для поддержания главной целевой функции);

- 4) подсистема определения проблемных ситуаций процесса системогенеза, построение их моделей;

- 5) подсистема стратегического и тактического планирования для поведения системы в вышеопределенных состояниях.

Отметим, что решение совокупности этих задач позволит обеспечить создание системы управления стабильным развитием систем любого типа.

Таким образом, проведен концептуальный анализ концептов понятия «антикризисное управление системой», построен его интенционал и определены задачи создания системы антикризисного управления.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Никаноров С. П. Концептуализация предметных областей. М.: Концепт, 2009. – 268 с.
2. Теслинов А. Г. Концептуальное мышление в решении сложных и запутанных проблем. – СПб.: Питер, 2009. – 288 с.
3. Разумов В. И., Сизиков В. П. Категориальный аппарат в современном естествознании. Режим доступа: http://www.philosophy.nsc.ru/journals/philsience/1_04/00_razumov.htm.

4. Википедия. Режим доступа: <http://ru.wikipedia.org/>
5. Перегудов Ф.И., Тарасенко Ф.П. Введение в системный анализ. – М.: Высшая школа, 1989.
6. Бутенко Д. В., Бутенко Л. Н. Задача Стратегического планирования и Теория развития систем // XXX Юбилейная Международная конференция и дискуссионный научный клуб IT+SE'2003 Новые информационные технологии в науке, образовании, телекоммуникации и бизнесе. Украина, Крым, Ялта-Гурзуф, 2003 г.
7. Бутенко Д. В., Бутенко Л. Н., Кошечкин Я. С. Концептуальная модель оценки финансово-экономической устойчивости предприятия // XXXVII Международная конференция и дискуссионный научный клуб IT+SE'2010 Информационные технологии в науке, образовании, телекоммуникации и бизнесе. Украина, Крым, Ялта-Гурзуф, 2010 г.

УДК 343.983.25

П. Е. Гребенюк, П. В. Зотов, А. М. Чмутин, Р. Р. Шарипов

ПРОГРАММНОЕ ВМЕШАТЕЛЬСТВО В ЦИФРОВЫЕ ИЗОБРАЖЕНИЯ И ЕГО ИДЕНТИФИКАЦИЯ

Волгоградский государственный университет
Волгоградский государственный технический университет
grebenyuk-vlg@yandex.ru, rezet-66@yandex.ru,
amchmutin@yandex.ru, james1401@yandex.ru

Рассмотрена задача установления подлинности цифровых фотографических изображений. Показано, что любое программное вмешательство приводит к артефакту – необратимо меняется структура числовой записи изображения. Доказано, что исключения из выявленной закономерности ничтожны. Предложен метод идентификации факта вмешательства. Описан оригинальный программный продукт.

Ключевые слова: цифровое изображение, графическая информация, редактирование, идентификация, экспертные приложения.

P. E. Grebenyuk, P. V. Zotov, A. M. Chmutin, R. R. Sharipov

PROGRAM BREAK INTO DIGITAL IMAGES AND IT'S IDENTIFIATION

Volgograd State University
Volgograd State Technical University

The task to establish the authenticity of the digital photographic images is considered. It is shown, that any program intervention results in artifact – the structure of the image numerical record irreversibly varies. It is proved, that the exceptions of the revealed effect are insignificant. The method to identify the fact of intervention is offered. The original software is described.

Key words: digital image, graphic information, editing, identification, expert applications.

Нередко особое значение при расследовании и раскрытии преступлений имеет использование информации, содержащейся в цифровых фотографических изображениях. Очевидно, что от достоверности этой информации зависит многое: в принципиальных вопросах может идти речь о виновности или невиновности обвиняемого, в процедурных – лишь о признании или непризнании конкретного цифрового фото уликой. Следовательно, встает проблема установления подлинности того или иного снимка. Настоящее сообщение и посвящено

выявлению артефактов, возникающих вследствие редактирования цифровых фотографических изображений в прикладном графическом пакете Photoshop. Последний сегодня является наиболее распространенным и доступным из программных средств растровой графики. Кроме того, по оценкам специалистов [1] Photoshop фирмы Adobe является «наиболее профессиональным» в соответствующем сегменте программного продукта. Соответственно, большинство известных подделок цифровых фото сделано с его помощью.

В памяти цифрового фотоаппарата изображения могут сохраняться во множестве фирменных форматов (NEF у Nikon, CRW у Canon, ORF у Olympus, MRW у Minolta, KDC у Kodak, ..., всевозможные прочие RAW). На выходе RAW переводится в универсальный (аппаратно-независимый) формат, в котором файл и существует вплоть до утилизации. Проанализировав универсальные форматы, с которыми может столкнуться эксперт, мы констатируем рост применения формата PNG для хранения цифровых фотографических изображений и прогнозируем именно его наибольшую перспективу в этой сфере. Тем более, использование формата PNG не сопряжено с патентными отчислениями ни для производителей фототехники, ни для разработчиков программного продукта.

Сравнительному анализу подвергались числовые матрицы, полученные из графических файлов в формате PNG с 48-битным цветовым разрешением. Контрольная группа этих файлов непосредственно импортировалась из памяти цифрового фотоаппарата Nikon D60 с помощью программы Nikon Capture NX. Испытуемая группа подвергалась дополнительной обработке в программном пакете Photoshop (были использованы все версии, начиная с Photoshop 5.5 и заканчивая Photoshop CS3). Для этой обработки в нашем исследовании были опробованы практически все штатные инструменты этого программного пакета.

В процессе обработки изображение: редактировалось; редактировалось и возвращалось к исходному виду; не редактировалось. Таким образом, под обработкой понимались, как минимум, две процедуры – открытие и сохранение файла. При обработке исходный PNG-формат файла: не менялся; менялся, например, на RAW, TIF, PDF, PSD, ..., одним словом, на все возможные, которые имеются в распоряжении пользователя Photoshop. В этом случае последнее в череде преобразований сохранение файла возвращало его в исходный формат.

Оригинальной программой PNG-файл из первой или второй группы конвертировался в числовые матрицы (R-матрица, G-матрица и B-матрица). Пример этого преобразования иллюстрируется рис. 1 и рис. 2.

По очевидным соображениям пример здесь, естественно, модельный. Изображение выбрано черно-белое, а для такового RGB-матрицы идентичны, поэтому на рис. 2 приведена только одна.

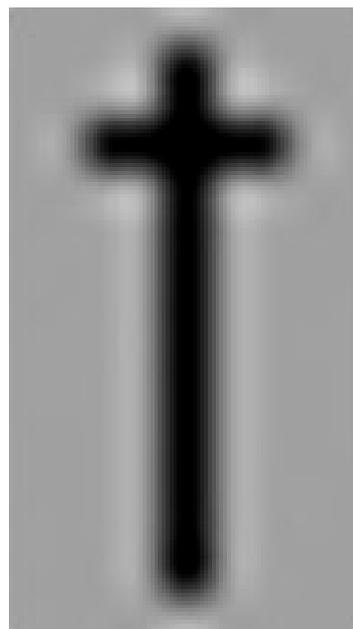


Рис. 1. Оригинал (контрольная группа)

```

9x16
49152 49151 49150 49149 49148 49147 49146 49145 49144
49151 49150 49149 49148      0 49146 49145 49144 49143
49150 49149 49148 49147      1 49145 49144 49143 49142
49149 49148      0      1      2      1      0 49142 49141
49148 49147 49146 49145      3 49143 49142 49141 49140
49147 49146 49145 49144      4 49142 49141 49140 49139
49146 49145 49144 49143      5 49141 49140 49139 49138
49145 49144 49143 49142      6 49140 49139 49138 49137
49144 49143 49142 49141      7 49139 49138 49137 49136
49143 49142 49141 49140      8 49138 49137 49136 49135
49142 49141 49140 49139      9 49137 49136 49135 49134
49141 49140 49139 49138     10 49136 49135 49134 49133
49140 49139 49138 49137     11 49135 49134 49133 49132
49139 49138 49137 49136     12 49134 49133 49132 49131
49138 49137 49136 49135     13 49133 49132 49131 49130
49137 49136 49135 49134 49133 49132 49131 49130 49129

```

Рис. 2. Результат (контрольная группа)

Цифры в первой строке матрицы – ширина и высота изображения, выраженные в pxl, – информация служебная (упомянутый конвертер в составе многофункционального пакета программных модулей был написан задолго до постановки задачи настоящего исследования и сейчас используется в учебном процессе физического факультета ВолГУ при подготовке экспертов-криминалистов по специальности 030502 [4]).

Далее работа шла с числовыми матрицами, – объектом исследования являлся массив цветковых координат. Отыскивались четные и нечетные значения цветковых координат, для чего сначала использовался простейший инструмент из Windows, Total или Norton Commander, позднее (для работы с большими файлами) была разработана специальная программа, – на

ней мы остановимся ниже. В процессе исследования было выявлено, что любая обработка исходного файла в Photoshop привносит следующий артефакт. Цветовые координаты, не превышающие середину динамического диапазона (32767), округляются до ближайших четных значений, а превышающие – до ближайших нечетных. Установленный артефакт иллюстрируется сравнением матриц на рис. 3 и рис. 2. Таким образом, алгоритм исследования заключался в соотношении четных/нечетных значений цветовых координат каждого пикселя с нижней/верхней половиной их динамического диапазона. Факт обработки считался выявленным, если в верхней не оказывалось четных значений, а в нижней – нечетных.

9x16									
49151	49151	49149	49149	49147	49147	49145	49145	49143	
49151	49149	49149	49147	0	49145	49145	49143	49143	
49149	49149	49147	49147	2	49145	49143	49143	49141	
49149	49147	0	2	2	2	0	49141	49141	
49147	49147	49145	49145	4	49143	49141	49141	49139	
49147	49145	49145	49143	4	49141	49141	49139	49139	
49145	49145	49143	49143	6	49141	49139	49139	49137	
49145	49143	49143	49141	6	49139	49139	49137	49137	
49143	49143	49141	49141	8	49139	49137	49137	49135	
49143	49141	49141	49139	8	49137	49137	49135	49135	
49141	49141	49139	49139	10	49137	49135	49135	49133	
49141	49139	49139	49137	10	49135	49135	49133	49133	
49139	49139	49137	49137	12	49135	49133	49133	49131	
49139	49137	49137	49135	12	49133	49133	49131	49131	
49137	49137	49135	49135	14	49133	49131	49131	49129	
49137	49135	49135	49133	49133	49131	49131	49129	49129	

Рис. 3. Артефакт (испытываемая группа)

Этот артефакт невозможно заметить, рассматривая обработанное изображение невооруженным глазом (зрение позволяет разрешить в лучшем случае в 200–300 раз большие изменения), он может быть выявлен только при анализе числовой матрицы. Поэтому мы и приводим только матрицу, а не само обработанное в Photoshop изображение. Таким образом, обнаружение данного артефакта в любой из матриц цифрового фотографического изображения дает основание полагать, что оно подвергалось воздействию средств Photoshop, и наоборот.

Здесь необходимо особо подчеркнуть, что при проведении исследования изображения по описанному выше алгоритму рассматривается только графическое наполнение файла, а не его служебные (EXIF) данные. Последних, вообще говоря, у файла может и не оказаться, особенно в тех случаях, когда:

- а) цифровая фотография сделана, к примеру, случайным свидетелем, а не профессиональным экспертом;
- б) ее происхождение не установлено.

Описание исследования завершим следующим резюме: выявленный артефакт имеет объективный характер, поскольку изученный признак воздействия на файл необратим, и достоверен, так как не связан с широко [5] доступной возможностью редактирования EXIF.

Однако, цифровое фотографическое изображение по своей природе стохастично, – это значит, что теоретически такая ситуация, когда и без вмешательства Photoshop все малые цветовые координаты – четные, а большие – нечетные, не может быть исключена. Как следствие, чтобы иметь право использовать найденный артефакт, требуется оценить его неоднозначность. Мы посчитали ее, как вероятность сфотографировать такой объект, изображение которого и без какой либо обработки удовлетворяет артефакту. По [6] такая вероятность определяется выражением

$$P = (\frac{1}{2})^{3N}$$

где $\frac{1}{2}$ – вероятность того, что цветовая координата любого пикселя примет четное/нечетное значение, 3 – число цветовых координат, N – число пикселей в исследуемом изображении. Очевидно, что уже при 100 pxl (это много меньше размера ПЗС-матрицы в камере самого дешевого мобильного телефона и много ниже наихудших характеристик простейшего сканера) вероятность ошибки идентификации артефакта пренебрежимо мала.

Исследуя происхождение артефакта, мы пришли к заключению, что следует говорить не об особенностях формата PNG, как такового, а об особенностях функционирования программного пакета Photoshop в целом. Мы имеем веские основания полагать, что выявленный артефакт отнюдь не ограничен предметной областью цифровых изображений в формате PNG, а распространяется на все 48-битные форматы растровой графики. Намеренно не касаясь патентованных форматов, мы декодировали только открытый формат PNG, имея в виду перспективу программно реализовать возможность идентификации Photoshop-воздействия на файл.

Как проведенные числовые оценки, так и изложенные соображения позволяют сделать однозначный вывод об устойчивости выявленно-го артефакта.

Упомянутая выше специальная программа, изначально разработанная для решения сугубо исследовательской задачи, была позднее трансформирована в оригинальный программный продукт, сориентированный на уровень подготовки штатного эксперта. Он реализован в среде Microsoft Visual Studio.

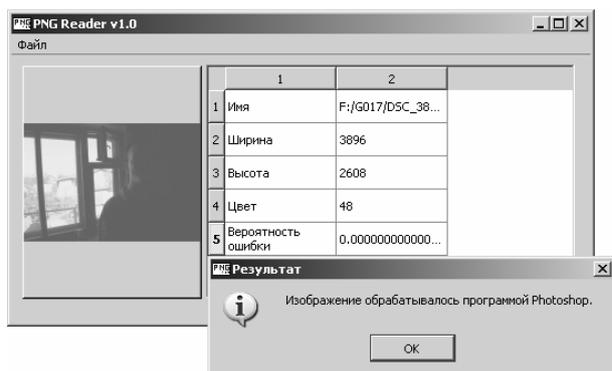


Рис. 4. Интерфейс программы (экранная копия)

Имеется возможность обрабатывать изображения размерами 10000×10000 pxl, – исследование такого файла занимает менее 2 мин (на компьютере с оперативной памятью 1 Гбайт и тактовой частотой 2,8 ГГц). Установка пяти-мегабайтного программного продукта (Windows XP, Windows Vista, Windows 7) занимает 30 с, – остальные системные требования не существенны. Особая функциональная возможность программы – это индикация вероятностной оценки неоднозначности вывода применительно к исследуемому изображению.

Программа прошла апробацию в ЭКЦ Волгоградского УВД [7] – напрашивается вывод, что она может использоваться как в оперативно-розыскной, так и в экспертной деятельности. Сегодня для нас главное направление дальнейшего развития работ – патентование созданного программного продукта.

Результаты проведенного исследования показали, что распределение четных и нечетных значений цветовых координат может служить устойчивым идентификационным признаком факта обработки графического файла в програм-

мном пакете Photoshop. Уточним, факт редактирования изображения (с возможным изменением криминалистически значимой информации путем добавления в изображение детали, ранее отсутствовавшей, сокрытия в изображении детали, ранее имевшей место, и замены в изображении одной детали другой, однотипной по внешнему виду) таким способом установлен быть не может. Категоричным может быть только вывод, что данный файл подвергался какой либо обработке (с сохранением результатов) в Photoshop. Однако уже этот факт может иметь весьма существенное значение в суде.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. [HTTP://www.schoolphotography.ru](http://www.schoolphotography.ru).
2. Чмутин А. М., Шарипов Р. Р. Исследование целостности цифровых фотоизображений методом вейвлет-анализа // Сб. докл. 19 Межд. конф. «Лазеры. Измерения. Информация». Т. 2. – СПб.: СПбГПУ, 2009. – С. 202–212.
3. Чмутин А. М., Шарипов Р. Р. Исследование целостности цифровых фотоизображений методом фурье-анализа // Тр. 17 Междунар. конф. «Лазерно-информационные технологии в медицине, биологии и геоэкологии». – Новороссийск: КубГУ, 2009. – С. 107.
4. Рвачева О. В., Чмутин А. М., Шарипов Р. Р. Информационные технологии в учебном процессе специальности «Судебная экспертиза» // Матер. Междунар. науч. конф. «Электронная культура. Информационные технологии будущего и современное электронное обучение (Modern IT & (E-) Learning)». – Астрахань: НОВА, 2009. – С. 306–310.
5. [HTTP://www.colorpilot.ru/exifpro/artile.html](http://www.colorpilot.ru/exifpro/artile.html).
6. Вентцель Е. С. Теория вероятностей. – М.: Наука, 1969. – 576 с.
7. Гребенюк П. Е., Зотов П. В., Чмутин А. М., Шарипов Р. Р. Прикладная программа PNG-Reader, реализующая числовой анализ цифровых фотографических изображений для выявления фактов их редактирования в программном пакете PHOTOSHOP: Акт о внедрении программной разработки от 20.08.2010.

УДК 004.021

*В. А. Камаев, П. П. Кудряшов, А. А. Стец***ИСПОЛЬЗОВАНИЕ КОНТУРНОГО АНАЛИЗА В ЗАДАЧЕ КОНТРОЛЯ КАЧЕСТВА АБРАЗИВНЫХ МАТЕРИАЛОВ****Волгоградский государственный технический университет**

kamaev@unix.cad.vstu.ru, pal_ok@mail.ru, st-sany@mail.ru

В статье описывается автоматический метод контроля качества абразивных материалов. Метод основан на специализированных алгоритмах анализа изображения абразивной поверхности. На основании предложенных алгоритмов, была разработана система автоматического анализа, которая позволяет производить контроль зернистости и качества абразивных поверхностей.

Ключевые слова: образивы, зернистость, контроль качества, контурный анализ.

*V.A. Kamaev, P.P. Kudryashov, A.A. Stets***USE OF THE PLANIMETRIC ANALYSIS IN THE PROBLEM OF QUALITY ASSURANCE OF ABRASIVE MATERIALS****Volgograd state technical university**

kamaev@unix.cad.vstu.ru, pal_ok@mail.ru, st-sany@mail.ru

In the article the automatic monitoring of quality of abrasive materials is described. The method is based on specialized analysis algorithms of the image of an abrasive surface. The system of the automatic analysis which allows to make control of granularity and quality of abrasive surfaces has been developed on the basis of the offered algorithms.

Key words: abrasives, granularity, quality assurance, the planimetric analysis.

Развитие вычислительной техники и науки способствовало автоматизации не только рутинной человеческой деятельности, но и выполнения задач, которые раньше не могли успешно решаться без участия человека. Одной из таких задач является задача распознавание образов.

Практически в любом производстве одним из этапов процесса является шлифовка или доводка поверхностей. Этот этап связан с использованием абразивных материалов, из которых производятся разнообразные абразивные изделия: наждачная бумага и ткань, шлифовальные и правочные бруски, керамические и вулканитовые головки и другое [7].

Ввиду того, качество абразивных материалов влияет на конечный результат выпускаемых производством изделий, контроль процесса изготовления и качества получаемых абразивных материалов является крайне важным для современных производств.

В настоящее время в Российской Федерации существует государственный стандарт, согласно которому производится анализ качества абразивных материалов [9]. Для оценки зернистости применяют исключительно ручные методы, которые являются трудоемкими, требуют больших затрат времени и не обладают необходимой точностью. Помимо этого, данные методы позволяют исследовать абразивные зерна только в несвязанном состоянии, при этом оце-

нить качество зерен, нанесенных на подложку, они не могут.

Таким образом, задача автоматизации оценки абразивной поверхности является актуальной.

Абразив представляет собой зерна, классифицированные на узкие размерные фракции, которые используются для полировки, дальнейшего измельчения, изготовления жесткого и эластичного абразивного инструмента [8].

Зернистость – характеристика, отражающая величины зерен. Зерна делятся на фракции. Фракция – совокупность зерен абразива, размер которых лежит в заданной области. Абразив характеризуется природой материала, его кристаллической структурой, твердостью и размером зерен. Все это определяет режущую способность абразива, а зернистость, кроме того, определяет достижимую шероховатость обрабатываемой поверхности [8].

На рис. 1 представлено изображение образца шлифзерна, полученное под микроскопом.

Для определения качества абразива нами предлагается автоматический анализ изображения его поверхности с использованием методов распознавания образов.

Автоматизированный анализ зернистости состоит из трех последовательных этапов:

этап предварительной подготовки исходного изображения;

этап поиска контуров;

этап обработки данных.

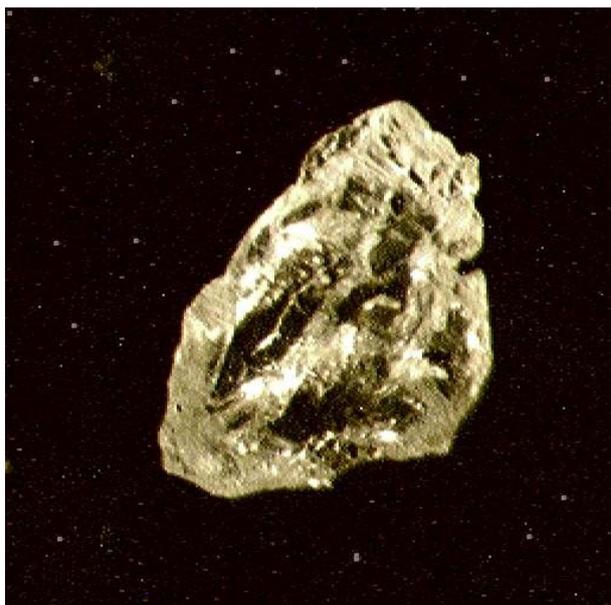


Рис. 1. Образец шлифзерна

Рассмотрим этапы анализа более подробно.

На рис. 1 можно видеть, что на абразивной поверхности помимо зерен присутствуют разнообразные нежелательные включения: структура подложки, пыль и т. п. Для повышения качества распознавания на первом этапе мы производим подавление подобных артефактов. Для этого к исходному изображению применяется ряд морфологических операций, что позволяет убрать нежелательные включения. Операции включают в себя последовательность эрозии и дилатации.

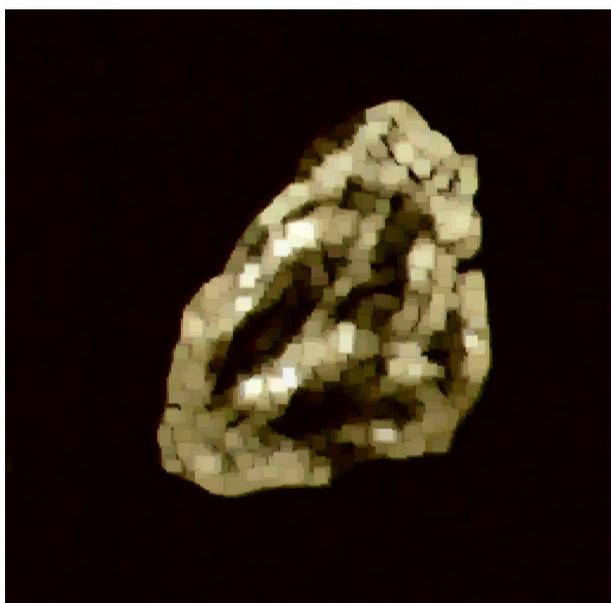


Рис. 2. Изображение после выполнения операции эрозия

После выполнения этих операций изображение переводится в монохромный вид. Для этого задается некоторый пороговый уровень $threshold$ [1].

$$M_{xy} = \begin{cases} 1, & img(x, y) > threshold \\ 0 & \end{cases} \quad (1)$$

Пиксеты, превосходящие заданный порог получают значение 1, остальные пиксеты – 0.

На втором этапе, этапе поиска контуров определяются все замкнутые не пересекающиеся контуры, обводящие локальные скопления нечерных (значение яркости > 0) пикселей [2]. Существует множество алгоритмов определения контуров на монохромном изображении с различной степенью оптимизации под характеристики изображения. Предлагается использование простейшего алгоритма, строящего вложенное дерево контуров, которое используется для последующего анализа. Данный алгоритм реализован в виде функций в библиотеке компьютерного зрения Intel OpenCV [2]. Эта библиотека активно применялась при реализации программных модулей.

Следующий этап – этап обработки данных. На этом этапе исключаются все вложенные контуры, которые обводят подобласти зерна различной яркости. А оставшиеся контуры, соответствующие отдельным зернам, аппроксимируются эллипсами (см. рис. 5). Далее происходит фильтрация эллипсов по специальным условиям. Так отбрасываются слишком мелкие и вытянутые эллипсы, которые, скорее всего, являются помехами на изображении, оставшимися после этапа шумоподавления.

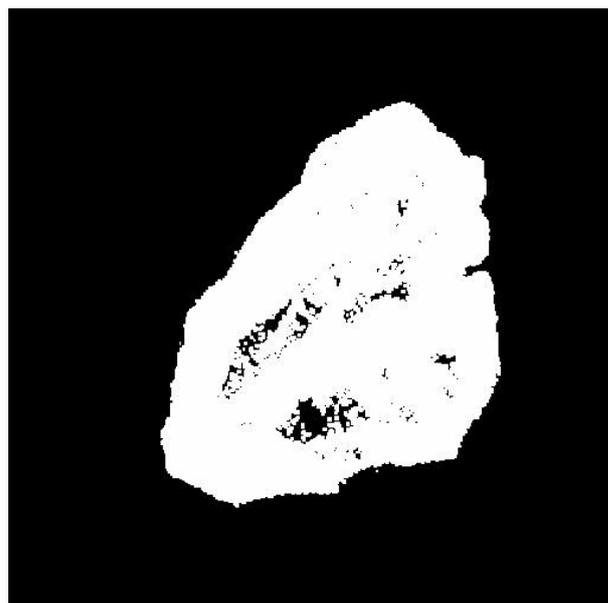


Рис. 3. Монохромное изображение

Слишком крупные эллипсы проверяются на случай склеенных зерен (см. рис. 6, 7). Для этого аппроксимируется площадь зерна, берется отношение площади зерна к площади эллипса. Если такое отношение меньше или равно заданной константе C , то это говорит о случае склеенных зерен. Эмпирически было установлено значение константы C , равное 0,96, которое позволяет с высокой степенью вероятности идентифицировать склеенные зерна. К прямоугольной области, содержащей склеенные зерна, снова применяются морфологические операции эрозии и дилатации с большим размером структурирующего элемента, что позволяет разрушить грани-

цу между зернами, т. е. отделить их друг от друга. Далее на этом участке снова производится поиск контуров. Поиск склеенных зерен производится рекурсивно до тех пор, пока анализируемые участки не станут слишком малы. По завершении процесса, эллипсы, обведившие склеенные зерна, удаляются, вместо них вставляются более точные эллипсы, каждый из которых обводит отдельное зерно.

Геометрические параметры полученных эллипсов (длины их осей) в дальнейшем могут использоваться для составления заключения о качестве исследуемого образца абразивной поверхности.

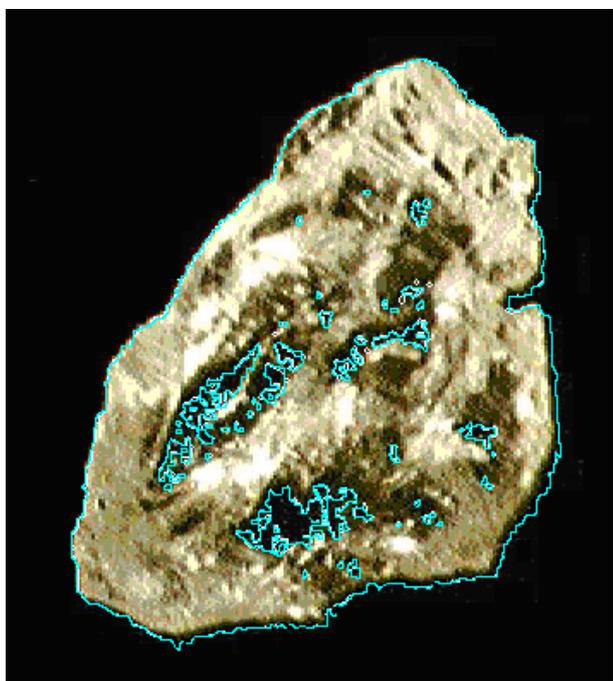


Рис. 4. Результат этапа поиска контуров

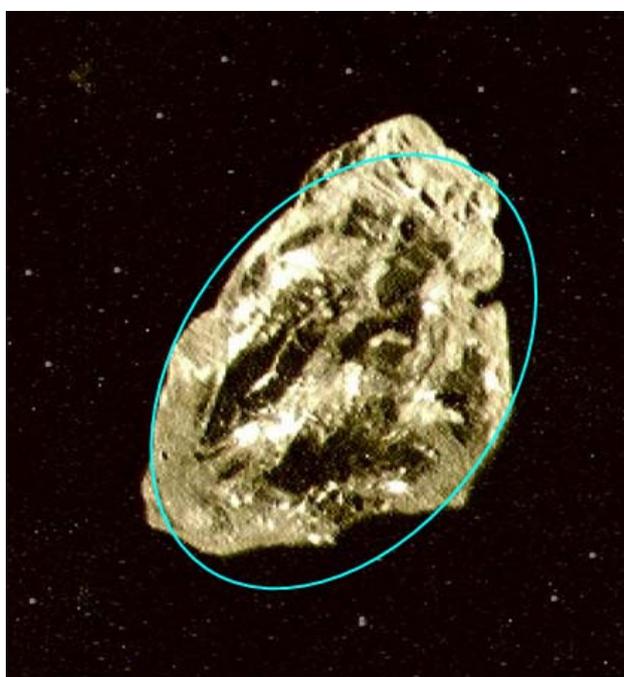


Рис. 5. Зерно, вписанное в эллипс

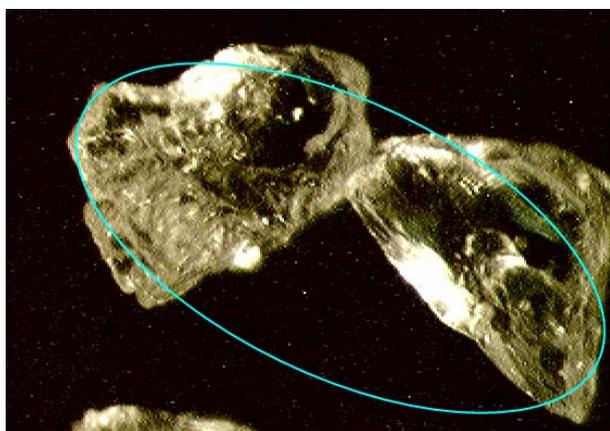


Рис. 6. Случай склеенных зерен до обработки

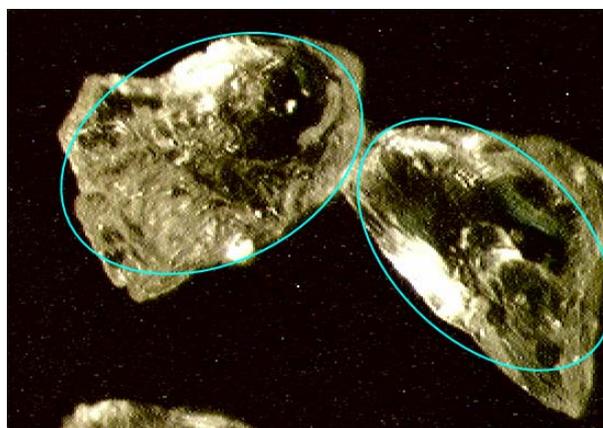


Рис. 7. Случай склеенных зерен после обработки

Предложенный подход был реализован в виде программной системы автоматического анализа качества абразивных материалов. Для реализации была использована среда разработки Visual Studio 2008, а также библиотека компьютерного зрения Intel OpenCV.

Для проведения испытаний системы автоматизированного анализа зернистости были использованы изображения абразивных поверхностей, полученных в реальных условиях на Волжском абразивном заводе. В среднем каждое изображение содержало около 100 абразивных зерен. Зерна представляли собой шлифзерно, зернистостью 50-П. При зернистости 50-П ведущая фракция составляет 500 мкм. В результате анализа зерна, составляющие ведущую фракцию, содержались в количестве 55 % от общего числа. Зерна комплексной фракции содержались в количестве 95 %. Такие параметры соответствуют ГОСТ 3647–80. «Материалы шлифовальные. Классификация. Зернистость и зерновой состав. Методы контроля».

Согласно ГОСТ 3647–80, исследуемый образец является качественным продуктом.

Однако в нескольких изображениях исследуемых образцов были обнаружены зерна зернистостями 850 и 880, которые являются слишком крупными для данной фракции, та как согласно ГОСТ 3647–80, для шлифзерна, ведущая фракция которого 50-П, максимальная зернистость составляет 800.

Таким образом, около 2 % зерен в исследуемых образцах превосходят предельную фракцию. То есть можно сделать вывод, что данная абразивная поверхность не отвечает требуемым характеристикам качества.

В целом было рассмотрено порядка 100 образцов абразивной поверхности. Несмотря на то, что ряд изображений полностью соответствовали ГОСТ, на 5 % из них было выявлено несоответствие ГОСТ. Данный пример показывает, что ручной анализ партии, проводимый на нескольких образцах не способен с должностью точностью определить возможные дефекты. С использованием предложенной системы становится возможным автоматический анализ целых партий абразивных поверхностей как на этапе производства абразивных материалов, так и на этапе анализа качества закупленной партии абразивных материалов в автомобильной, деревообрабатывающей, мебельной промышленности и др.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Петров, В. О. Алгоритм текстурной сегментации растровых изображений при решении прикладных задач медико-биологического анализа / В. О. Петров, В. А. Камаев, С. В. Поройский // Современные проблемы науки и образования. – 2009. – № 6. – С. 106–110.
2. Bradski G., Kaebler A. O'Reilly-LearningOpenCV, 2008. – 577 с.
3. Кудряшов П. П. Гибридный алгоритм обнаружения человеческих лиц / П. П. Кудряшов, С. А. Фоменков // Информационные технологии. – 2007. – № 10. – С. 20–23.
4. Стокман Д., Шапиро Л. Компьютерное зрение. – Бином. Лаборатория знаний, 2006 г. – 752 с.
5. Дэвид А. Компьютерное зрение. Современный подход. 2004. – 928 с.
6. Малинин В. В. Распознавание образов на ЭВМ. М: Учеба, 2005. – 573 с.
7. Буткарев А. И. Полировка. «АБ Универсал». Технологии, материалы, оборудование, инструменты, 2002.
8. Гаршин А. П. Абразивные материалы. Л., 1983.
9. ГОСТ 3647–80. Материалы шлифовальные. Классификация. Зернистость и зерновой состав. Методы контроля. – М, 1982.

УДК 044.9

*А. О. Матлин, С. А. Фоменков***ПОСТРОЕНИЕ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ
СОЗДАНИЯ ИНТЕРАКТИВНЫХ ТРЕНАЖЕРОВ****Волгоградский государственный технический университет**

alexander.o.matlin@gmail.com, saf@vstu.ru

На примере виртуального интерактивного тренажера для студентов медицинских специальностей рассматривается средство автоматизированного создания интерактивных средств обучения, и выделяются общие задачи автоматизации построения интерактивных лабораторных работ.

Ключевые слова: виртуальные тренажеры, информационная система, интерактивные средства обучения.

*A. O. Matlin, S. A. Fomenkov***ARCHITECTURE OF THE AUTOMATED SYSTEM
FOR INTERACTIVE LEARNING TOOLS DEVELOPMENT****Volgograd State Technical University**

On the example of a virtual interactive simulator for medical students is considered a means of automated creation of interactive learning tools and general automation tasks allocated for constructing interactive labs.

Key words: virtual simulators, information system, interactive learning tools.

Качественное образование предполагает получение студентами не только теоретических знаний, но и получение ими практических навыков. Для этого нами предлагается использование в образовательном процессе интерактивных средств обучения, позволяющих с помощью автоматизированной информационной системы приобретать студентам практические навыки.

Актуальность данного направления работ особенно остро видна на примере студентов медицинских вузов. В медицинских вузах получение студентами практических навыков в области хирургии затруднено тем, что студент не имеет возможности отрабатывать те или иные хирургические практики на пациентах клинических отделений.

Интерактивные виртуальные тренажеры предназначены для проведения виртуальных лабораторных работ. Целью Интерактивных виртуальных тренажеров является приобретение учащимися практических навыков для последующей работы с реальными пациентами.

Разработанная автоматизированная система (далее – АС) предназначена для проведения виртуальных лабораторных работ в учебных заведениях медицинского направления с целью повышения уровня практических навыков студентов и отработки порядка действий при выполнении хирургических доступов. АС также может применяться студентами лично во внеучебное время для повышения уровня своих профессиональных знаний и подготовке к прак-

тическим занятиям в учебном заведении. Еще одним направлением использования АС является ее использование хирургами в рамках подготовки к операции.

АС позволяет создавать широкий набор различных хирургических доступов. Хирургические доступы состоят более чем из 20 шагов. Для получения положительной оценки по результатам выполнения лабораторной работы, пользователю необходимо пройти все шаги от начала и до конца за отведенный промежуток, набрав не менее 61 балла. Шаг в рамках лабораторной работы определен как некоторая последовательность действий, которую пользователь должен выполнить, чтобы можно было приступить к следующему шагу. Графически шаг представляет собой набор графических файлов, наложенных друг на друга таким образом, чтобы получилось комплексное изображение человека на определенной стадии операции. Для выполнения действий пользователь может выбирать различные инструменты – их список постоянен в течение всей лабораторной работы. Каждый инструмент для определенного шага имеет свое графическое представление. После выбора инструмента, можно проводить им различные действия, направленные на осуществление хирургического доступа.

В АС происходит автоматизированная обработка ошибок пользователя, за которые снимаются баллы. В рамках лабораторной работы предусмотрены два вида возможных ошибок пользователя:

1. Выбран неправильный инструмент;
2. Действие инструментом произведено вне границ допустимой области.

Первый тип ошибки возникает, когда была произведена попытка действия инструментом, который не является допустимым на текущем шаге. Второй тип ошибки возникает в том случае, если инструмент был выбран правильно, однако действия им были осуществлены вне пределов допустимой области. При появлении каждой ошибки, возникает окно, в котором указано описание ошибки, совершенной пользователем, и количество штрафных баллов за совершенную ошибку.

Для реализации АС были использованы современные технологии разработки программного обеспечения Microsoft. Основной используемой технологией, лежащей в основе АС, является Windows Presentation Foundation [1], представляющая собой высокоуровневый объектно-ориентированный функциональный слой, позволяющий создавать 2D- и 3D-интерфейсы.

В качестве визуальной среды создания пользовательского интерфейса был использован продукт Microsoft Expression Blend 4 [2]. Разработка АС велась в среде разработки Microsoft Visual Studio 2010 Team System [3].

АС поддерживает сетевой вариант ее использования, в рамках которого информация с рабочих станций в учебных аудиториях поступает на сервер и преподаватель в режиме реального времени имеет возможность следить за успеваемостью как каждого студента в отдельности, так и потока в целом. В качестве хранилища данных используется Microsoft SQL Server 2005, обеспечивающий производительность при доступе к данным и надежное их хранение.

В настоящее время АС внедрена в учебный процесс Волгоградского государственного медицинского университета. Использование Интерактивных виртуальных тренажеров позволило студентам Волгоградского государственного медицинского университета приобрести практические навыки, которое они не могли получить ранее во время учебы в вузе, что, несомненно, привело к повышению качества образования в университете.

В результате создания данной АС были выявлены общие для построения виртуальных интерактивных средств обучения подходы и инструменты. В результате чего был создан прототип автоматизированного средства создания интерактивных средств обучения (далее – Средство создания).

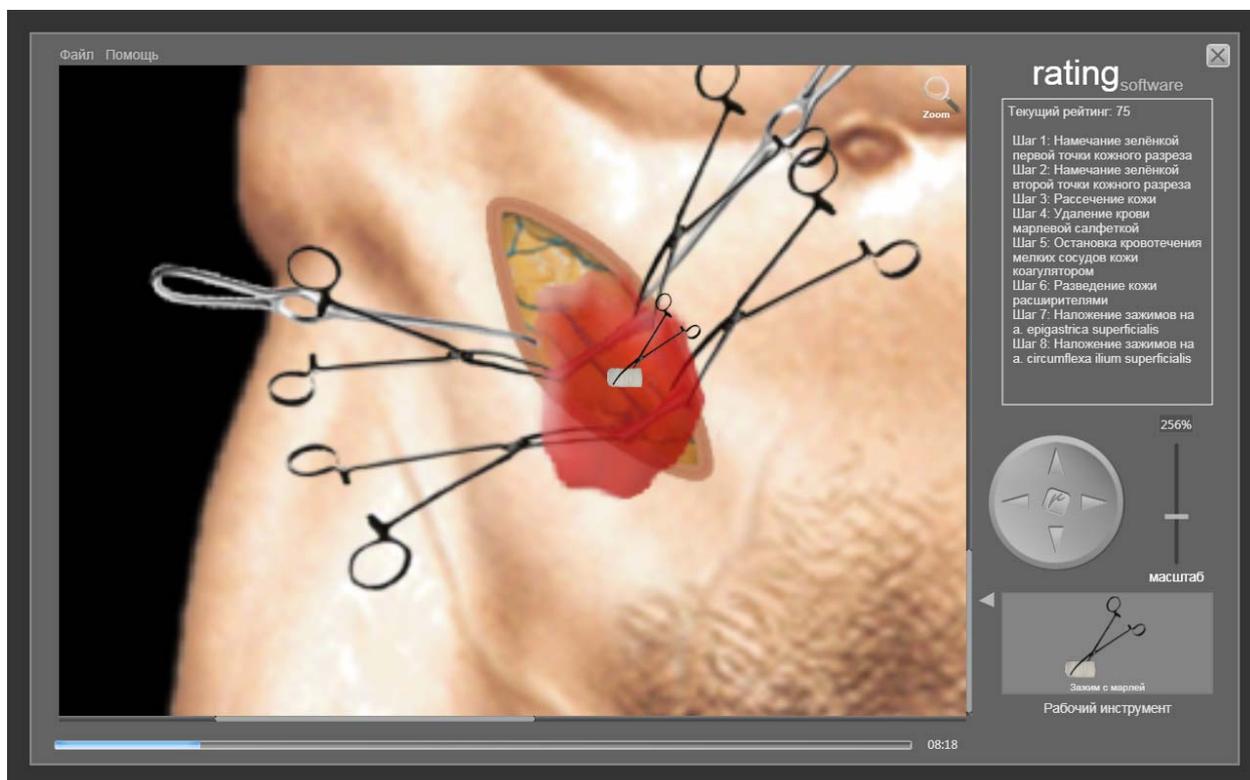


Рис. 1. Использование инструмента «Ватка»

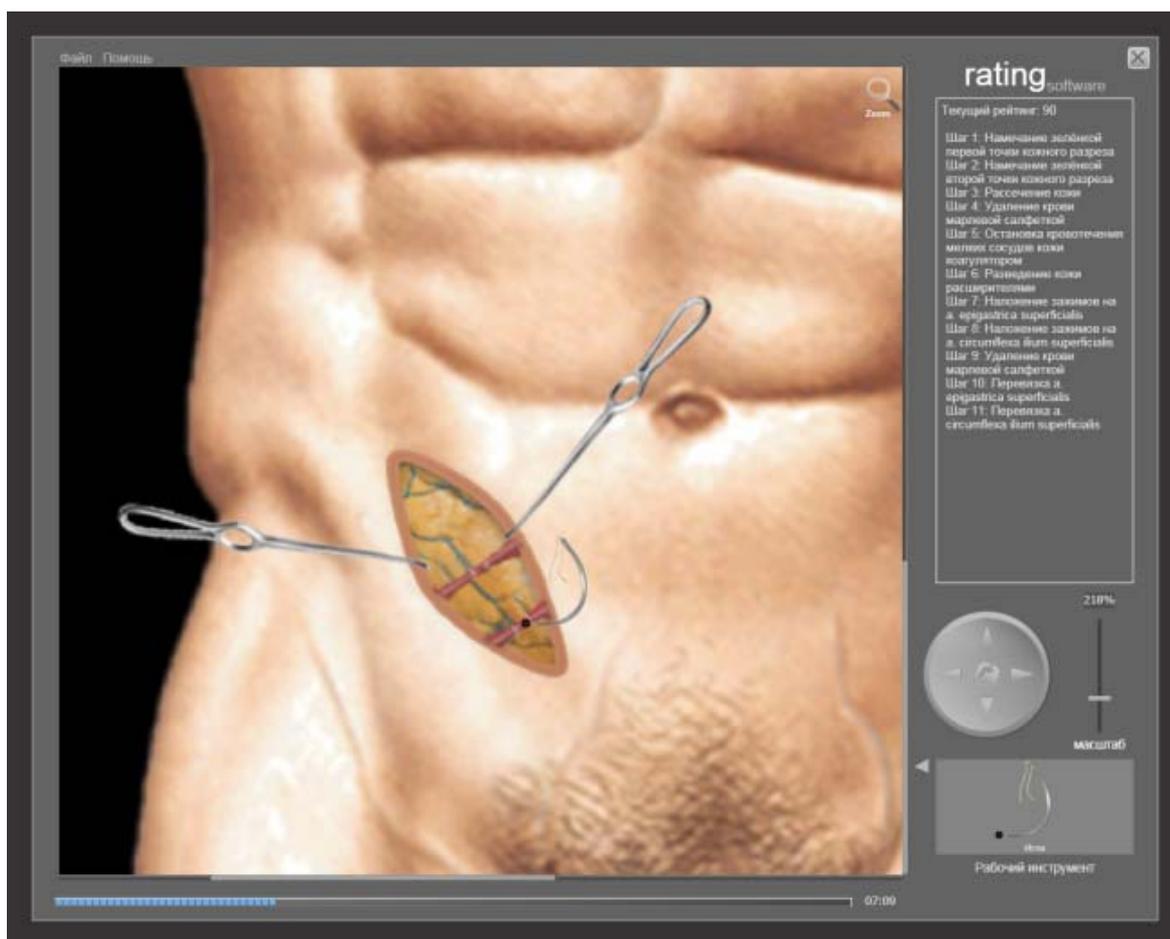


Рис. 2. Использование инструмента «Нить»

Данное Средство создания позволяет решать следующие задачи автоматизации в построении интерактивных тренажеров:

1. Создавать последовательность шагов лабораторной работы (набор связанных между собой графических изображений);

2. Указывать активные области на графических изображениях, в которых пользователь будет выполнять манипуляции с инструментарием лабораторной работы во время ее проведения;

3. Устанавливать набор инструментов, с помощью которых пользователь будет иметь возможность выполнять задания лабораторной работы в активных областях графических изображений;

4. Устанавливать соответствия между активными областями графических изображений и инструментами для проверки корректности выбора инструментария пользователем;

5. Определять набор ошибок, которые могут быть совершены пользователем и которые не-

обходимо отслеживать АС во время выполнения лабораторной работы;

6. Создавать готовую к использованию лабораторную работу в виде исполняемого файла.

Развитие данной работы заключается в совершенствовании Средства создания интерактивных средств обучения за счет увеличения автоматизируемых шагов создания интерактивной виртуальной лабораторной работы и повышению удобства его использования.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Windows Presentation Foundation [Электронный ресурс]. – 2010. – Режим доступа: <http://msdn.microsoft.com/en-us/library/ms754130.aspx>
2. Microsoft Expression Studio [Электронный ресурс]. – 2010. – Режим доступа: <http://www.microsoft.com/expression/>
3. Среда разработки Visual Studio [Электронный ресурс]. – 2010. – Режим доступа: <http://msdn.microsoft.com/ru-ru/vstudio/default.aspx>

УДК 002.6:004.91

*Ю. А. Орлова***РАСШИРЕННАЯ НЕЧЕТКАЯ АТРИБУТНАЯ ГРАММАТИКА
НАД ФРЕЙМОВОЙ СТРУКТУРОЙ ТЕКСТА ТЕХНИЧЕСКОГО ЗАДАНИЯ*****Волгоградский государственный технический университет**
yulia.orlova@gmail.com

В работе рассмотрена расширенная нечеткая атрибутивная грамматика над фреймовой структурой технического задания, которая позволяет автоматизировать процесс создания моделей программного обеспечения, описанного в техническом задании.

Ключевые слова: нечеткая атрибутивная грамматика, фреймовая структура, анализ текста, модели программного обеспечения.

*Y. A. Orlova***EXTENDED FUZZY ATTRIBUTE GRAMMAR ABOVE FRAME STRUCTURE
OF TEXT OF A TECHNICAL SPECIFICATION****Volgograd State Technical University**

In work the extended fuzzy attribute grammar above frame structure of text of a technical specification which allows to automate process of construction models of software based on technical specification is considered.

Key words: fuzzy attribute grammar, frame structure, text analysis, software models.

Разработка и анализ технической документации требует от лиц, занимающихся проектированием программного обеспечения семантической обработки большого объема технического текста, глубокого знания предметной области и навыков в проектировании. Трудоемкость процесса анализа текста приводит к необходимости его автоматизирования. Однако необычайная сложность проблемы синтеза и анализа семантики технического текста, для решения которой необходимо использовать симбиоз методов искусственного интеллекта, прикладной лингвистики, психологии и т. п., приводит к тому, что она до сих пор не решена. В данной работе мы пытаемся автоматизировать начальный этап проектирования программного обеспечения – семантический анализ текста технического задания.

Для семантического анализа текста технического задания необходимо создание унифицированной структуры текста технического задания, то есть создание такой грамматики, которая позволит наиболее полно отобразить содержимое технического задания.

Проблема состоит в том, что часть компонентов ТЗ содержит информацию, которая по своему характеру является нечеткой, что обусловлено вариантностью и подвижностью границ языковой нормы и статистическим характером отдельных видов информации. Неточность информации, содержащейся в компонентах ТЗ, относится к семантическому и предметно-

зависимому уровню ТЗ и обусловлена сложностью процесса формализации описываемых явлений. Рекомендации по проведению такой формализации формулируются в виде описаний на ЕЯ, апеллирующих к языковой интуиции человека, и могут трактоваться по-разному различными специалистами.

Задачей исследования является автоматизированная обработка текста документа, который изначально создается человеком на естественном языке с соблюдением примерной структуры, требования к которой изложены в ГОСТах.

На данном этапе развития методов анализа текста не представляется возможным анализировать естественный язык без каких-либо ограничений, поэтому необходимо выявить особенности существующей практики написания технических заданий и сформулировать дополнительные требования, которым должно удовлетворять техническое задание, чтобы оно могло быть проанализировано с помощью предлагаемой в данной работе системы.

Предлагаемая методика анализа текста технического задания содержит формализмы, необходимые для представления семантики требований к программному обеспечению на ранних этапах проектирования. В соответствии с предлагаемой методикой система рассматривается как черный ящик, а предъявляемые к ней требования представляются в виде спецификации функций и определения потоков входных и выходных воздействий. Методика анализа текста

* Работа выполнена при поддержке грантов РФФИ 10-01-00135-а, 10-01-00165-а, 10-01-90012-Бел_а.

технического задания состоит из трех этапов: семантическая обработка текста, создание фреймовой структуры и создание диаграмм потоков данных системы, описанной в техническом задании.

Для реализации первого этапа методики была разработана семантическая модель текста технического задания, включающая требования, сформулированные в виде документа на ограниченном естественном языке; второго этапа – фреймовая структура, являющаяся внутренним представлением требований; третьего этапа – модель программного обеспечения в виде описания требований на графическом языке Data Flow Diagrams.

Семантическая модель текста технического задания содержит разработанную расширенную нечеткую атрибутивную грамматику над фреймовой структурой формального документа «Техническое задание», которая позволяет наиболее полно отобразить содержимое ТЗ.

Расширенная нечеткая атрибутивная грамматика, необходимая для автоматизированного анализа текста технического задания, определена в виде:

$$AG = \langle N, T, P, S, B, F, A, R(A) \rangle,$$

где N – конечное множество нетерминальных символов; T – непересекающееся с N множество терминальных символов; P – конечное множество правил; S – выделенный символ из N , называемый начальным символом; B – множество лингвистических переменных $\beta_{k,i}$, соответствующих терминальным символам T (переменная i на k уровне); F – множество функций принадлежности $f_{k,i}$, определяющих степень принадлежности лингвистических переменных $\beta_{k,i}$; A – конечное множество атрибутов, $A = A_{\text{син}} \cup A_{\text{сем}}$, где $A_{\text{син}}$ – синтаксические атрибуты, $A_{\text{сем}}$ – семантические атрибуты; $R(A)$ – конечное множество семантических действий.

Лингвистические переменные из множества $B = \{\beta_{k,i}\}_{k,i}$, используемые для анализа текста технического задания, описываются пятеркой:

$$\beta_{k,i} = \langle \beta, T(\beta), U, G, M \rangle,$$

β – название лингвистической переменной (наименование и область применения, основание для разработки, назначение разработки, технические требования к программному изделию, стадии и этапы разработки и т. д.);

$T(\beta)$ – языковые выражения. Для лингвистических переменных верхнего уровня они являются лингвистическими переменными, соответствующими терминалам правой части пра-

вила. Для лингвистических переменных нижнего уровня – нечеткими переменными, то есть выражениями естественного языка.

U – универсум, $T(\beta) \subset U$;

G – правила морфологического и синтаксического описания языковых выражений, которые определяют синтаксические атрибуты $A_{\text{син}}$.

Язык представления выражений состоит из констант и правил их последовательного применения. На морфологическом уровне константами являются граммы (рд – родительный падеж, мн – множественное число). На синтаксическом – названия отношений и групп (ПОДЛ – отношение между подлежащим и сказуемым, ПГ – предложная группа). Для каждого слова входного текста выдается множество морфологических интерпретаций следующего вида: лемма; морфологическая часть речи; набор общих граммем; множество наборов граммем.

Используются следующие синтаксические правила образования текста: деепричастный, причастный, вводный оборот; необособленное согласованное определение в препозиции; количественная группа; последовательность чисел вперемешку со знаками препинания; фрагмент с личной формой глагола, с кратким причастием, с кратким прилагательным, с предикативом, с инфинитивом, с тире, со сравнительным прилагательным; существительное + числовой идентификатор; правила для построения ФИО; слова степени с группой прилагательного или причастия; однородные прилагательные, наречия, инфинитивы, прилагательные сравнительной степени; группы даты, временных отрезков; аналитическая форма сравнительной степени прил. или наречия; наречие + глагол; одно или несколько прилагательных, согласованных по роду, числу и падежу с существительным и другие.

M – семантическое правило для лингвистических переменных, которое индуцируется морфологическими и синтаксическими правилами, так как смысл термина в T частично определяется его синтаксическим деревом, и семантическими атрибутами $A_{\text{сем}}$.

Методы представления связей между правилами транслируются на язык нечеткой математики. При этом связи представляются нечеткими отношениями, предикатами и правилами, а последовательность преобразований этих отношений – как процесс нечеткого вывода.

Лингвистические переменные верхнего уровня являются составными, то есть включа-

ют лингвистические переменные нижнего уровня. Благодаря этому можно построить дерево лингвистических переменных и установить зависимость между ними.

Функции принадлежности из множества $F = \{f_{k,i}\}_{k,i}$ лингвистических переменных $\{\beta_{k,i}\}_{k,i}$, необходимы для построения нечеткого вывода. В частности, каждому правилу грамматики из множества P ставится в соответствие функция принадлежности $f_{k,i}$. Эта двойственная система подстановок используется для вычисления смысла лингвистической переменной.

Синтаксические атрибуты A_{sin} , используемые в грамматике: 'Название' – текст представляет собой наименование раздела; 'Содержимое' – текст представляет собой содержимое раздела; 'Клауза' – клауза; 'Клауза ТИРЕ'-Фрагмент с тире; 'Группа ГЕНИТ_ИГ' – именительная группа, связанная родительным падежом и др.

Семантические атрибуты, используемые в грамматике, содержат название атрибута A_{sem} и семантическое действие $R(A)$: "Фрейм СИСТЕМА = Создание" – создается фрейм СИСТЕМА; "Слот НАЗВАНИЕ СИСТЕМЫ = Присваивание" – значение присваивается слоту НАЗВАНИЕ СИСТЕМЫ; "Фрейм ПОТОК ДАННЫХ = Создание" – создается фрейм ПОТОК ДАННЫХ; "Слот ВХОД = Присваивание", "Слот ВЫХОД = Присваивание" – значение присваивается слотам ВХОД, ВЫХОД; "Слот КОЛ-ВО ДАННЫХ = Присваивание" – значение присваивается слоту КОЛИЧЕСТВО ДАННЫХ; "Слот ТИП ДАННЫХ = Присваивание" – значение присваивается слоту ТИП ДАННЫХ; "Слот НАЗВАНИЕ ПОТОКА ДАННЫХ = Присваивание" – значение присваивается слоту НАЗВАНИЕ ПОТОКА ДАННЫХ; "Фрейм ФУНКЦИЯ = Создание" "Слот НАЗВАНИЕ ФУНКЦИИ = Присваивание" – значение присваивается слоту НАЗВАНИЕ ФУНКЦИИ и др.

Семантический анализ текста производится на основе разработанной грамматики текста ТЗ:

1. Каждая лингвистическая переменная технического задания подвергается разбору, в результате чего получается лингвистическое дерево, конечными вершинами которого являются нечеткие переменные.

2. Нечетким переменным на конечных вершинах дерева назначается их смысл и затем с помощью системы правил P и соответствующих функций принадлежности $f_{k,i}$ вычисляется

смысл лингвистической переменной, соответствующей левой части правила.

Продукционные правила P верхнего уровня служат для разбора разделов верхнего уровня. Правила для разбора разделов состоят из двух частей: первая часть служит для разбора названия раздела; вторая часть служит для разбора текстового содержимого раздела.

Для некоторой лингвистической переменной $\beta_{k,i}$ значение функции принадлежности: $\mu_{k,i} = f_{k,i}(\mu_{k+1,1}, \mu_{k+1,2}, \dots, \mu_{k+1,n})$, где конкретное значение $\mu_{k,i}$ – степень принадлежности лингвистической переменной $\beta_{k,i}$. Первоначально будем говорить, что все лингвистические переменные нижнего уровня несут одинаковый вклад в значение функции принадлежности, поэтому можно говорить, что функция принадлежности лингвистической переменной $\beta_{k,i}$:

$$f_{k,i}(\{\mu_{k+1,j}\}) = q_{k+1,i} * \sum_{j=1}^n \mu_{k+1,j}$$

где $\mu_{k+1,j}$ – степень принадлежности лингвистической переменной $\beta_{k+1,j}$; $q_{k+1,i} = 1/n$ – вклад степеней принадлежности в значение функции. На нижнем уровне k функции принадлежности определены.

Вычисленная $\mu_{k,i}$ сравнивается с μ_i , являющейся предельным значением степени принадлежности. Если $\mu_{k,i} > \mu_i$, и в правилах указаны синтаксические или семантические атрибуты, то создаются фреймы и слоты, в которые помещается текст из соответствующей лингвистической переменной.

3. После этого дерево урезают так, чтобы вычисленные лингвистические переменные оказались конечными вершинами оставшегося поддерева.

Этот процесс повторяется до тех пор, пока не будет вычислен смысл лингвистической переменной, соответствующей корню исходного дерева. Основное назначение описанной процедуры состоит в том, чтобы связать смысл лингвистической переменной со смыслом составляющих ее нечетких переменных посредством грамматики.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Заболеева-Зотова, А. В.* Автоматизация начальных этапов проектирования программного обеспечения / А. В. Заболеева-Зотова, Ю. А. Орлова // Изв. ВолгГТУ. Серия «Актуальные проблемы управления, вычислительной техники и информатики в технических системах»: межвуз. сб. науч. ст. / ВолгГТУ. – Волгоград, 2010. – Вып. 8, № 6(66). – С. 51–55.

2. *Орлова, Ю. А.* Алгоритмическое обеспечение анализа текста технического задания и построения моделей

программного обеспечения / Ю. А. Орлова // Изв. ВолгГТУ. Серия «Актуальные проблемы управления, вычислительной техники и информатики в технических системах»: межвуз. сб. науч. ст. / ВолгГТУ. – Волгоград, 2010. – Вып. 8, № 6(66). – С. 68-72.

3. *Заболеева-Зотова, А. В.* Автоматизация семантического анализа документации технического задания / А. В. Заболевеева-Зотова, Ю. А. Орлова // Вестник компьютерных и информационных технологий. – 2008. – № 9. – С. 26–34.

4. *Заболеева-Зотова, А. В.* Атрибутная грамматика формального документа «Техническое задание» / А. В. Заболевеева-Зотова, Ю. А. Орлова // Изв. ВолгГТУ. Серия «Актуальные проблемы управления, вычислительной техники и информатики в технических системах»: межвуз. сб. науч. ст. / ВолгГТУ. – Волгоград, 2008. – Вып. 4, № 2. – С. 39–43.

5. *Заболеева-Зотова, А. В.* Автоматизация процедур семантического анализа текста технического задания / А. В. Заболевеева-Зотова, Ю. А. Орлова // Известия ВолгГТУ. Серия «Актуальные проблемы управления, вычислительной техники и информатики в технических системах»:

межвуз. сб. науч. ст. / ВолгГТУ. – Волгоград, 2007. – Вып. 3, № 9. – С. 49–52.

6. *Заболеева-Зотова, А. В.* Моделирование лексического анализа текста технического задания / А. В. Заболевеева-Зотова, Ю. А. Орлова // Изв. ВолгГТУ. Серия «Актуальные проблемы управления, вычислительной техники и информатики в технических системах»: межвуз. сб. науч. ст. / ВолгГТУ. – Волгоград, 2007. – Вып. 2, № 2. – С. 39–42.

7. *Розалиев В. Л.* Предпосылки, возможности, перспективы создания автоматизированной системы распознавания эмоциональности речи / В. Л. Розалиев // Известия ВолгГТУ. Серия «Актуальные проблемы управления, вычислительной техники и информатики в технических системах»: межвуз. сб. науч. ст. / ВолгГТУ. – Волгоград, 2008. – Вып. 4, № 2. – С. 58–61.

8. *Розалиев В. Л.* Построение модели эмоций по речи человека / В. Л. Розалиев // Известия ВолгГТУ. Серия «Актуальные проблемы управления, вычислительной техники и информатики в технических системах»: межвуз. сб. науч. ст. / ВолгГТУ. – Волгоград, 2007. – Вып. 3, № 9. – С. 62–65.

УДК 004.81:159.942.52

В. Л. Розалиев, А. С. Бобков, О. С. Федоров

ПРИМЕНЕНИЕ НЕЙРОННЫХ СЕТЕЙ И ГРАНУЛЯЦИИ ПРИ ПОСТРОЕНИИ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ ОПРЕДЕЛЕНИЯ ЭМОЦИОНАЛЬНОЙ РЕАКЦИИ ЧЕЛОВЕКА*

Волгоградский государственный технический университет

vladimir.rozaliev@gmail.com, bobkovArt@gmail.com, fedorov.oleg.vlg@gmail.com

В работе рассмотрены модули автоматизированной системы определения эмоциональных реакций человека по телодвижениям и речи.

Ключевые слова: эмоциональная реакция, речь и телодвижения человека, временной ряд, лингвистическая переменная, грануляция, нейронная сеть.

V. L. Rosaliev, A. S. Bobkov, A. S. Fedorov

APPLICATION OF NEURAL NETWORKS AND GRANULATION UNDER CONSTRUCTION AUTOMATED SYSTEM FOR DETERMINING THE HUMAN EMOTIONAL REACTION

Volgograd State Technical University

The paper discusses the automated system modules determine the emotional reactions of human body movements and speech.

Key words: emotional response, speech and body movements of human, time series, linguistic variable, granulation, a neural network.

Каждую секунду человек осознано или не осознано испытывает определенные эмоциональные реакции. Безусловно, внешние проявления переживаний могут быть и не заметны, их возможно скрыть или же показать напускные эмоции. Однако, если очень внимательно проанализировать поведение человека можно выявить настоящие переживания. Работа, ведущаяся на кафедре «Системы автоматизированного проектирования и поискового конструирования» Волгоградского государственного технического университета, направлена на оп-

ределение эмоциональных реакций людей по речи и телодвижениям. В статье будет рассмотрена взаимосвязь двух модулей определения эмоциональных реакций и подробно рассмотрены механизмы работы модуля, направленного на анализ движений и поз человека.

На данном этапе работы реализована система определения эмоциональных реакций по речи человека при работе с системой автоматизации начальных этапов проектирования [2, 4, 5, 6, 7] и разрабатывается система определения эмоциональных реакций по движениям человека. [1, 3].

* Работа выполнена при поддержке грантов РФФИ 10-01-00135-а, 10-01-00165-а, 10-01-90012-Бел_а.

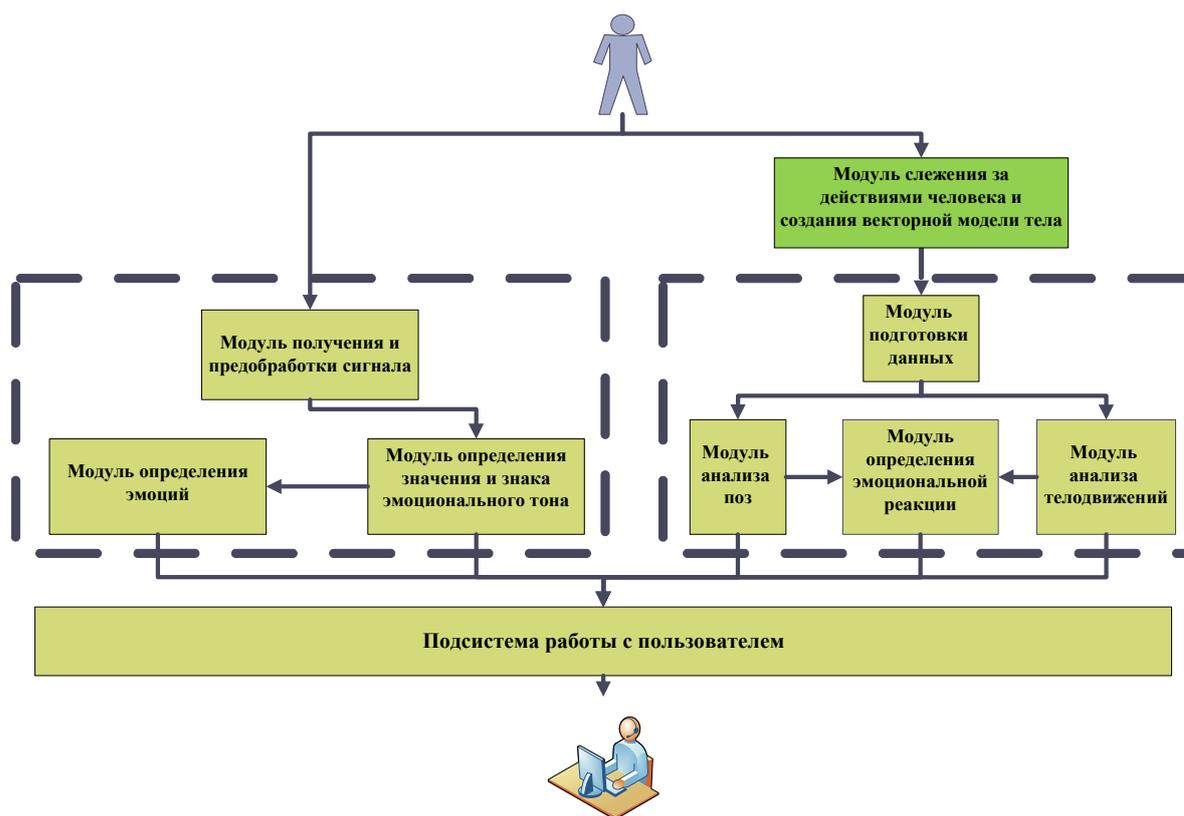


Рис. 1. Упрощенный вид архитектуры автоматизированной системы определения эмоциональных реакций

Рассмотрим разрабатываемую подсистему подробно. Входом в подсистему служит распознанное движение в формате bvh, а также данные необходимые для подготовки к эксплуатации. Эти данные включают в себя информацию о характерных позах и телодвижениях. Выход – информация о распознанных эмоциональных реакциях.

Первоначально распознанное движение поступает на вход подсистемы предобработки данных. Здесь данные фильтруются, выделяются блоки данных, которые описывают статические и динамические зоны тела. После подсистемы предобработки данных разделенные блоки отдельно обрабатываются соответствующими подсистемами: анализа поз и телодвижений. Информация, полученная после анализа, соединяется и формируется общий результат в подсистеме сопоставления результатов. Затем данные дополняются экспертными знаниями, которые хранятся в базах данных характерных поз и телодвижений и поступают на выход пользователю. Также в системе представлены два файловых хранилища, в которых располагаются файлы, необходимые для обучения подсистем анализа. На основе них формируются обучающие выборки.

Авторами разработан подход к разграничению тела человека. Чтобы повысить эффективность идентификации эмоционального состояния человека, информация о теле разбивается на две зоны: верхняя и нижняя (рис. 2). Верхняя зона содержит узлы тела, относящиеся к рукам, голове, шее, спине. Нижняя зона включает в себя узлы, относящиеся к ногам и тазу.

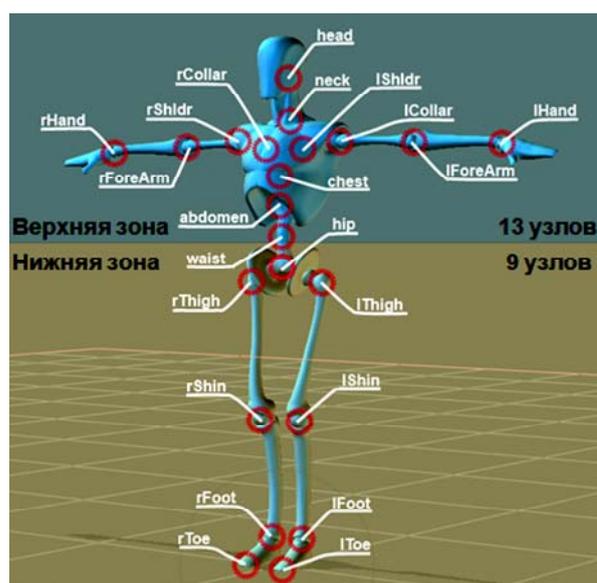


Рис. 2. Разграничение зон тела человека

Рассмотрим более детально процесс анализа динамических блоков. Авторами был выявлено, что информацию о телодвижениях человека, представленную в формате bvh можно представить в виде временных рядов. [1, 7] Для решения задачи анализа телодвижений был произведен обзор методов моделирования временных рядов. В результате был выбран нечеткий подход, поскольку он позволяет в полной мере учесть как специфику входной информации, так и трудности, связанные с ее обработкой.

Для анализа информации о телодвижениях необходимо формализовать активность телодвижений человека. Активность «А» выражается в количестве телодвижений человека: чем меньше телодвижений, и как следствие меньше изменений в каналах файла, тем значение активности «А» – меньше. Активность телодвижений человека в момент времени t определяется формулой:

$$A(t) = \sum_{n=1}^m (T_n(t) * k_n),$$

где m – количество временных рядов, характеризующих движение, $T_n(t)$ – изменение n -го временного ряда в момент времени t , k_n – коэффициент чувствительности для n -го временного ряда (ВР).

Этот коэффициент зависит от степени влияния n -го временного ряда на перемещение массы тела человека в пространстве. Активность зависит непосредственно от того, какой частью тела субъект совершил движение.

Изменение значения временного ряда (ВР) в момент времени t определяются формулой:

$$T_n(t) = |Z_n(t) - Z_n(t - \Delta t)|,$$

где $Z_n(t)$ – значение n -го ВР в момент времени t , Δt – величина интервала времени, в данном случае $\Delta t = 0,33$, это связано с ограничениями, накладываемыми технологией безмаркерного захвата телодвижений человека IriSoft (Shoot3D).

Коэффициент чувствительности, учитывающий распределение массы и длины тела человека будет равен:

$$k_n = \sum_{i=0}^j (p_{di} * m_{di}),$$

где i – номер дочернего элемента, j – максимальный номер дочернего элемента, m_{di} – коэффициент массы тела, показывает, какую долю от всей массы тела занимает данная

часть, p_{di} – коэффициент пропорциональности массы тела.

Для дальнейшей обработки информации о телодвижениях человека авторами была произведена фазсификация (трансформация исходного временного ряда в нечеткий временной ряд) временных рядов, осуществлены грануляции (введение лингвистических переменных) первого и второго уровней.

Введем лингвистическую переменную «Нечеткий угол»:

$$L_1 = \langle Name_1, \tilde{X}_1, X_1, G_1, P_1 \rangle$$

где $Name_1$ = «Нечеткий угол» – имя переменной, \tilde{X}_1 – множество термов лингвистической переменной (ЛП), X_1 – область определения, универсальное множество значений углов, G_1 – синтаксические правила вывода новых термов (Modus ponens), P_1 – семантические правила, контекстно-зависимый способ вычисления смысла на основе функции принадлежности каждого терма (соответствие термов ЛП точному значению ВР на основе функций принадлежности).

Множество термов ЛП «Нечеткий угол» в зависимости от узла, который описывается, будет состоять из различного количества элементов. Связано это, прежде всего, с подвижностью суставов. Так, например если разбить множество узлов, описывающих угол вращения локтевого сустава, максимальный поворот которого равен 154 градуса, на 9 элементов: $A0$ = «нулевой», $A1$ = «около $\pi/10$ », $A2$ = «около $\pi/5$ », ..., $A8$ = «около $9\pi/10$ », то для описание лучезапястного сустава, относительно оси, перпендикулярной поверхности ладони, максимальный поворот которого равен 55 градуса, такое количество элементов будет избыточно и ошибка при этом велика. Графики функций принадлежности к термам представлены на рис. 3.

После чего проводим гранулирование второго уровня, вводим НП «Тенденция ВР»:

$$L_2 = \langle Name_2, \tilde{X}_2, X_2, G_2, P_2 \rangle$$

где $Name_2$ = «Тенденция ВР» имя ЛП; $\tilde{X}_2 = \{B0, B1, B2, B3\}$ – множество термов ЛП, состоящее из четырех элементов: $B0$ – «покой», $B1$ – «малое движение», $B2$ – «среднее движение», $B3$ – «рывок»; $X_2 \in \{\tilde{\alpha}_{ij}\}$ – область определения, универсальное множество, где $\tilde{\alpha}_{ij} = RTend(\tilde{z}_i, \tilde{z}_j)$, где $\tilde{\alpha}_{ij}$ – нечеткая лингвистическая оценка, характеризующая степень различия значений нечеткий двух углов \tilde{z}_i, \tilde{z}_j .

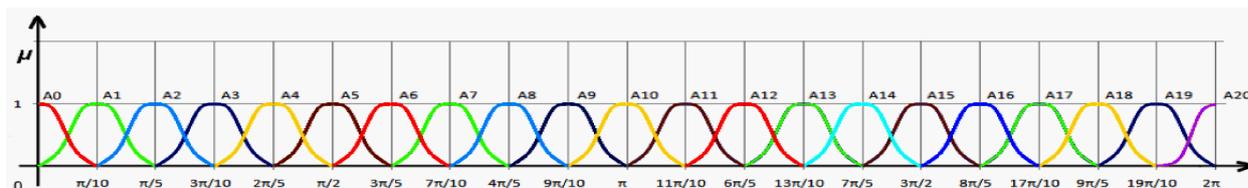


Рис. 3. Функции принадлежности к термам ЛП «Нечеткий угол»

Способ построения нечеткой лингвистической оценки приведен в таблице.

Таблица

Правило построение нечеткой лингвистической оценки

	z_0	z_1	z_2	z_3	z_4	...	z_{20}
z_0	V0	V1	V2	V3	V3	...	V3
z_1	V1	V0	V1	V2	V3	...	V3
z_2	V2	V1	V0	V1	V2	...	V3
z_3	V3	V2	V1	V0	V1	...	V3
z_4	V3	V3	V2	V1	V0	...	V3
...
z_{20}	V3	V3	V3	V3	V3	...	V0

Для решение задачи анализа статических зон тела человека авторами были проанализированы различные методы решения задачи классификации [1, 3, 6]. Лучшим методом оказалась классификация с помощью нейронных сетей типа многослойный персептрон – 96,62 % правильно распознанных образов на всей выборке.

В основе работы подсистемы анализа поз лежат нейросетевые методы. На данном этапе работы выборка состоит из 597 входных векторов, представляющих 61 позу. Таким образом, На каждую позу приходится около 10 обучающих векторов, которые представляют собой различные проявления этой позы. Например, поза «Руки в замок на груди». Субъект, может сидеть, стоять, облокотиться на что-нибудь, но при этом держать руки в замок на груди. Размер каждого вектора составляет 63 переменных. Выборка была поделена на обучающую – 75 % входных образов, тестовую – 15 % входных образов, контрольную 10 % – входных образов.

Нейронная сеть определяет положительная или отрицательная и уверенная или неуверенная эмоциональная реакция. Анализировались 50 нейронных сетей типа многослойный персептрон. Используемый трехслойный персептрон состоит из 63 нейронов на первом слое, 14 на втором и 6 на третьем. Для обучения с помощью алгоритма Бройдена – Флетчера – Гольдфарба – Шанно (Broyden, , Goldfarb, Shanno – BFGS) потребовалось 55 эпох. Функция ошибки – кросс энтропия. Функция активации скрытого слоя – гиперболический тангенс. Функция активации внешнего слоя – софтмакс.

Был проведен анализ на чувствительность нейронной сети. Результаты представлены в виде графика на рис. 4.

По графику можно сделать вывод о том, что нейронная сеть наиболее чувствительна к каналам, которые описывают следующие узлы: плечо, локоть, кисть, бедро, колено, корневой узел модели (общий наклон тела), голова.

Результаты работы позволяют описать характерные позы и их соответствия эмоциональным реакциям с помощью нечетких правил. Для автоматизации процесса распознавания эмоционального состояния человека по движениям и позам разработаны: классификация характерных поз, и их соответствие эмоциональным реакциям; база данных, хранящая интерпретацию, словесное описание и изображение характерных поз; нейросетевая модель идентификации позы человека и эмоциональной реакции. Автоматизирован процесс предобработки данных. Произведена фазификация динамической информации, а также грануляции 1-го и 2-го уровней. Разработана модель представления эмоционального состояния человека по пантомимике в виде нечеткого временного ряда. Разработан подход к разграничению зон тела человека. Разработаны алгоритмы определения временных интервалов. Разработано ПО для автоматизированной предобработки данных и определения эмоциональной реакции.

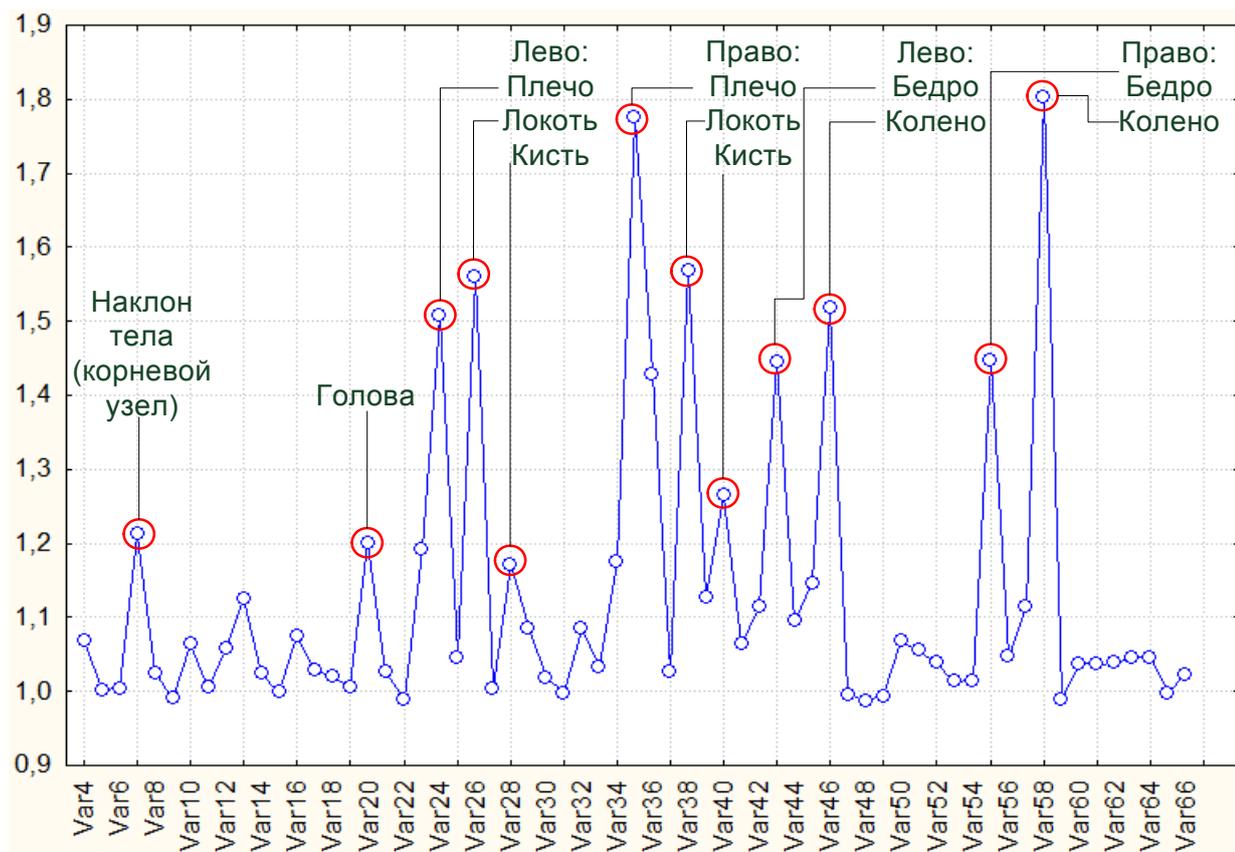


Рис. 4. Анализ нейронной сети на чувствительность

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Розалиев В. Л. Предпосылки, возможности, перспективы создания автоматизированной системы распознавания эмоциональности речи / В. Л. Розалиев // Известия ВолГГТУ. Серия «Актуальные проблемы управления, вычислительной техники и информатики в технических системах»: межвуз. сб. науч. ст. / ВолГГТУ. – Волгоград, 2008. – Вып. 4, № 2. – С. 58–61.
2. Розалиев В. Л. Построение модели эмоций по речи человека / В. Л. Розалиев // Известия ВолГГТУ. Серия «Актуальные проблемы управления, вычислительной техники и информатики в технических системах»: межвуз. сб. науч. ст. / ВолГГТУ. – Волгоград, 2007. – Вып. 3, № 9. – С. 62–65.
3. Розалиев, В. Л. Моделирование эмоционального состояния человека на основе гибридных методов / В. Л. Розалиев, А. В. Заболеева-Зотова // Программные продукты и системы: международный науч.-практ. журнал. – Тверь, 2010 – Вып. 2 (90). – С. 141–146.
4. Заболеева-Зотова, А. В. Автоматизация семантического анализа документации технического задания / А. В. Заболеева-Зотова, Ю. А. Орлова // Вестник компьютерных и информационных технологий. – 2008. – № 9. – С. 26–34.
5. Заболеева-Зотова, А. В. Атрибутная грамматика формального документа «Техническое задание» / А. В. Заболеева-Зотова, Ю. А. Орлова // Изв. ВолГГТУ. Серия «Актуальные проблемы управления, вычислительной техники и информатики в технических системах»: межвуз. сб. науч. ст. / ВолГГТУ. – Волгоград, 2008. – Вып. 4, № 2. – С. 39–43.
6. Заболеева-Зотова, А. В. Автоматизация процедур семантического анализа текста технического задания / А. В. Заболеева-Зотова, Ю. А. Орлова // Известия ВолГГТУ. Серия «Актуальные проблемы управления, вычислительной техники и информатики в технических системах»: межвуз. сб. науч. ст. / ВолГГТУ. – Волгоград, 2007. – Вып. 3, № 9. – С. 49–52.
7. Заболеева-Зотова, А. В. Моделирование лексического анализа текста технического задания / А. В. Заболеева-Зотова, Ю. А. Орлова // Изв. ВолГГТУ. Серия «Актуальные проблемы управления, вычислительной техники и информатики в технических системах»: межвуз. сб. науч. ст. / ВолГГТУ. – Волгоград, 2007. – Вып. 2, № 2. – С. 39–42.

УДК 681.3.01:621.311

*М. В. Щербаков¹, Н. Л. Щербакова¹, А. Бребельс²***МЕТОДИКА ВЫБОРА ЗНАЧИМЫХ ПАРАМЕТРОВ
ДЛЯ КРАТКОСРОЧНОГО ПРОГНОЗИРОВАНИЯ ЭНЕРГОПОТРЕБЛЕНИЯ***¹Волгоградский государственный технический университет, Россия

vstu@gebeus.ru, snl@gebeus.ru

²Католическая высшая школа Кемпен, Бельгия

adriaan.brebels@portacapena.com

Рассматривается проблема выбора наиболее значимых параметров для краткосрочного прогнозирования энергопотребления. Приводятся обоснования выбора, основанные на научно-практических исследованиях и построениях модели прогнозирования. Предложена методика выбора наиболее значимых параметров влияющих на потребления электроэнергии для широкого круга объектов. Методика включает шаги, соответствующие классам параметров: значения потребления электроэнергии в прошлом: сегментация профилей потребления, определение временного лага; значения типов дней; значений о внешней температуре; переменные использования объекта. Приводятся рекомендации по применению методики.

Ключевые слова: краткосрочное прогнозирование, энергопотребление, интеллектуальный анализ данных, выбор переменных модели, модель прогнозирования.

*M. V. Shcherbakov, N. L. Shcherbakova, A. Brebels***FRAMEWORK FOR SELECTING THE SIGNIFICANT VARIABLES
SET FOR SHORT-TERM ENERGY CONSUMPTION FORECASTING**

This article deals with the problem of choosing the most relevant parameters for short-term forecasting of energy consumption. Provides the rationale for selection based on empirical research studies and the construction of prediction model. The selecting framework of the most significant parameters influencing the energy consumption for a wide range of objects is suggested in the paper. The framework includes steps corresponding to the classes of parameters: the values of electricity consumption in the past: the consumer profile segmentation, the definition of time lag; the types of days set, the external temperature values set, usage profile of the object. We give advice on application frameworks.

Key words: short-term forecasting, energy consumption, data mining, choice of model's variables, the prediction model.

Введение

Проблема энергосбережения актуальна для всех цивилизованных стран мира в первую очередь по экономическим причинам. Пути решения каждая страна ищет путем анализа проблем энергетического обеспечения и усовершенствования энергосберегающей политики. Для стран с конкурентным рынком электроэнергии, таких как Великобритания, Финляндия, Германия, Норвегия, Швеция, Испания, Украина, Беларусь, Россия и др., помочь в решении этой проблемы может прогнозирование энергопотребления [1–4]. При более точном прогнозе потребителю приходится меньше электроэнергии покупать на менее выгодном балансирующем рынке. Отклонения фактического графика нагрузок субъекта оптового рынка от заявленных значений больше определенного процента приводит к покупке электроэнергии с балансирующего рынка по большей цене. Отклонение в меньшую сторону тоже карается оплатой непоставленной электроэнергии, определяемой разницей между заявленным и фактическим потреблением по установленным расценкам [5]. В связи с таким положени-

ем возрастает ответственность прогнозирования энергопотребления субъектов на оптовом рынке электроэнергии.

Проблема поиска значимых параметров

Разработка моделей краткосрочного прогнозирования актуальна и для поставщиков и потребителей электроэнергии. Для их построения необходимо воспользоваться общими методиками анализа данных (KDD, CRISP-DM, Data Mining Project Cycle) [6–8].

При прогнозировании работы поставщиков учитываются приблизительно одинаковые исходные данные для обработки. Например, по результатам опроса экспертов – энергетиков для системы Forecast Elektra была составлена таблица значимости факторов влияющих на энергопотребление [9]. Оценка значимости факторов проведена по 100 – бальной шкале. Приведем выдержку из таблицы. Статус дня (подразумевает профиль графика потребления конкретных суток, основанный на статистике нескольких лет, а так же предыдущих недель) (0 – рабочий день со вторника по четверг, 1 – рабочий понедельник, 2 – рабочая пятница,

*Работа поддержана грантом РФФИ № 10-07-97008-р_поволжье_а.

3 – суббота, 4 – воскресенье, 5 – праздничный день) – 100; потребление электроэнергии за текущие сутки – 94; потребление электроэнергии в предыдущие сутки – 72; потребление электроэнергии в предыдущую неделю – 60; потребление электроэнергии две недели назад – 42; среднесуточная температура воздуха – 88.

Состав исходных данных может меняться в зависимости от специфики работы конкретной энергосбытовой компании. Например, ОАО «Омская ЭСК» использует нейросеть для прогнозирования энергопотребления [10] на основании следующих данных: часовой (получасовой) интервал; порядковый день месяца; порядковый номер месяца в году; среднесуточная температура; порядковый номер дня в неделе; признак выходного дня; объем потребления за прошедшие сутки. Наиболее точный результат ($MAPE < 10\%$) достигнут при обучающей выборке из 30 предыдущих дней.

При прогнозировании работы потребителей электроэнергии эти переменные остаются актуальными, но специфика объектов прогнозирования вносит дополнительные переменные, игнорирование которых сводит на нет все методики краткосрочного прогнозирования.

В работе [11] приводятся следующие, нехарактерные для поставщиков электроэнергии, переменные: угол поворота жалюзи, освещенность номера (средние данные по 14-ти площадкам, на которые разбивается комната (около 800 м^2), площадь окон, открытие заслонки в системе с переменным расходом воздуха (%), уровень комфортности.

В исследовании [12] рассматривались торговые центры, размещенные в Бельгии и Голландии. Для прогнозирования потребления электроэнергии в них характерно влияние переменных, не наблюдаемых у других объектов. Специфика заключается в следующем 1) график работы различен для каждого рабочего дня, имеются выходные дни и праздники; 2) значения минимального и максимального потребления в течение дня могут отличаться в 10–12 раз; 3) зачастую показатели датчиков температуры воздуха, установлены не в соответствии с требованиями (не помещены в изоляционные кожухи), что приводит к недостоверным данным; 4) число наблюдений в течение дня равно 96 (т. е. каждые 15 минут); 5) по результатам автокорреляции временного ряда энергопотребления наиболее значимые корреляции найдены между значениями с лагом 1, 672 (96·7), 1344

(96·2·7), что соответствует потреблению 15 минут назад, неделю назад и 2 недели назад; 6) выборка данных составляла 10 месяцев.

Следует отметить, что приведенные результаты опираются на анализ специфики объекта прогнозирования и можно выделить существенную проблему, связанную с 1) отсутствием методики обоснованного выбора значимых параметров и инструментов их выявления; 2) сложностью (невозможностью) автоматизации процесса построения модели прогнозирования потребления электроэнергии. Отметим, что автоматизация процесса прогнозирования может увеличить эффективность процесса и снизить издержки [13–16].

Методика выбора значимых параметров

Ниже представлена методика выбора значимых переменных и включения их в модели прогнозирования потребления, которая может быть автоматизирована. Методика состоит из следующих шагов.

Шаг 1. Выбор и включение значений потребления электроэнергии $E(t, \tau)$ в прошлом.

Шаг 1.1. Сегментация профилей потребления. Определяются сегменты временного ряда соответствующие графику работы здания. Так выделяются: 1) рабочие регулярные дни RW , 2) рабочие нерегулярные NRW (особый режим работы, предпраздничные, послепраздничные), 3) не рабочие дни DO (выходные, праздники). Если объект энергопотребления имеет график работы и он различен для каждого регулярного рабочего дня, то выделяются рабочие регулярные дни соответствующие графикам (как правило, дням недели) RW_k , где $k = 1, \dots, M$, а M – множество различных графиков работы. Таким образом, все множество наблюдений W целесообразно разделить на подмножества W_j , где $j = 1, \dots, N$, N – число сегментов данных, соответствующих наборам RW , NRW , DO . Кроме этого, следует учитывать переход на летнее и зимнее время и в случае необходимости осуществлять сдвиг на 1 час.

Шаг 1.2. Определение временного лага. Требуется определить множество временных отсчетов $T = \{t_1, t_2, \dots, t_n\}$, и множество Γ расстояний между временными отсчетами T , $\Gamma = \{\tau_1, \tau_2, \dots, \tau_{n-1}\}$, где $\tau_i = t_{i+1} - t_i$ для наблюдений W . На данном этапе осуществляется всесторонний анализ временного ряда потребления. Для определения временных точек следует воспользоваться автокорреляционной функцией

ей. Как правило наиболее значимыми переменными могут быть: 1) $E(t - n)$, т. е. день назад, где n – число временных отчетов в течении дня, в которые осуществляется сбор информации; 2) $E(t - 7 \cdot n)$, т. е. неделю назад; 3) $E(t - 365 \cdot n)$, т. е. год назад; 4) потребление в текущий момент времени год назад в днях с такой же средней температурой. Использование автокорреляционной функции позволит определить множество Γ для каждого из набора W_j .

Шаг 2. Включение типов дней. На данном этапе включается информация о типе дня. Как правило, переменные характеризующие тип являются категориальными и требует дальнейшего преобразования.

1. Признак рабочего регулярного, рабочего нерегулярного, нерабочего дня или признак функционирования объекта (например, часы открытия, закрытия в торговых центрах).

2. Признак праздничного, предпраздничного и послепраздничного дня.

3. Номер дня в неделе и номер недели или номер дня в месяце.

4. Номер месяца в году или признак сезона. Этот параметр следует указывать в том случае, если выборка превышает больше чем один год.

5. Целесообразно включать признаки для дней из множества T .

Шаг 3. Включение значений внешней температуры. Как правило, значимость такой переменной зависит от типа здания и производства. Следует рассматривать включение в модель 1) средней температуры за прошедшие сутки и средней прогнозируемой температуры на время прогнозирования; 2) средней температуры за l временных отчетов от точки прогнозирования. Параметр l определяется при выполнении плана экспериментов.

Шаг 4. Профили использования здания. Здесь следует отметить включение в модель следующих показателей: 1) количества людей в здании в момент времени, 2) интенсивность использования потребителей электроэнергии, связанное с технологическими процессами (например, применяется шкала оценки), 3) загрузка предприятия в соответствии с планом производства.

Основываясь на ряде публикаций [1–5, 9–11] и экспериментов при выполнении совместной НИР № 39/267-10 «Разработка подходов к минимизации потребления электроэнергии на основе статистического и интеллектуального анализа временных рядов потребления электро-

энергии» с бельгийской компанией Порта Капена, набор параметров позволяет добиться построения модели с ошибкой $MARE$ 3–10 % в зависимости от специфики рассматриваемого объекта прогнозирования и процедур предварительной обработки информации.

Испытание, результаты и выводы

1. Предложенная в статье методика выбора значимых переменных и включение их в модель прогнозирования потребления электроэнергии базируется на подходах к синтезу прогнозных моделей для различных объектов потребления или поставки электроэнергии. Методика может быть использована 1) на начальном этапе построения моделей прогнозирования; 2) на этапе проектирования систем автоматизации прогноза. Данная методика была применена для реализации моделей прогнозирования потребления электроэнергии в торговых центрах Бельгии и Голландии [8].

2. Для дальнейшего формирования прогнозной модели, основанной на методике, следует отметить, что при изучении и рассмотрении объекта краткосрочного прогнозирования принимается решение о выборе входных параметров, при которых будет наблюдаться наилучший результат. Эти параметры должны учитывать специфику рассматриваемого класса объектов прогнозирования. Например, для краткосрочного прогнозирования энергопотребления в торговых центрах нельзя включать в качестве предиктора значение потребления день назад, т. к. для каждого дня имеется свой график работы, а потребления при закрытом торговом центре значительно ниже, чем при открытом (например, в среду в 21:00 потребление ~ 30КВтЧ, а в четверг в это же время ~ 250 КВтЧ), и другие моменты.

3. Если ранее использовали простой метод линейной регрессии или метод ежедневного сопоставления показателей, то теперь появилась необходимость учитывать нелинейные влияния колебаний температуры, скорости ветра и общей освещенности и т. д. В связи с этим, появилось новое требование к методикам прогнозирования в электроэнергетике и заключается в отказе от укрупненного расчета суточных объемов потребления в пользу учета нагрузки в разрезе 15-минутных интервалов. [17].

4. В соответствии с методикой можно сформировать алгоритм подбора значимых предикторов и последовательного включения

в модель с оценкой улучшения качества прогнозирования. В качестве метода оценки качества может быть использован подход оценки качества для множества моделей и объектов [12].

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Михайлова С. Опыт стран Европы и Азии в сфере энергосбережения. Электронный журнал энергосервисной компании «Экологические системы». – 2007. – № 8 http://esco-ecosys.narod.ru/2007_8/art071.htm (дата обращения: 11.10.2010).
2. Тараканов А. Краткосрочное прогнозирование нагрузок на основе информации комплекса РСДУ2. – 2004. <http://www.ema.ru/view/articles/202> (дата обращения: 11.10.2010).
3. Кудинов Е. В. Прогнозирование нагрузки: состояние и перспективы. <http://www.er.gov.ua/doc.php?p=2432&wid=fa66006014f04a85e29b9853cfe1cd0a> (дата обращения: 11.10.2010).
4. Энергосбережение в Беларуси. <http://www.ecoteco.ru/library/magazine/1/economy/energoberezhnie-v-belarusi/> (дата обращения: 11.10.2010).
5. Канторович С. А. Прогноз потребления электроэнергии // URL <http://www.basegroup.ru/solutions/case/electra/> (дата обращения: 11.10.2010).
6. Larose, D. T. Discovering knowledge in data: an introduction to data mining / Daniel T. Larose // John Wiley & Sons, Inc. ISBN 0-471-66657-2 (cloth). – 2005.
7. Frawley, W. J. Knowledge discovery in databases: an overview/ W. J. Frawley, G. Patetsky-Shapiro, and C. J. Matthews // Cambridge: AAAI/MIT Press. – 1991.
8. Zhao, H. T. Data Mining with SQL Server 2005 / ZhaoHui Tang, Jamie MacLennan // Wiley Publishing, Inc.
9. Климов А. Forecast Elektra – прогнозирование потребление электроэнергии для энергосбытовых компаний. http://www.bi-grouplabs.ru/Rech/electricity/forecast_elektra.html (дата обращения: 11.10.2010).
10. Грицай А., Гаак А. Краткосрочное прогнозирование потребления электроэнергии с использованием нейронной сети. Энергорынок. – 2007. – № 11. <http://www.e-m.ru/er/2007-11/23347/> (дата обращения: 11.10.2010).
11. Satsuki Yoneda, Takashi Kurio, Minoru Yoshida, Takaharu Kawase, Mutsuo Honma, Tamio Ito. Knowledge Management for comfortableness and the energy saving of building management. Seventh International IBPSA Conference. Rio de Janeiro, Brazil. – 2001.
12. Brebels, A. Mathematical and statistical framework for comparison of neural network models with other algorithms for prediction of Energy consumption in shopping centres / Shcherbakov M. V., Kamaev V. A., et all // Proceedings of the 37 Int. Conf. Information Technology in Science Education Telecommunication and Business, suppl. to Journal Open Education. Yalta – Gurzuf. – 2010. – P. 96–97.
13. Заболева-Зотова А. В., Орлова Ю. А. Автоматизация процедур семантического анализа текста технического задания // Известия Волгоградского государственного технического университета. – 2007. – Вып. 3, № 9 (35). – С. 49–53.
14. Щербаков М. В. ICDMS – программное обеспечение как сервис для решения задач идентификации на основе коннективистских систем // Известия Волгоградского государственного технического университета. – 2009. – Т.12, №. 7. – С. 88–91.
15. Panchenko, D. P. & Shcherbakov, M.V. (2009) Implementation of Genetic Algorithms for Transit Points Arrangement // Intelligent Processing : suppl. to Int. Journal «Information Technologies and Knowledge» Vol. 3. – Int. Book Series «Information Science & Computing», 9, 129–131.
16. Камаев В. А., Щербаков М. В., Скоробогатченко Д. В. Автоматизированная система прогнозирования транспортно эксплуатационного состояния дорог. Вестник компьютерных и информационных технологий, № 4, 2004, М.: «Машиностроение», С. 2–6.
17. Коморник С., Калечиц Е. Требования к системам прогнозирования энергопотребления. Энергорынок. – 2008. – № 3. http://www.e-m.ru/er/2008-03/23417/?phrase_id=82928 (дата обращения: 11.10.2010).

УДК 681.3.01:621.311

*М. В. Щербаков¹, Н. Л. Щербакова¹, Д. П. Панченко¹, А. Бребельс², А. П. Тюков¹, М. А. Аль-Гунаид¹***СПЕЦИФИКА ПРИМЕНЕНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ МОДЕЛЕЙ
АНАЛИЗА ДАННЫХ ДЛЯ ПОВЫШЕНИЯ ЭНЕРГЕТИЧЕСКОЙ ЭФФЕКТИВНОСТИ***¹Волгоградский государственный технический университет, Россия

vstu@gebeus.ru, snl@gebeus.ru

²Католическая высшая школа Кемпен, Бельгия

adriaan.brebels@khk.be

Рассматривается специфика применения интеллектуальных моделей анализа данных для решения комплексной задачи повышения энергетической эффективности. Вводится понятие интеллектуального управления энергосбережением. Дается формализация коннективистских систем как базиса для построения интеллектуальных моделей. В статье предлагаются следующие методики: решения задачи прогнозирования потребления электроэнергии; выявления закономерности между потреблением электроэнергии и температурой на базе моделей деревьев решения и моделирования «что-если»; выявления выбросов на основе моделей классификации и кластерного анализа. Приводятся результаты испытания методик и выводы.

Ключевые слова: интеллектуальный анализ данных, энергетическая эффективность, энергосбережение, краткосрочное прогнозирование, коннективистская система, выявление выбросов.

*M. V. Shcherbakov, N. L. Shcherbakova, D. P. Panchenko, A. Brebels, A. P. Tyukov, M. A. Al-Gunaid***INCREASING THE ENERGY EFFICIENCY THROUGH DATA MINING**

Paper considers the specifics of the data mining models implementation for solving the complex task in improving energy efficiency. The concept of intelligent energy saving management is introduced. The connectionists systems formal description as a basis for the models generating is represented. The article proposes the following frameworks: the short term energy consumption forecast framework, methodology to identify patterns between energy consumption and temperature profiles based on the decision tree model and «what-if» modeling; framework for outliers detection based on classification and cluster analysis. The results of implementation and conclusions are discussed.

Key words: data mining, energy efficiency, energy saving, connectionists system, short-term forecasting, outliers detection.

Введение

Проблема эффективного использования электроэнергии является актуальной для предприятий и производств в странах со свободным сектором экономики. Безусловно, каждое предприятие может иметь ряд установленных нормативных документов, оптимизирующих потребление электроэнергии и связанных с производственным циклом. В соответствии с Федеральным законом от 23.11.2009 N 261-ФЗ «Об энергосбережении и о повышении энергетической эффективности...» предприятиям необходимо реализовывать меры по повышению энергосбережения [1]. Начальным мероприятием является учет и анализ текущего потребления электроэнергии. Системы сбора и передачи информации широко распространены зарубежом и на предприятиях России. Например, в систему EcoSCADA, разработанную Бельгийско-Польской компанией Порта Капена каждые 15 минут передается различные данные, в том числе и о потреблении электроэнергии [2, 3].

Следует выделить направление интеллектуального анализа данных в сфере энергосбережения, позволяющего выявлять закономерности

между потреблением электроэнергии и различными факторами окружающей среды, параметрами использования производственных мощностей и другими, которые собираются в SCADA системах [4, 5]. Такое направление можно назвать – *интеллектуальным управлением энергосбережением*, представляющим процесс формирования и применения комплекса мер эффективного использования топливно-энергетических ресурсов на основе интеллектуального анализа данных.

При решении задачи интеллектуального управления энергосбережением можно выделить частные задачи, решения которых рассмотрены в статье.

1. В первую очередь рассматривается актуальная проблема краткосрочного прогнозирования суточного потребления электроэнергии. Достоверный прогноз позволяет точно оценить затраты на электроэнергию, определить временные интервалы в которых ожидается максимальная нагрузка на сеть. Кроме этого, повышение точности прогнозирования, может рассматриваться как один из источников экономии энергоресурсов [6].

*Работа выполнена при финансовой поддержке РФФИ (проект № 10-07-97008-р_поволжье_а).

2. Выявление закономерностей между потреблением электроэнергии и погодными условиями (например, температурой окружающего воздуха). Такой анализ позволит на основании прогнозных значений потребления электроэнергии осуществлять гибкую настройку программ систем климат контроля (HVAC) и добиваться снижения потребления электроэнергии в рамках поддержания установленного уровня комфортности.

3. Выявление аномальных значений в данных и их объяснение. В процессе передачи информации от сенсора к базе данных могут возникать проблемы. Их причинами являются либо сбой оборудования, либо аномальный режим работы объекта. Выбросы могут характеризоваться значительным завышением потребления электроэнергии, либо, напротив, снижением до 0 кВт·ч. Выявление таких аномальных значений позволит оптимизировать процедуры контроля за сбором и передачей информации и определять события, связанные с резким повышением потребления электроэнергии.

Характеристика объекта

В качестве объектов рассматриваются здания торговых центров, размещенных в Бельгии и Голландии. Отметим специфику рассматриваемых объектов: 1) имеется график работы (часы открытия, закрытия, выходные дни и праздники) торговых центров; 2) потребление электроэнергии значительно зависит от графика работы (потребление в ночные часы или в праздники может отличаться от дневного рабочего в 10–12 раз). 3) число временных отсчетов в течение дня равно 96; 4) в системе EcoSCADA также учитывается информация о параметрах, влияющих на потребление электроэнергии, такие как, погодные условия (температура воздуха, влажность, интенсивность солнечного освещения) и условия использования здания (время открытия, закрытия, календарь).

Формализация коннективистских систем

В качестве базиса интеллектуальных моделей предлагается использовать понятие коннективистских систем [7]. Пусть объект характеризуется кортежем

$$O = \langle t, Y, U, \Xi, W \rangle \tag{1}$$

где Y – множество переменных состояния, $Y = \{y_i\}$, $i = 1, \dots, |Y|$, U – множество управляющих воздействий, $U = \{u_i\}$, $j=1, \dots, |U|$ –

множество возмущающих воздействий $\Xi = \{\xi_k\}$, $k = 1, \dots, |\Xi|$. В каждый момент времени $n = 1, 2, \dots, T$ ко входам объекта приложены внешние управляющие воздействия $U(n)$, возмущения $\Xi(n)$ и регистрируются значения состояний $Y(n)$. Задана $W_{(y,u,\xi),n}$ – матрица значений элементов множеств Y, U, Ξ в коротком интервале наблюдения (КИН) T .

Отметим следующие особенности объекта идентификации (ограничения на класс рассматриваемых объектов).

а) $|Y| = 1$.

б) Задана область изменения переменных $\{y_i\}$, $\{u_i\}$, $\{\xi_k\}$, т. е. для каждой переменной существует верхнее и нижнее значения.

Коннективистская система представляется в виде

$$CS = \langle UN, RE, LY, F, PR, \Phi \rangle \tag{2}$$

где UN – множество элементов, RE – множество связей, LY – множество групп элементов (слов), F – множество функций преобразования информации UN , PR – процедура настройки значений связей RE для минимизации функционала Φ .

Схематично такая структура представлена на рис. 1.

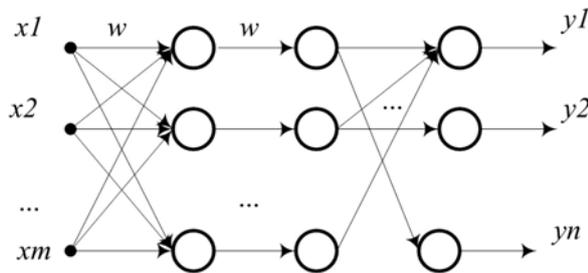


Рис. 1. Структура коннективистской системы

Формально коннективистская система (2) при данной формализации обобщает в себе различные модели (например, скользящего среднего, линейной регрессии, нейросетевые модели).

Выделим фреймовую модель коннективистской системы

$$FCS = \langle FR, CS, AG, \Phi \rangle \tag{3}$$

где FR – множество фреймов объединяющих подмножество систем CS , AG – множество коннективистских систем, выполняющих агрегацию выходной информации систем CS . Такое обобщение позволяет автоматизировать процесс

подбора прогнозных моделей, в процессе которого уменьшается значение функции потерь Ф.

Методика и решение задачи прогнозирования потребления электроэнергии с применением коннективистских моделей

Для применения коннективистских моделей предлагается методика формирования прогнозной модели, основанная как на классических подходах к прогнозированию электроэнергии, так и на подходах извлечения знаний из данных CRISP-DM [8], KDD [9] и Data Mining Project Cycle (DM PC) [10]. Методика прогнозирования формально относится к методике краткосрочного прогнозирования [6], однако, горизонт прогнозирования равен 96 точкам. В качестве функции потерь выбрано среднеквадратичное отклонение прогнозного и измеренного значения потребления электроэнергии.

Ниже дано общее описание методики для построения и апробации коннективистской системы прогнозирования потребления электроэнергии для торговых центров на следующий день.

1. Формирование цели создания модели прогнозирования или определение функции потерь.

2. Анализ потребителей электроэнергии и выявление прямых и косвенных признаков, влияющих на потребление. Целью данного этапа является принятие решения о формировании множества независимых переменных для включения в модель.

3. Предварительный анализ данных. Данный шаг содержит 4 основных процедуры: исключение аномалий, восстановление пропусков в данных (подстановки среднего значения), кодирование категориальных переменных и нормализация.

4. Построение модели и ее настройка. На данном шаге осуществляется выбор типа модели, составление плана экспериментов испытания модели, параметрическая оптимизация модели.

5. Определение качества модели. Последний шаг связан с определением качества модели и принятием решения о качестве модели в соответствии с целями прогнозирования (функции потерь). Выделим следующие классические оценки определения точности модели: среднеквадратичное отклонение RMSE и средняя абсолютная процентная ошибка (MAPE).

В качестве базовой коннективистской системы применяются нейросетевые модели и разработанная система анализа данных [11].

Нейросетевая модель, включающая потребление энергии и параметр использования зда-

ний. Предполагается зависимость потребления энергии от текущей температуры, энергопотребления и профиля использования здания в тот же момент времени одну и две, три и четыре недели назад. Также предполагается использование статуса (открытый/закрытый). Для данной модели RMSE, кВт ч составил 17,97 и 14,78 на обучающей и тестовой выборках соответственно, MAPE – 14,01 % и 11,06 % соответственно.

Нейросетевая модель, включающая нелинейную комбинацию параметров. Предполагается зависимость потребления энергии от нелинейной комбинации следующих параметров: текущей температуры, энергопотребления и профиля использования здания в тот же момент времени одну и две, три и четыре недели назад, статуса здания (открытый/закрытый), средней температуры 8 точек назад.

Для данной модели RMSE, кВт ч составил 12,36 и 24,51 на обучающей и тестовой выборках соответственно, MAPE – 11,15 % и 23,17 % соответственно.

Методика решения задачи выявления закономерности между потреблением электроэнергии и температурой

Выявление закономерностей между потреблением электроэнергии и погодными условиями является одной из самых востребованных задач. Используя значение температуры можно реализовать более точную прогнозную модель. Решение такой задачи позволяет ответить на вопрос об изменении потребления электроэнергии (в кВт ч) при изменении температуры на N градусов.

Для решения такой задачи предлагается следующий подход.

1. Сбор данных о потреблении электроэнергии и температуре окружающей среды.

2. Формирование модели на базе дерева решений, включающей в качестве входных параметров: значение температуры в текущий момент времени, значение потребления электроэнергии в предыдущие моменты времени, дискретные временные отсчеты в течении дня.

3. Построение модели «что-если» и получение значений потребления электроэнергии в текущий момент времени при изменении показаний температуры от минимального значения к максимальному.

4. Количественная оценка изменения выходного параметра в зависимости от изменения входного (температуры).

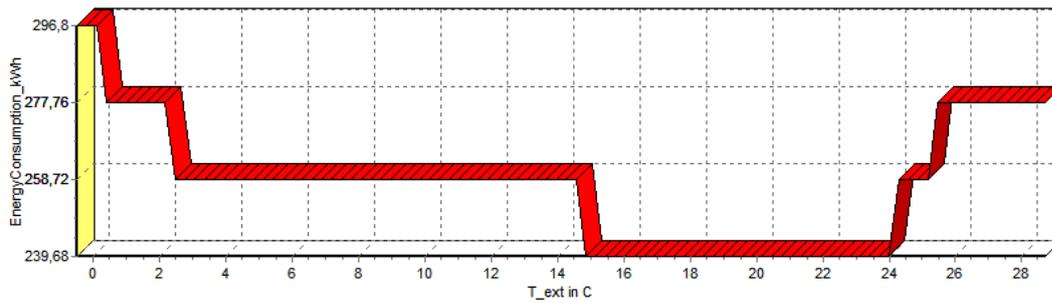


Рис. 2. Результаты моделирования зависимости потребления электроэнергии от внешней температуры

Следует отметить особенность, связанную с необходимостью квантования выходных параметров. Алгоритмом квантования может выступать подход кластерного анализа, описанный в следующем пункте. На рис. 2 показан график зависимости потребления электроэнергии от внешней температуры (результат получен с использованием аналитической системы Deductor [14]). По оси абсцисс отражены значения температуры °C в интервале $[-0.5; 28.5]$, по оси ординат – значения среднего потребления электроэнергии (КВт·ч) в дневные часы для данных значений температур. Можно сформулировать следующие выводы для данной задачи.

1. Существует интервал температур, при котором потребление электроэнергии минимально. Это связано с т. н. базовой температурой (для Бельгии 16 °C, для России 18 °C) при которой нет необходимости в нагреве или охлаждении помещения.

2. Использование электрических приборов нагрева при температуре близкой к 0 является более затратным, нежели использование охлаждающих приборов к температуре близкой к максимально наблюдаемой (в зданиях с электрическими системами обогрева или смешанными электричество – газ).

3. Изменение температуры на 2 °C зимой приводит к среднему увеличению энергопотребления на 38 КВт·ч для рассматриваемых зданий.

Выявление аномальных значений в данных и их объяснение

Выбросы (аномалии) – наблюдения, сильно отличающиеся от основной массы элементов выборки. Они возникают в силу различных причин: сбоя оборудования, значительного изменения режима работы объекта обследования. Выбросы существенным образом влияют на статистические характеристики выборки, что

приводит к получению менее точных прогнозных моделей. Для выявления выбросов использовались самоорганизующиеся карты Кохонена и алгоритм кластеризации k -means.

Так для решения использована следующая методика.

1. Трансформация временного ряда в N временных рядов длиной D , где N – число временных наблюдений в течение дня, а D – число дней наблюдений. Т.о. каждый новый временной ряд представляет множество значений потребления электроэнергии в фиксированный момент времени в различные дни (например, первый ряд включает значения потребления в 00:15 в дни входящие в интервал наблюдения).

2. Формирование коннективистской системы, реализующей алгоритм классификации Кохонена. Число нейронов входного слоя равно 1, что соответствует значению потребления в d день ($d \in D$). Число нейронов в рабочем слое Кохонена может варьироваться, но не быть меньше D .

3. Провести обучение сети Кохонена.

4. Проанализировать полученные кластеры. В случае принятия решения о том, что в составе кластера входят выбросы, заменить выбросы средним значением кластера, имеющего максимальное число элементов.

Данный подход использовался для выявления выбросов во временном ряду предыдущей задачи. Были выявлены следующие кластеры (результаты приведены для временного отсчета 00:15): 1 кластер (30,4 % элементов) – в который вошли значения из интервала $[18; 19.2]$; 2 кластер (51,8 % элементов) – самый значимый, в который вошли значения из интервала $[20.4; 22.8]$; 3 кластер (10,7 % элементов) – в который вошли значения из интервала $[40.8; 56.4]$ и 4 кластер (7,1 % элементов) – в который вошли значения из интервала $[130.8; 133.2]$. Отметим, что 2 последних кластера можно расце-

нить как выбросы (значительное в ~2 и ~6 раз превышение значений в кластерах № 1 и № 2).

Отметим, что данная процедура может быть формализована и выполняться автоматически при различных заданных порогах чувствительности алгоритмов.

Результаты и выводы

1. В статье рассматриваются методики решения задач, связанных с интеллектуальным управлением энергосбережением на базе моделей интеллектуального анализа данных и базиса коннективистских систем. Рассмотрены 3 актуальные практические задачи.

2. Предложена методика решения задачи прогнозирования потребления электроэнергии с применением коннективистских систем, основанная как на классических подходах к прогнозированию электроэнергии, так и на подходах извлечения знаний из данных. Наиболее точной, из рассматриваемых методов прогнозирования, является модель искусственных нейронных сетей, включающая потребление энергии и параметр использования зданий. Использование в нейронной сети на основе нелинейной комбинации параметров в качестве входных параметров факторов, полученных комбинацией исходных параметров, привело к снижению ошибки прогнозирования (RMSE снизился на 5,61 %). Данная методика может быть использована для решения задачи краткосрочного суточного прогнозирования потребления электроэнергии торговых центрах (или объектов подобного класса) для которых объем выборки данных менее одного года.

3. Предложена методика решения задачи выявления закономерности между потреблением электроэнергии и температурой. Данная методика может использовать любую коннективистскую модель, имеющую в качестве набора входных переменных значения потребления электроэнергии и температуры. Анализ и количественный расчет осуществляется на основе модели «что-если» при изменении параметра температуры от минимума к максимуму.

4. Предлагается методика выявления выбросов на основе моделей классификации и кластерного анализа. Данный подход в отличие от статистических подходов, основанных на пара-

метрах выборки позволяет учитывать знания экспертов при принятии решения о принадлежности группы переменных выбросам.

Рассмотренный спектр задач и использованные подходы интеллектуальной обработки информации может быть использован для повышения энергетической эффективности.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Федеральный закон от 23.11.2009 N 261-ФЗ «Об энергосбережении и о повышении энергетической эффективности и о внесении изменений в отдельные законодательные акты Российской Федерации» (принят ГД ФС РФ 11.11.2009)
2. Европейская программа осведомленности о потреблении электроэнергии BeAware: [сайт]. URL: <http://www.energyawareness.eu/beaware/> (дата обращения: 30.06.2010).
3. Система сбора информации о потреблении электроэнергии ЭкоСкада [Электронный ресурс]. – Режим доступа: <http://www.ecoscada.com>
4. AleaSoft Energy Forecast – Режим доступа: [<http://www.aleasoft.com/>] (дата обращения: 01.09.2010)
5. The TESLA Model Forecast Solution [<http://www.teslaforecast.com/TeslaModel.aspx>] (дата обращения: 01.09.2010)
6. Меламед М. А. Современные методы анализа и прогнозирования режимов электропотребления в энергетических системах // Итоги науки и техники. – Сер. Энергетические системы и их автоматизация. – 1988. – Т. 4. – С. 4–11.
7. Kasabov N. Evolving connectionists systems. The Knowledge Engineering Approach. Softcover. 2007. – 451 p.
8. LAROSE, D. T. Discovering knowledge in data: an introduction to data mining / Daniel T. Larose // John Wiley & Sons, Inc. ISBN 0-471-66657-2 (cloth). – 2005.
9. FRAWLEY, W. J. Knowledge discovery in databases: an overview/ W. J. Frawley, G. Patetsky-Shapiro, and C. J. Mathews // Cambridge: AAAI/MIT Press. – 1991.
10. ZHAO, H. T. Data Mining with SQL Server 2005 / ZhaoHui Tang, Jamie MacLennan // Wiley Publishing, Inc.
11. Щербачков М. В. ICDMS – программное обеспечение как сервис для решения задач идентификации на основе коннективистских систем // Известия ВолгГТУ, серия Актуальные проблемы управления, вычислительной техники и информатики в технических системах. 2009. № 7. – С. 88–91.
12. Камаев В. А., Щербачков М. В., Скоробогатченко Д. В. Автоматизированная система прогнозирования транспортно-эксплуатационного состояния дорог // Вестник компьютерных и информационных технологий. 2004. № 4. С. 2–6.
13. Аналитическая платформа Deductor www.basegroup.ru (дата обращения: 01.09.2010)
14. ГОСТ Р 51387–99 Энергосбережение. Нормативно методическое обеспечение. Основные положения.
15. ГОСТ Р 51541–99 Энергосбережение. Энергетическая эффективность. Состав показателей. Общие положения.

УПРАВЛЕНИЕ В СОЦИАЛЬНЫХ И ЭКОНОМИЧЕСКИХ СИСТЕМАХ

УДК 002:378

Ю. М. Брумштейн, В. П. Ватuleва, Н. Ф. Мангаладзе, В. Я. Тналиева

АНАЛИЗ НЕКОТОРЫХ НЕТРАДИЦИОННЫХ ВОЗМОЖНОСТЕЙ ПОСТРОЕНИЯ И ИСПОЛЬЗОВАНИЯ ТЕСТОВЫХ ЗАДАНИЙ С ПРЕДЛАГАЕМЫМИ НАБОРАМИ ОТВЕТОВ

Астраханский государственный университет

brum2003@mail.ru, kubba@rambler.ru, mangaladze@mail.ru, t_vero_nica_ya@mail.ru

Дана общая характеристика тестовых заданий (ТЗ). Предложены варианты диверсификации формулировок вопросов в ТЗ. Исследованы подходы к автоматизации создания ТЗ. Показана целесообразность использования базы ТЗ и описаны методики формирования на ее основе тестов. Проанализированы вопросы авторских прав на ТЗ и тесты. Представлены некоторые дополнительные варианты технологических приемов тестирования и оценки их результатов по отдельным ТЗ и тестам.

Ключевые слова: тестовые материалы, тестовые задания, диверсификация формулировок, автоматизация разработки, база тестовых заданий, авторские права, приемы тестирования, оценка результатов.

J. M. Brumshteyn, N. F. Mangaladze, V. P. Vatulova, V. J. Tnalieva

THE ANALYSIS OF SOME NONCONVENTIONAL POSSIBILITIES FOR CONSTRUCTION AND USE TEST TASKS WITH OFFERED ANSWERS SETS

The Astrakhan state university

The general characteristic of test tasks (TT) is given. Variants for formulations diversification of questions in TT are offered. Approaches to TT creation automation are investigated. The expediency of TT base using is shown and formation techniques of tests on its basis are described. Questions of copyrights on TT and tests are analyzed. Some additional variants of testing processing methods and their results estimation for separate TT and tests are presented.

Key words: Test materials, Test tasks, Formulations diversification, Development automation, Base of test tasks, Copyrights, Testing receptions, Results estimation.

Применение компьютеризованных систем тестирования (КСТ) может быть эффективным средством контроля знаний при обучении [3], способом определения интеллектуального уровня тестируемых лиц и т. д. Для применения КСТ создаются и используются многочисленные программные оболочки и тестовые материалы [1, 4]. Однако некоторые частные вопросы, связанные с КСТ, остаются исследованными недостаточно полно и глубоко. Ниже предпринята попытка системного анализа таких вопросов для тестовых заданий (ТЗ) с «предлагаемыми наборами ответов» (ПНО) [1], которые содержат текст. Этот тип ТЗ наиболее распространен – его доля в общей массе составляет до 80–90 %. Другие типы ТЗ: ранжирование ответов; на установление соответствия; со «свободными» ответами.

Общая характеристика ТЗ с ПНО и тестов на их основе

Каждое ТЗ с ПНО содержит *вопрос и несколько ответов*. Тестируемому обычно сообщается что: (а) только один ответ верный; (б) верны один или больше ответов. Для «а» типичен (например, [4]) выбор ответа с помощью «радиокнопки»; для «б» – «чек-боксов». Другие возможные варианты указания количества правильных ответов (ПО) для вопроса почти не применяются: (1) точное значение количества ПО; (2) « n или больше»; (3) «не более n »; (4) интервал – «от n_1 до n_2 ». Варианты без верных ответов (для «2» – $n = 0$, для «3» $n = 0$, для «4» $n_1 = 0, n_2 > 0$) – не используются.

Вероятность случайного правильного выбора набора ПО (СПВНПО) при равной «правдоподобности» ответов можно оценить по фор-

мулам комбинаторики. Для неравной правдоподобности можно применить стохастическое моделирование с достаточно большим числом «испытаний». ТЗ с выбором неверных ответов из числа предложенных, применяются реже и считаются более сложными чем с выбором ПО.

Обработка результатов ТЗ с ПНО обычно предполагает «четкую логику» – каждый из ответов может быть верен или нет. Поэтому практически не применяются в вопросах формулировки типа: «Выберите маловероятные варианты ответов»; «Выберите варианты, которые встречаются достаточно часто», а в ответах – ««да – скорее да – скорее нет – нет» (хотя они очень распространены при проведении анкетных опросов).

В ТЗ с ПНО могут содержаться: (а) только текст, в т. ч. и не русскоязычный; (б) формулы и числа; (в) текст + формулы; (г) текст + графика (статическая и/или анимированная); (д) текст + звук; (е) текст + графика + звук. На практике используются в основном варианты «а» и «в», в меньшей степени «б» и «г». Количество ТЗ в тесте, их сложность, типы и пр. определяются целями тестирования, необходимой сложностью ТЗ, планируемым расходом времени. Обычно в КСТ есть «рекомендации» по ограничению количества слов в «вопросах-ответах». Поэтому подготовка коротких формулировок с однозначным толкованием, требует творческих усилий. Для тестов, используемых в вузах (в т. ч. для дистанционных технологий обучения), типична «ручная» разработка индивидуальных наборов ТЗ для каждого теста. Тогда в качестве объекта авторских прав (АП) по ГК РФ, ч. 4, гл. 71 может выступать тест, если его создание носило творческий характер. При этом имущественные права на тест могут принадлежать организации-работодателю, а личные неимущественные – непосредственно автору-разработчику (или их коллективу). С позиций АП возможен переход материалов тестов в категорию «общественного достояния» (ГК РФ, ст.1282), но только через 70 лет после смерти автора. Практика отказа от АП через лицензию Creative-Commons (знак «СС» – права не охраняемы) пока не получила в России большого распространения.

Сложность теста определяется: сложностями входящих в него ТЗ; их количеством; плановым временем тестирования. Сложность отдельных ТЗ может быть оценена: экспертно [2]; по некоторым формальным показателям

для совокупностей «вопросы-ответы»; по результатам предшествующих тестирований групп обучающихся [3] и др. Для последнего случая традиционна [3] оценка «сложности» (фактически – «легкости») для i -го ТЗ (D_i) по $D_i = E_i$, где: E_i – «экспериментальное» значение доли лиц верно выполнивших ТЗ. Более объективным и интуитивно понятным было бы

$$D_i = 1/((E_i - R_i)/R_i) = R_i/(E_i - R_i) \quad (1)$$

где: R_i – доля верно выполненных ТЗ при случайном выборе ответов.

В силу высокой трудоемкости создания ТЗ актуальны *вопросы автоматизации* этого процесса, предотвращения и устранения ошибок/недочетов в ТЗ. Такие средства есть в большинстве «средств разработчика» КСТ, в т. ч. в виде шаблонов «вопрос-ответы» для ТЗ с ПНО. Если «вопросы-ответы» для ТЗ разрабатываются в текстовых редакторах (с последующей «конвертацией» для конкретного КСТ), то используются встроенные средства контроля орфографии и синтаксиса, при необходимости – тезаурусы для поиска синонимов. Однако логические ошибки в формулировках «вопросов-ответов» при этом не выявляются. Для поддержки разработчиков ТЗ эффективным было использование «выпадающих списков» для типовых формулировок «вопросов-ответов». Например, для вопросов: «*термин *** определяется как*»; «*выберите правильный ответ из числа представленных ниже*». Для ответов в ТЗ – «*ни один из представленных вариантов ответов неверен*» и др. Такие списки могли бы формироваться разработчиками по мере создания тестов. Другие направления поддержки: частотный анализ текста материала, по которому необходимо составлять совокупность ТЗ; сравнение частот встречаемости слов (с учетом словоформ) в совокупности ТЗ и в учебнике по курсу/(разделе курса), для которого создается тест. Полностью автоматическое формирование «вопросов-ответов» для ТЗ в настоящее время является трудноразрешимой задачей.

Формирование базы тестовых заданий и создание тестов на их основе

В ряде случаев целесообразно формирование некоторой *общей базы ТЗ и компоновка* из них тестов (в т. ч. и автоматизированная – с заданными характеристиками) для: регулярных тестирований групп учащихся; тестирования лиц, обучающихся по «индивидуальным траек-

ториям»; разовых тестирований по определенным темам (дидактическим единицам); комплексных тестирований по группам тем и пр. Для ТЗ, предназначенных для компоновки из них тестов, целесообразно «прикрепление» дескрипторов, описывающих их сложность, авторские права (имущественные и личные неимущественные), тематическую направленность (ТН), тип и пр.

Автоматизированный отбор ТЗ для конкретных тестов в отношении сложности может осуществляться по условиям: не выше заданной; в указанном интервале сложностей; не ниже заданной; с заданными долями ТЗ для уровней сложности низкая, средняя, высокая (или для диапазонов уровней сложности по формуле (1), заданных в количественной форме).

АП на отдельные ТЗ могут быть, если их содержание носит оригинальный, творческий характер (ТХ). Когда в ТЗ используются графика и/или звук, то с позиций ГК РФ такие ТЗ могут считаться «сложными» объектами (ст. 1240). Если признаются АП на отдельные ТЗ, то тесты должны рассматриваться как «составные» произведения (ст. 1259, п. 2, подп.2 ГК РФ), при условии, что «подбор» ТЗ для них носил ТХ (говорить о ТХ «расположения» ТЗ в тесте можно, лишь если при тестировании не меняется порядок их воспроизведения). С позиций ГК РФ АП составителей тестов могут существовать независимо от АП на отдельные ТЗ.

В простейшем случае одно ТЗ может *относиться* только к одному ТН. При этом для него достаточен один «код классификации» (КК), в т. ч. и «составной» (иерархический) – с указанием темы и подтемы (по типу используемого в [4]). Однако чаще ТЗ соответствуют нескольким ТН, причем в разной степени. Тогда описание тематического профиля теста (ТПТ) целесообразно в виде совокупности пар чисел, первое – определяет КК ТН, второе – процент соответствия ТЗ этому ТН (количество пар для разных ТЗ может различаться). При наличии базы ТЗ это даст возможность, в частности: использовать отдельные ТЗ в разных тестах; осуществлять отбор «ТЗ», соответствующих сразу нескольким ТН (путем задания пороговых «процентов соответствия»); оценивать охват совокупностью ТЗ в тесте всех необходимых ТН. Оценим ТПТ вектором $\{\Psi_i\}_{i=1...I}$

$$\Psi_i = (\sum_{j=1}^J \xi_{i,j}) / T; \quad T = \sum_i \sum_{j=1}^J \xi_{i,j} \quad (2)$$

где: $0 \leq \Psi_i < 1$; I – количество учитываемых ТН для ТПТ (определяется всеми КК, которые есть хотя бы в одном ТЗ теста); J – количество ТЗ в тесте; $\xi_{i,j}$ – процент соответствия j -го ТЗ i -му КК. Соответствие фактического ТПТ $(\{\Psi_k\}_{k=1...K})$ и желательного $(\{\Theta_l\}_{l=1...L})$ может быть оценено парным коэффициентом корреляции по Пирсону между $(\{\Psi_m^*\}_{m=1...M})$ и $(\{\Theta_m^*\}_{m=1...M})$, где: m – определяется всеми КК входящими хотя бы в один из наборов $\{\Psi_k\}$ и $\{\Theta_l\}$, а $\{\Psi_m^*\}$ и $\{\Theta_m^*\}$ представляют собой наборы $\{\Psi_k\}$ и $\{\Theta_l\}$, дополненные нулевыми значениями для отсутствующих КК.

Некоторые вопросы технологии проведения тестирования

Для заранее выбранного набора ТЗ в тесте типичны *последовательности* предъявления: в предопределенной последовательности; каждое ТЗ однократно в случайном порядке; тестируемые могут сами выбирать последовательность работы с ТЗ. Подварианты для последнего случая: однократная работа с каждым ТЗ; возможность возврата к нему и корректировки ответов (если не было подтверждено «завершение» работы с ТЗ или тестом в целом). Единообразие набора ТЗ во всех трех вариантах может приводить к «взаимным консультациям» тестируемых во время прохождения теста. Повторное предъявление некоторых ТЗ в случайном порядке (и со случайным порядком показа ответов) может быть полезным для оценки «не случайности» выбора, т. е. надежности результатов тестирования.

Схемы «адаптации» ТЗ при адаптивном тестировании могут быть различны [3]. В простейшем случае по результатам анализа предыдущих ответов может определяться «группа сложности» очередного ТЗ для предъявления. Далее из базы на основе дескрипторов выбирается конкретное ТЗ (или меняется уровень сложности предопределенного ТЗ).

Варианты политики *продолжительности тестирования*: без ограничения времени на ответы; с ограничением времени по совокупности всех ТЗ, их групп, индивидуально каждого ТЗ. Кроме того, для каждого ТЗ могут быть заданы два времени: предельное и контрольное – причем пропорционально превышению последнего начисляются штрафные баллы (ШБ).

Иногда может быть целесообразным давать тестируемым «повторные попытки» на ТЗ (до получения полностью правильного ответа),

но с начислением ШБ за такие попытки. При этом тестируемому целесообразно сообщать «степень правильности» ответа, а не просто «ТЗ не засчитано».

Для повышения объективности контроля самостоятельности работы тестируемых помимо традиционного наблюдения сотрудниками отделов тестирования, преподавателями и специалистами-тестологами может быть целесообразным организация видеонаблюдения через web-камеры.

В большинстве КСТ предусмотрена возможность представлять при тестировании ответы для каждого ТЗ в случайном порядке. При достаточно большом количестве разработанных ответов к ТЗ может осуществляться «управление» количеством воспроизводимых ответов (КВО), что будет влиять на сложность ТЗ (формально – через снижение вероятности СПВНПО). При этом, если выбор ответов из «базового» списка осуществляется случайно, то необходимо обеспечить: «сохранение» сочетания правильных и неправильных ответов, соответствующих формулировке вопроса; сбалансированность ответов по сложности и правдоподобности. Если более сложные ответы в «базовом» списке находятся в конце, то увеличение КВО будет увеличивать еще и среднюю сложность ответов в ТЗ.

Обычно в КСТ все возможные ответы для ТЗ с ПНО предъявляются тестируемому *одновременно*. Их разная правдоподобность повышает вероятность СПВНПО. При последовательном предъявлении ответов для ТЗ (например, в одном и том же «окошке»): вероятность СПВНПО падает; возможно «досрочное» прерывание процесса предъявления ответов, если ПО опознан как неверный или наоборот.

В принципе возможен и вариант ТЗ, когда набор ответов фиксирован, а вопросы выбираются случайно из некоторого набора для ТЗ. Для реализации это удобно, когда есть единственный ПО для каждого вопроса.

Рассмотрим теперь возможности предоставления *тестируемым альтернатив поведения* при последовательных ответах на ТЗ. (А) Для слишком «сложного» (с точки зрения тестируемого) набора ответов – «нажатие» кнопки «уменьшить количество ответов для выбора». При этом: КСТ «гасит» некоторые ответы; выбор упрощается, но тестируемому начисляются ШБ. (Б) Отказ от ответа на непонятные для тестируемого ТЗ – это целесообразно, если при

неверно выбранных ответах на ТЗ можно «заработать» отрицательные баллы, а при отказе получить «0».

Анализ возможных вариантов оценки ответов на тестовые задания

Оценка ответа для ТЗ с ПНО в простейшем случае – «зачтено» (только при полностью правильном выборе ответов и «не просроченном» времени) или «не зачтено» – в «сумматор» баллов по тесту (S) добавляется либо «0», либо «1». Если для i -го ТЗ есть дескриптор сложности (D_i), то

$$S = S + 0 \text{ или } S = S + D_i \quad (3)$$

Альтернатива – добавление в «сумматор» кроме баллов за правильно выбранные верные ответы (V_V) ШБ за «не выбранные верные ответы» (N_V) и «неверные ответы, выбранные как верные» (V_N).

$$S = S + \lambda_1 V_V - \lambda_2 N_V - \lambda_3 V_N \quad (4)$$

где $\lambda_1, \lambda_2, \lambda_3$ – весовые коэффициенты. Еще вариант: каждый « f »-й ответ по ТЗ индивидуально расценивается в баллах (B_f) – верные с «+», неверные – с «-»; для очередного ТЗ, где выбрано « J » ответов из « F » возможных

$$S = S + \sum_{j=1}^J B_j \quad (5)$$

Если тестируемые могут «реагировать» на каждый из ПНО в виде «да – скорее да – скорее нет – нет», то вместо (5) имеем

$$S = S + \sum_{j=1}^J \mu B_j \quad (6)$$

где: $\mu = 1$ для «да» и «нет»; $\mu = 0,5$ для «скорее да» и «скорее нет».

Для учета времени T_m на ответ по m -му ТЗ при контрольном (C_m) и предельном (L_m) временах целесообразно вычитать из «сумматора» ШБ

$$S = S - P_m ;$$

$$P_m = K_m (T_m - C_m) \text{ при } C_m \leq T_m \leq L_m \quad (7)$$

где коэффициенты K_m «согласовывают» ШБ с баллами за ответы по ТЗ.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Аванесов В. С. Теоретические основы разработки заданий в тестовой форме. – М.: МГТА, 1995.
2. Анохин А. Н. Методы экспертных оценок. – Обнинск: ИАТЭ, 1996. – 148 с.
3. Звонников В. И., Чельщикова М. Б. Современные средства оценивания результатов обучения. – М.: Изд. центр «Academia», 2007. – 224 с.
4. [Электронный ресурс] www.fepo.ru

УДК 681.3.07:005.336.4

*Ю. М. Брумштейн, Т. А. Горюнова, Н. Н. Пугина***АНАЛИЗ МЕТОДОВ И МАТЕМАТИЧЕСКИХ МОДЕЛЕЙ ОБЕСПЕЧЕНИЯ ВУЗА
КАДРАМИ ВЫСШЕЙ КВАЛИФИКАЦИИ****Астраханский государственный университет**
brum2003@mail.ru, tanyshka05@mail.ru, astra8512@mail.ru

Рассмотрены условия работы типичного регионального вуза и роль кадров высшей квалификации (КВК). Проанализированы источники пополнения вузов КВК, возможные подходы к управлению этими процессами. Особое внимание уделено работе аспирантуры и докторантуры. Исследованы каналы кадровых потерь для вуза, указаны меры по снижению таких потерь. Предложены математические модели, описывающие динамику КВК в вузах и подходы к оптимизации управления КВК в условиях ресурсных ограничений.

Ключевые слова: региональные вузы, кадры высшей квалификации, подготовка кадров, кадровые потери, математические модели, оптимизация управления, критерии оптимальности.

*J.M.Brumshteyn, T.A.Gorjunova, N.N.Pugina***THE METHODS AND MATHEMATICAL MODELS ANALYSIS OF UNIVERSITIES
TOP QUALIFICATION PERSONNEL MAINTENANCE****The Astrakhan state university**

Typical regional universities working conditions and top qualification personnel (TQP) the role are considered. Sources of universities TQP replenishment, possible approaches to management of these processes are analyzed. The special attention is given to postgraduate and doctoral studies work. Channels of universities personnel losses are investigated, measures on decrease of such losses are specified. The mathematical models, describing dynamics TQP in universities, and approaches to TQP management optimization under resource restrictions conditions are offered.

Key words: Regional universities, Top qualification personnel, Personnel training, Personnel losses, Mathematical models, Optimization management, Optimality criteria.

Важнейшим условием успешности работы вуза является обеспечение его преподавательскими кадрами (ОПК), и, особенно, лицами с учеными степенями и званиями – кадрами высшей квалификации (КВК), которые являются ключевым компонентом «интеллектуального капитала» вузов [2]. В условиях межвузовской конкуренции, наличия у вузов нормативных и ресурсных ограничений прогнозирование динамики КВК и оптимизация принимаемых решений очень актуальны. В данной статье эта проблематика рассмотрена с учетом специфики региональных вузов (РВ), построены математические модели для оптимизации обеспечения РВ КВК.

**Характеристика условий работы
типичного регионального вуза**

В типичном «бюджетном» РВ сейчас обучается от нескольких тысяч до нескольких десятков тысяч студентов, а количество преподавателей составляет от нескольких сот до нескольких тысяч. У ряда РВ есть внутрирегиональные филиалы и представительства (что позволяет расширить клиентскую базу обучения), в которых работают как «местные» преподаватели, так и «приезжающие» сотрудники головного РВ.

Источники финансирования деятельности РВ: средства федерального бюджета на подготовку студентов и аспирантов; платные образовательные услуги РВ; средства, получаемые по федеральным целевым программам; средства грантов; спонсорская помощь и т. д. В своей деятельности РВ конкурируют за студентов (бюджетных и коммерческих): с другими РВ; региональными филиалами иногородних вузов (прежде всего московских); в меньшей мере – с головными иногородними вузами. В рамках расширения Болонского процесса в России вероятно увеличится конкуренция РВ с зарубежными вузами, особенно в отношении магистрантов.

В аспирантуру РУ поступают в основном их же выпускники. Варианты видов обучения: коммерческое; бюджетное (очное и заочное); на условиях «возмещения расходов» обучающимся путем «отработки» в РВ после защиты диссертации. Важный фактор конкурентоспособности аспирантур – наличие при РУ советов по защите диссертаций.

В своей работе РВ конкурируют за КВК с: другими РВ в том же населенном пункте (НП); филиалами иногородних вузов в НП (они стараются привлечь КВК-совместителей из РУ, т. к. обычно не могут «удержать» их в качестве штатных сотрудников); другими организация-

ми (органами госуправления, коммерческими структурами и пр.); в значительно меньшей степени – с головными иногородними вузами России; редко – с зарубежными вузами (в т. ч. из-за низкой трудовой мобильности КВК).

Факторы конкурентной привлекательности РВ с точки зрения преподавателей: уровень зарплат и иных выплат; условия труда и особенности контингента студентов; лабораторная база; перспективы карьерного и научного роста; возможности получения командировок и стажировок (включая зарубежные), грантов и пр. При этом оценка большинства факторов является субъективной и в ряде случаев носит вероятностный характер.

Предполагается изменение схемы финансирования бюджетной сферы России, в т. ч. и вузов. Они смогут выбрать одну из трех форм: (а) казенные; (б) бюджетные; (в) автономные учреждения. Очевидно, что эти изменения могут привести к усилению межвузовской конкуренции, сделать более актуальными вопросы ОПК РВ. Два основных направления ОПК РВ (пополнение преподавателями, особенно КВК и уменьшение оттока их из вузов) рассматриваются в данной статье подробно.

Основные источники и методы управления пополнением РВ КВК

Вузы (особенно РУ) стремятся увеличить долю КВК до нормативных требований МинобрНауки или выше. Решение этой задачи может достигаться путем: (а) перевода лиц, не имеющих ученых степеней на работу по гражданско-правовым договорам; (б) привлечения на штатную работу в конкретный РВ лиц «со степенями», работающих пока вне его; (в) то же – на работу по совместительству; (г) подготовки КВК через аспирантуру и докторантуру в самих РВ (научными руководителями могут быть штатные сотрудники РВ и лица, приглашенные со стороны); (д) подготовки КВК через аспирантуру и докторантуру иных вузов, академических институтов, прикладных НИИ; (е) подготовки КВК через форму «соискательства».

Вариант «а» позволяет быстро добиться улучшения показателя «остепененности» для РВ, но, снижает привлекательность РВ как места работы для «не остепененных» сотрудников. Поэтому вероятность их увольнения становится выше по сравнению с «работой в штате».

Вариант «б» требует серьезной мотивации перехода, причем вероятность его реализации

тем выше, чем больше «преимущества» на новом месте работы по сравнению с текущим. Такой переток кадров для РВ имеет место в основном внутри НИ, т. к. переезд в другой НИ сопряжен со значительными финансовыми расходами, разрывом привычных связей и пр. При этом предоставление вузами жилья принимаемым на работу КВК для РВ стало редким исключением. Вариант «в» для вузов реализовать проще чем «б», но совместители обычно менее удобны для РВ, в т. ч. из-за ограничений в их «окнах доступности». Кроме того, совместители (особенно из вузов-конкурентов) могут представлять потенциальную опасность для РВ в отношении «утечек» коммерческих секретов, использования объектов интеллектуальной собственности (включая методические и программные разработки), тематики и содержания заявок на гранты, использования результатов НИР и пр. Вариант «г» является для РВ традиционным средством, но ресурсоемок, требует длительного времени для реализации, а положительные результаты носят вероятностный характер. Вариант «д» используется в основном, когда кадры по соответствующему направлению не могут быть подготовлены непосредственно в РВ, в т. ч. из-за отсутствия/нехватки научных руководителей, лабораторной базы и пр. На него не очень охотно идут и сами обучающиеся, т. к. постоянное проживание при очном обучении в других НИ может быть слишком дорогостоящим. С другой стороны, при заочном обучении поддержка связи с научным руководителем через Интернет плюс поездки к нему не всегда эффективны. Лиц, готовящих диссертации в порядке «соискательства» (пункт «е»), мы учитывать не будем, т. к. эта форма стала редкой для многих РВ.

Каналы потерь РВ в отношении КВК и методы борьбы с ними

Основные каналы (причины) потерь КВК: уход преподавателей на пенсию с прекращением работы в вузе; смертность от заболеваний, дорожно-транспортных происшествий, несчастных случаев до достижения пенсионного возраста; переход на штатную работу в другие организации в пределах НИ, в т. ч. конкурирующие вузы. Для совместителей – отказ от работы в РВ по совместительству из-за различных причин. «Временные» кадровые потери РВ могут быть связаны с: длительной нетрудоспособностью преподавателей; уходом их в декретные

отпуска и пр. «Частичные потери» РВ в отношении КВК могут быть связаны с: получением преподавателями инвалидности; общим снижением эффективности работы в пожилом возрасте; активной работой по совместительству в других вузах.

Меры борьбы с потерями преподавательских кадров (особенно КВК): обеспечение высокого уровня зарплат и комфортных условий работы, перспектив научной деятельности; улучшение условий труда и отдыха; создание благоприятного морально-психологического климата в коллективе и пр. Закрепление преподавателей (в т. ч. КВК) путем предоставления жилья сейчас уже редко, т. к. существующие служебные квартиры обычно заняты и освобождаются редко, а на новое жилье у РВ обычно нет средств.

Математические модели для оценки динамики КВК в вузах

Для простоты ниже мы будем рассматривать только детерминированные постановки «непрерывных» во времени задач и считать, что они описываются обыкновенными дифференциальными уравнения (ОДУ) [3].

Пусть в рассматриваемом вузе в текущий момент времени в очной аспирантуре обучаются $N^{(O)}$ аспирантов, а в заочной – $N^{(Z)}$. Если эмпирические вероятности защит диссертаций в течение года на одного обучающегося для этих контингентов составляют $V_K^{(O)}$ и $V_K^{(Z)}$, то оценка скорости *прироста* числа кандидатов наук в вузе может быть дана как

$$dK^{(e1)} / dt = \lambda_K (N^{(O)}V_K^{(O)} + N^{(Z)}V_K^{(Z)}) \quad (1)$$

где: λ_K – учитывает долю лиц, которые после защиты остаются работать в вузе или переходят на работу в него ($1 \geq \lambda_K \geq 0$); индекс «e1» в левой части (1) соответствует приросту «1-го типа». Формула (1) – упрощенная, т. к. не учитываются эффекты «запаздывания» защит по отношению к моментам обучения (поэтому (1) корректна, лишь если $N^{(O)}$ и $N^{(Z)}$ не слишком сильно меняются во времени); различия между «процентами защищающихся» по разным специальностям подготовки, для разных условий обучения и пр. (учет эффектов «запаздывания» значительно усложняет постановку задач и будет рассмотрен в отдельной работе). Верхнюю границу для $V_K^{(O)}$ можно принять как 0.33 (исходя из трех лет обучения), а для $V_K^{(Z)}$ – 0.25

(4 года обучения). Фактические значения $V_K^{(O)}$, $V_K^{(Z)}$ ниже, в т. к. до защиты «доходят» не все лица даже из числа тех, кто завершил обучение в аспирантуре.

Динамика количества лиц, обучающихся в аспирантуре РВ определяется, балансом притока P (прием,) и оттока Q (завершение учебы, отчисление – в т. ч. и в связи с защитой диссертации).

$$dN^{(O)} / dt = P_{(O)} - Q_{(O)}, \quad dN^{(Z)} / dt = P_{(Z)} - Q_{(Z)} \quad (2)$$

При этом приток в аспирантуру зависит от времени ($P_{(O)} = P_{(O)}(t)$, $P_{(Z)} = P_{(Z)}(t)$), а отток еще и от объема текущего контингента – поэтому $Q_{(O)} = Q_{(O)}(t) * N^{(O)}(t)$; $Q_{(Z)} = Q_{(Z)}(t) * N^{(Z)}(t)$. Переход в РВ на штатную работу кандидатов наук «извне РВ» определяется разницей между условиями работы для «кандидатов» в вузе $U^{(K,vuz)}$ и вне его $U^{(K,out)}$

$$dK^{(e2)} / dt = f_1(U^{(K,vuz)} - U^{(K,out)}) \quad (3)$$

где f_1 описывает нелинейную зависимость. Для прироста числа «докторов» наук в РВ могут быть предложены аналогичные формулы

$$dD^{(e1)} / dt = \lambda_D (M^{(O)}V_D^{(O)} + M^{(S)}V_D^{(S)}) \quad (4)$$

где: $M^{(O)}$, $M^{(S)}$ – количества лиц обучающихся в докторантуре и подготавливающих докторские диссертации вне докторантуры

$$dD^{(e2)} / dt = f_2(U^{(D,vuz)} - U^{(D,out)}) \quad (5)$$

В общем случае «условия» в формулах (4), (5) могут отличаться для разных «профилей» КВК, в т. ч. из-за различий спроса и предложения.

Для «номер» вуза по кандидатам наук можно использовать

$$dK^{(e3)} / dt = -\beta_K K - f_3(U^{(vuz)} - U^{(out)}) - \delta * dD^{(e1)} / dt \quad (6)$$

где: K – текущее количество кандидатов наук в вузе. В правой части (6): первый член описывает потери, связанные со смертностью, выходом на пенсию и т. п.; второй – с переходом на работу в другие организации; третий – с переходом «защитившихся» кандидатов наук в ранг доктора наук (δ учитывает возможность того, что не все лица готовившие диссертации в вузе были его штатными сотрудниками в «ранге» кандидата наук). Отметим, что (6) не учитывает возможность защит кандидатских диссертаций лицами, уже имеющими ученую степень кандидата и доктора наук (обычно их доля в РВ менее 1%). В общем случае не равны нулю

значения ни f_2 , ни f_3 , т. е. перетоки КВК идут в обоих направлениях.

Аналогично (6) для «потерь» вуза по докторам наук можно принять

$$dD^{(63)} / dt = -\beta_D D - f_4 (U^{(D,viz)} - U^{(D,out)}). \quad (7)$$

причем из-за большего возраста докторов наук обычно $\beta_D > \beta_K$.

Таким образом, в простейшем случае динамику количества КВК в вузе можно описать системой дифференциальных уравнений

$$\begin{cases} dK / dt = dK^{(61)} / dt + dK^{(62)} / dt + dK^{(63)} / dt \\ dD / dt = dD^{(61)} / dt + dD^{(62)} / dt + dD^{(63)} / dt \end{cases}, \quad (8)$$

дополненной начальными условиями для количеств КВК

$$K_{t=0} = K_0; \quad D_{t=0} = D_0 \quad (9)$$

Формулы (8),(9) позволяют при различных параметрах численно промоделировать ожидаемую динамику изменения количества КВК с использованием явных разностных схем по времени, т. к. ОДУ не «жесткие» [3].

Результаты моделирования важны, прежде всего, с позиций «управления вузом». Такое «управление» с математической точки зрения сводится, в первую очередь, к распределению финансовых средств (Z) по совокупности Y направлений за X периодов времени. Опишем это распределение матрицей $[Z_{x,y}]_{x=1...X; y=1...Y}$ с заданными ограничениями

$$\begin{cases} \left\{ \left(Z_x^* = \sum_{y=1}^Y Z_{x,y} \right) \leq G_x^* \right\}_{x=1...X}; \\ (Z^{**} = \sum_{x=1}^X Z_x^*) \leq G^{**} \end{cases} \quad (10)$$

Для перехода от «затрат» (Z) к коэффициентам ОДУ (примем, что их количество равно ε) необходимо задание прямоугольной матрицы «чувствительности» коэффициентов в (1)...(7) к величинам затратам по отдельным направлениям ($[W]$) с размерностью $\varepsilon * Y$. Оценить элементы $[W]$ можно: экспертно [1]; ограниченно – в рамках «калибровки» задач. Простейший вариант для $[W_{i,j}]$, относящийся к периоду « α »

$$[W_{i,j}]_x = \Phi_1 \{Z_x^*\} \quad (11)$$

где Φ_1 означает некоторую функцию. Однако возможен и вариант с «последствием», например по

$$[W_{i,j}]_x = \xi_1 [W_{i,j}]_{x-1} + \xi_2 \Phi_2 \{Z_{x-1}^*\} + \xi_3 \Phi_3 \{Z_x^*\} \quad (12)$$

где ξ_1, ξ_2, ξ_3 – некоторые весовые коэффициенты.

Критерии «оптимальности» управления (КОУ) и ограничения по параметрам для (1)...(7) могут задаваться по-разному. Для КОУ возможны:

$$\begin{aligned} R_1 &= \alpha_1 K_{t_f} + \alpha_2 D_{t_f} \quad \text{или} \\ R_2 &= (\alpha_1 K_{t_f} + \alpha_2 D_{t_f}) / S_{t_f} \end{aligned} \quad (13)$$

где: α_1, α_2 – коэффициенты; S – общее количество штатных сотрудников вуза; t_f – конечный момент времени планирования в будущем. Другой подход – оценка КОУ по «средним значениям» за период от $t = \Theta$ до $t = t_f$ (причем обычно $\Theta > 0$). Аналогом R_2 при этом будет

$$R_{22} = \max \left(\left(\sum_{t=\Theta}^{t_f} (\alpha_1 K_t + \alpha_2 D_t) / S_t \right) / (t_f - \Theta + 1) \right) \quad (14)$$

Ограничениями на параметры могут быть, например, нормативные требования МинОбр-Науки по долям КВК для кандидатов и докторов наук ($G_i^{(K)}; G_i^{(D)}$) с учетом категории РВ (университет, академия, институт)

$$\{(K_i / S_i) \geq G_i^{(K)}\}_{i=t_0...t_f}; \{(D_i / S_i) \geq G_i^{(D)}\}_{i=t_0...t_f} \quad (15)$$

Возможные оптимизационные задачи для распределения расходов (нахождение $[Z_{x,y}]$): максимизация одного из R_* при фиксированной Z^{**} и ограничениях (15); минимизация Z^{**} при ограничениях на минимальные $\{R_1\}_{t=\Theta...t_f}$ или $\{R_2\}_{t=\Theta...t_f}$; минимизация Z^{**} при ограничении на R_3 за период $t = \Theta...t_f$ и др. Оптимизация может быть выполнена различными методами, включая градиентные [3], генетические алгоритмы и пр.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Анохин А. Н. Методы экспертных оценок. – Обнинск: ИАТЭ, 1996. – 148 с.
2. Багов В. П., Селезнев Е. Н., Ступаков В. С. Управление интеллектуальным капиталом. – М.: ИД «Камерон», 2006 – 248 с.
3. Калиткин Н. Н. Численные методы. – М.: Изд. физ-мат. литературы, 1996. – 523 с.

УДК [004+075.8]:61

*Ю. М. Брумштейн, Д. А. Захаров, Ю. Ю. Аксенова***АНАЛИЗ ВОЗМОЖНЫХ ПОДХОДОВ К ОЦЕНКЕ УРОВНЕЙ ОСНАЩЕННОСТИ МЕДОБОРУДОВАНИЕМ СТАЦИОНАРНЫХ МЕДУЧРЕЖДЕНИЙ****Астраханский государственный университет**
brum2003@mail.ru, dmitrizahar@mail.ru, aks-uliana@mail.ru

Работа посвящена проблеме объективной оценки уровней оснащённости медицинским оборудованием стационарных медучреждений. Рассмотрены факторы, влияющие на этот уровень. Предложены качественные методы оценивания. Подробно проанализированы возможные варианты количественных оценок и вопросы их информационного обеспечения.

Ключевые слова: стационарные медицинские учреждения, медицинское оборудование, уровень оснащённости, факторы влияния, методы оценки, информационное обеспечение.

*U. M. Brumshiteyn, D. A. Zaharov, J. J. Aksenova***THE ANALYSIS OF POSSIBLE APPROACHES TO THE ESTIMATION OF MEDICAL EQUIPMENT PROVISION LEVELS FOR STATIONARY MEDICAL INSTITUTIONS****Astrakhan State University**

The work is devoted to the problem of objective estimation of medical equipment provision levels for stationary medical institutions. The factors influencing this level are considered. Qualitative methods for the estimation are offered. Possible variants of quantitative estimations and questions of their information supply are analyzed in detail.

Key words: Stationary Medical Institutions, The Medical Equipment, Provision Level, Influencing Factors, Estimation Methods, Information Supply.

Качество и доступность медицинской помощи в стационарных медучреждениях (СМУ) России в значительной мере определяются их уровнями оснащённости медицинским оборудованием (УОМО) – диагностическим, лечебным, вспомогательным и др. Единообразное понимание термина УОМО фактически отсутствует, а методы его оценки в существующей литературе и нормативных документах отражены слабо. В данной статье авторы ставили себе целями: анализ содержания термина УОМО; рассмотрение существующих и возможных подходов к оценкам УОМО СМУ.

Основные типы СМУ России с точки зрения их финансирования, организации закупок медоборудования, его техобслуживания

Типы СМУ, их функциональная специализация, источники/объемы финансирования значительно влияют на состав, количество и качество МО в них – фактически имеющегося и необходимого. Основные направления классификации СМУ: многопрофильные (универсального характера) и специализированные; организации высокотехнологичной медпомощи (ОВТМ) и «обычные»; бюджетные СМУ (БСМУ) и коммерческие; для БСМУ – административная подчиненность (федеральная, областная, муниципальная); по источникам финансирования. Для большинства БСМУ бюджетное финансирование сейчас «двухканальное» – через фонд обязательного

медицинского страхования (ФОМС) и из соответствующего бюджета – федерального, областного, муниципального, ведомственного. При этом у БСМУ есть серьезные ограничения по направлениям расходования бюджетных средств, в т. ч. в отношении закупок/ремонта медоборудования (МО). Сейчас рассматривается вопрос о переходе бюджетных медучреждений (включая БСМУ) с «двухканального» финансирования на «одноканальное» – только через ФОМС. «Ведомственные» СМУ финансируются из бюджетов ведомств (корпораций) и часто обслуживают лишь сотрудников «своего» ведомства (организации) и членов их семей. Высокий УОМО таких СМУ может быть важным фактором «социального пакета» и, как следствие, конкурентоспособности ведомственных организаций.

Большинство БСМУ получают средства и за счет оказания платных медуслуг – по договорам с организациями и непосредственно от пациентов. Другими источниками получения средств на МО (закупки, ремонт и пр.) или самого МО для БСМУ могут быть: пилотные и целевые программы; спонсорская помощь и др. При этом: оборудование для СМУ в основном (по [4] до 98 %) покупается за счет бюджета и может быть «вопросом престижа местной администрации»; закупки МО часто осуществляются централизованно – на уровне населенных пунктов или регионов; взятие МО в аренду

или лизинг СМУ практически не используется; фактически отсутствует вторичный рынок для «бывшего в употреблении» МО, что нередко тормозит (особенно в ОБТМ) вывод из эксплуатации морально устаревшей или изношенной техники. Судя по сообщениям СМИ [3], принято решение о направлении на нужды здравоохранения в 2011–2012 гг., дополнительно 460 млрд. руб., в т. ч. на «ремонт, реконструкцию и обновление оборудования» медучреждений – 300 млрд. руб. Необходимую сумму предполагается получить за счет увеличения отчислений предприятий от зарплаты в ФОМС с 3,1 % до 5,1 %. Средства предполагается распределять по регионам на основе «конкретных планов ремонта и закупок оборудования для медучреждений» [3]. Это делает проблемы оценки/управления УОМО БСМУ еще более актуальными.

Для коммерческих СМУ УОМО – важный фактор конкурентоспособности. Ими всегда подчеркивается наличие/использование современных технологий, включая «компьютерное обследование организма».

Важнейшую роль в обеспечении УОМО СМУ играет сервисное обслуживание (СО) МО за счет: штатных специалистов СМУ по МО; внешних организаций СО, в т. ч. специализирующихся на МО; внештатных для СМУ специалистов; сервисных центров (СЦ) фирм-производителей (ФП) МО. Отметим еще рост дистанционного (через Интернет) мониторинга технического состояния и условий эксплуатации высокотехнологичного МО (ВТМО) в СМУ со стороны ФП и их СЦ.

Качественная оценка УОМО СМУ

Для этой группы методов преобладает «сравнительное оценивание», что может соответствовать [2] шкалам: «порядка»; «отношений»; «разностей». На практике СМУ часто сравниваются не по всему парку МО, а только по ВТМО или даже по наиболее сложному (лидирующему) МО в СМУ. Медработниками принимаются во внимание функциональные возможности и качественные характеристики такого МО. Для пациентов может быть важнее репутация ФП, которая ими часто оценивается исходя из более распространенных видов изделий той же фирмы. Для СМУ универсального характера лидирующим МО сейчас обычно являются: 3D УЗИ-аппараты; компьютерные томографы (КТ) – рентгеновские, магнито-

резонансные, позитронно-эмиссионные. Однако на всю Россию пока приходится лишь около 1500 КТ [4]. Стоимость лидирующего диагностического ВТМО обычно выше, чем «собственно лечебного».

Другой подход – оценка УОМО конкретного СМУ по «времени отставания» в годах от «эталонных СМУ». Такие оценки часто не вполне объективны – особенно в СМИ, выступлениях политиков и др. В частности это касается сравнений районных или областных Российских больниц с ведущими СМУ (в т. ч. специализированными) г. Москвы, США, Западной Европы – вместо сопоставления со СМУ, находящимися на равном иерархическом уровне и выполняющих аналогичные функции.

Третий подход – сравнение СМУ по «возрасту парка оборудования» с применением «шкалы порядка» [2]. Пациентами СМУ возраст МО оценивается обычно субъективно – по его внешнему виду (в т. ч. соответствию современному дизайну, наличию повреждений и пр.). Отметим, что восприятие внешнего вида зданий СМУ и их помещений, нередко ассоциируется пациентами с возрастом МО.

Количественные оценки УОМО СМУ

Экспертное оценивание [1] может применяться в качественной (с использованием «шкал порядка») и количественной формах. В последнем случае типичный вариант – общая оценка УОМО конкретного СМУ (Q_1) на основе индивидуальных оценок K экспертов

$$Q_1 = \left(\sum_{k=1}^K (Z_k \lambda_k) \right) / \sum_{k=1}^K \lambda_k \quad (1)$$

где: $\{Z_k\}$ – набор оценок экспертов; $\{\lambda_k\}$ – их экспертные коэффициенты (ЭК). Преимущество (1) – простота расчета. Недостаток – методические сложности в рациональном подборе экспертных групп и назначении ЭК.

У экспертов могут быть разные представления о необходимом уровне УОМО СМУ. Поэтому для «нормализации» оценок экспертов как «эталон» (с заданной величиной УОМО) может быть предложено конкретное СМУ.

Если компетентности экспертов в отношении отдельных «параметров» УОМО СМУ значительно различаются, то вместо (1) лучше

$$Q_2^{(j)} = \left(\sum_{k=1}^K (Z_k^{(j)} \lambda_k^{(j)}) \right) / \sum_{k=1}^K \lambda_k^{(j)} \quad (2)$$

где для j -го параметра УОМО и k -го эксперта: $Z_k^{(j)}$ – оценка УОМО; $\lambda_k^{(j)}$ – величины ЭК

(для разных «j» они различны). Для СМУ «в целом»

$$Q_2 = \frac{\sum_{j=1}^J (Q_2^{(j)} \Psi^{(j)})}{\sum_{j=1}^J (\Psi^{(j)})} \quad (3)$$

где $\Psi^{(j)}$ – «степень важности» (весовой коэффициент) для j -го «параметра» при оценке УОМО СМУ. Набор $\{\Psi^{(j)}\}_{j=1...J}$ может: задаваться лицом (организацией), осуществляющей экспертное оценивание; определяться экспертным оцениванием – той же или иной группой экспертов.

Оценка УОМО может быть дана и путем сравнения фактических показателей обеспеченности СМУ МО с нормативными требованиями (НТ). Отметим: НТ в отношении УОМО существуют для отдельных подразделений СМУ и кабинетов [5]; большинство НТ носит рекомендательный характер; в последнее время НТ оформляются, в основном, приказами МинЗдравСоцРазвития; в НТ прописаны минимальные требования к составу МО по функциональному назначению, но обычно без конкретных характеристик; требования НТ различаются в зависимости от «количества коек» в СМУ, врачей в кабинетах и пр. Причины по которым НТ могут не удовлетворяться: недостаток МО; выход МО из строя и его списание; изменения НТ в сторону «усиления требований» и пр. В СМУ МО: часто эксплуатируется до полного физического «износа»; передача МО из областных/городских БСМУ в районные БСМУ и поликлиники не практикуется.

Качественная оценка УОМО СМУ – удовлетворяются ли НТ полностью или нет. Простейшая количественная оценка УОМО (U_1) – сравнение имеющегося МО в СМУ с НТ по функциональным группам (ФГ) МО

$$U_1 = G_+ / (G_+ + G_-); \quad 0 < U_1 \leq 1 \quad (4)$$

где: G_+, G_- – количества ФГ МО, для которых НТ удовлетворяются и нет. При учете «важности» ФГ МО для СМУ обобщение формулы (1) дает

$$U_2 = \frac{\sum_{i=1}^I W_i \beta_i}{\sum_{i=1}^I W_i} \quad 0 < U_2 \leq 1 \quad (5)$$

где: I – количество учитываемых ФГ МО; $\beta_i = 1$ если НТ для ФГ МО удовлетворяются и $\beta_i = 0$ если «нет»; весовые коэффициенты (W_i) для ФГ МО могут быть определены, например, экспертно. Если НТ оговаривают для ФГ МО не только состав оборудования, но и количества их единиц, то

$$U_3 = \frac{\sum_{i=1}^I W_i |K_i^{(f)} - K_i^{(n)}|}{\sum_{i=1}^I W_i};$$

$$W_i = \begin{cases} W_i^+ & \text{при } K_i^{(f)} > K_i^{(n)} \\ W_i^- & \text{при } K_i^{(f)} \leq K_i^{(n)} \end{cases} \quad (6)$$

где $K_i^{(f)}; K_i^{(n)}$ фактическое и нормативное количества МО для i -й группы. Логично считать, что $\{W_i^+ > 0\}, \{W_i^- < 0\}$, причем $\{W_i^+ / W_i^- < 1\}$.

Тогда $U_3 < 0$ говорит о низком УОМО. При учете для СМУ филиалов/подразделений оценка для СМУ «в целом» может быть дана в виде

$$U_4 = \frac{\sum_{j=1}^J (H^{(j)} U_*^{(j)})}{\sum_{j=1}^J H^{(j)}} \quad (7)$$

где: J – количество подразделений/филиалов; $U_*^{(j)}$ – для j -го подразделения/филиала оценивается по одной из формул типа (4)...(6); $H^{(j)}$ – определяет степень «важности» j -го подразделения/филиала (например, исходя из объемов медобслуживания с использованием МО).

Оценки УОМО на основе стоимостных показателей МО достаточно типичны, но имеют серьезные недостатки: в качестве «основных средств» в бухгалтериях СМУ числится только дорогостоящее оборудование; начисление на него «амортизации» осуществляется без учета фактического физического и морального износа стоимости МО, приобретенного в разное время, не подвергаются дисконтированию; затраты на ремонт МО и его техобслуживание не учитываются; после окончания нормативных сроков эксплуатации МО в СМУ часто продолжает применяться (при наличии заключения о его «годности» соответствующей сервисной организации и заключения метрологической организации – для измерительного МО) – в то же время «балансовая стоимость» такого МО уже «нулевая»; в сервисных подразделениях СМУ учет обслуживаемого МО не всегда полон и чаще всего осуществляется «без стоимостей»; специализированные «оценочные фирмы» для получения объективных оценок стоимости МО не привлекаются. Для оценок целесообразно использование относительных стоимостных показателей для МО. В простейшем случае

$$U_5 = \frac{\sum_{m=1}^M B_m}{N} \quad (8)$$

где B_m – текущая (балансовая) стоимость m -й единицы МО (ЕМО); M – количество учитываемых ЕМО. При этом в (8) могут учитываться лишь ЕМО со стоимостями приобретения выше некоторой пороговой. В качестве N при

нормировке могут быть использованы количества: коек в СМУ; медперсонала; врачей; «пролеченных» пациентов; комбинации этих показателей; площади помещений СМУ. Для ОБТМ значения U_5 – высокие.

**Простейшая оценка
на основе среднего «возраста» МО в СМУ**

$$U_6 = (\sum_{m=1}^M T_m) / M \quad (9)$$

где T_m – срок эксплуатации (СЭ) m -й ЕМО. Для (9) может быть целесообразно: «отсечение» МО со стоимостями приобретения ниже пороговой; использование разных формул (9) для групп МО с разной стоимостью приобретения. При учете различий плановых СЭ ЕМО ($T_m^{(nor)}$) целесообразно нормировать возраста каждой из ЕМО на эти сроки. Тогда вместо (6)

$$U_7 = (\sum_{m=1}^M (T_m / T_m^{(nor)})) / M \quad (10)$$

Понятно, что чем старше парк МО, тем U_7 выше. При $U_7 > 1.5-2$ актуальность вопросов замены оборудования в СМУ становится высокой.

Физическое старение оборудования более объективно оценивать не по календарным срокам, а по фактическим продолжительностям работы, например – в часах. При этом может быть использована

$$U_8 = (\sum_{m=1}^M (t_m^* / t_m^{(nor)})) / M ; t_m = \sum_{l=1}^{L_m} t_m^{(l)} \quad (11)$$

где: L_m – количество «сеансов» диагностики/лечения для m -й ЕМО; $t_m^*, t_m^{(nor)}$ – фактическая и нормативная продолжительности работы. Отметим, что (11) не учитывает различия в «интенсивностях» работы МО для разных сеансов/процедур (это важно в основном для лечебного МО).

Итак, сделаем выводы. 1. В настоящее время отсутствуют унифицированные подходы к оценкам УОМО для СМУ. 2. Предложенные качественные и количественные варианты оценок УОМО нуждаются в дальнейшем обсуждении и апробации на конкретном материале. 3. Оценки на основе сравнения фактических показателей с НТ в принципе более объективны, но нуждаются в адекватном информационном обеспечении.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Анохин А. Н. Методы экспертных оценок. Обнинск: ИАТЭ, 1996. – 148 с.
2. Анфилатов В. С., Емельянов А. А., Кукушкин А. А. Системный анализ в управлении. – М.: Финансы и статистика, 2006. – 368 с.
3. Батенёва Т. Здравоохранение идет на поправку. За два года больницы и поликлиники страны получают 460 млрд. рублей на перевооружение и модернизацию // Известия, № 71/28086 от 22.04.2010, С.1–2.
4. Волков К. Томография покажет // Известия, 19.05.2010, № 88 (28103), С. 11.
5. Организация физиотерапевтической помощи в лечебных учреждениях: Методическое пособие / Под ред. Г. Н. Пономаренко – изд. 2-е. – СПб. 2009 – 128 с.

УДК 004.054, 004.4'233

А. М. Дворянкин, Е. В. Кривенко

ОБЗОР СВОБОДНО РАСПРОСТРАНЯЕМЫХ СРЕДСТВ АВТОМАТИЗИРОВАННОГО ТЕСТИРОВАНИЯ ВЕБ-ПРИЛОЖЕНИЙ

Волгоградский государственный технический университет
dvam@vstu.ru, ieugene@mail.ru

Рассматриваются свободно распространяемые средства автоматизированного тестирования веб-приложений.
Ключевые слова: средства автоматизированного тестирования, веб-приложения, Selenium, WATIR, CubicTest.

А. М. Dvoryankin, E. V. Krivenko

THE REVIEW OF COMPUTER-AIDED TESTING FREE SOFTWARE OF WEB APPLICATIONS

Volgograd State Technical University

The review of computer-aided testing free software of web applications.
Key words: computer-aided testing software, web applications, Selenium, WATIR, CubicTest.

Средства автоматизированного тестирования веб-приложений представляют собой набор инструментальных средств для быстрой разработки автоматизированных тестов и обеспечивают [1]:

- Проведение многократного регрессионного тестирования;
- Быструю обратную связь с разработчиками во время процесса разработки;

- Практически неограниченные итерации выполнения тестового примера;
- Настраиваемые сообщения об обнаруженных ошибках приложений;
- Поддержка различных методологий разработки;
- Правильность документирования тестовых примеров;
- Обнаружение ошибок, пропущенных при ручном тестировании.

Преимущества средств автоматизированного тестирования очевидны – высокая скорость выполнения, повторяемость, большая функциональность, возможность тестирования уже имеющегося в сети ресурса.

Рассмотрим наиболее известные и популярные среди тестировщиков веб-приложений в силу хорошей документированности, постоянного обновления и поддержки разработчиками свободно распространяемые средства автоматизированного тестирования.

Selenium [1].

Selenium представляет собой набор инструментальных средств автоматизированного тестирования и состоит из 3 инструментальных средств:

- Selenium-IDE;
- Selenium-RC (Remote Control);
- Selenium-Grid.

Selenium-IDE представляет собой интегрированную среду разработки для построения тестов. Она является плагином к браузеру Firefox и предоставляет удобный интерфейс для разработки и запуска тестов.

Selenium-RC (Remote Control) представляет собой автоматизированный сервер тестирования веб-приложений, написанных на сценарных языках программирования для большей гибкости и расширяемости в разработке тестов логики функционирования приложения. Selenium-RC предоставляет API и библиотеки для каждого поддерживаемого языка: HTML, Java, C#, Perl, PHP, Python, и Ruby. Это позволяет использовать Selenium-RC с сценарными языками программирования высокого уровня для разработки тестов автоматизированного тестирования, интегрированных с средой выполнения проекта.

Selenium-Grid предоставляет решения Selenium-RC для больших программных комплексов или комплексов, работающих в нескольких операционных средах.

Возможности языка разработки тестов Selenise для Selenium-IDE:

- Верификация элементов страницы.
- Определение местоположения элементов.
- Проверка совпадения текста с шаблоном.

Наличие операторов управления позволяет управлять выполнением теста.

Команды хранения позволяют сохранять переменные и проверяемые параметры, также существует возможность интеграции Selenise и JavaScript.

Selenium-RC – серверное решение для тестов, которые требуют более сложного подхода, чем линейное выполнение. Selenium-RC использует все возможности языков программирования для написания более сложных тестов, таких как работа с файловой системой, системами управления базами данных, интернет-протоколами.

Selenium-RC состоит из 2 компонент:

- Сервер, который запускает и останавливает браузеры, интерпретирует и запускает команды на языке Selenise в тестируемой программе, а также работает как прокси-сервер, перехватывая и проверяя HTTP-запросы между браузером и тестируемым приложением. Сервер Selenium принимает команды от тестируемой программы, и возвращает назад в программу результаты выполнения тестов. Сервер формирует Selenium Core и автоматически встраивает его в браузер (Selenium Core – программа на JavaScript, которая интерпретирует и выполняет команды Selenise).

Сервер принимает команды Selenise от тестируемой программы используя простые запросы HTTP GET/POST. Сервер написан на языке Java и не требует какой-либо специальной установки.

- Библиотека стороны клиента, которая обеспечивает взаимодействие между каждым языком программирования и сервером Selenium-RC. Она обеспечивает программную поддержку, которая позволяет выполнять команды Selenium из тестовых программ. Существуют различные библиотеки для каждого из поддерживаемых языков программирования.

Библиотека берет команды Selenise и посылает их серверу для выполнения специфических действий или тестирования приложения. Она также получает результат выполнения команды и посылает их обратно в тестовую программу. Программа может получать и сохранять результаты в переменных и сообщать об успешном или ошибочном завершении.

Имеются клиентские библиотеки для поддержки языков Java, Python и языков платформы .NET.

2. Библиотека для интерпретатора Ruby с открытым кодом WATIR (Web Application Testing in Ruby) [2,3].

Библиотека WATIR понимает структуру веб-страницы и позволяет получить доступ к ее элементам. Библиотека WATIR используется для написания сценариев тестирования веб-страниц. С помощью набора таких сценариев можно автоматизировать процесс тестирования веб-приложений. Ruby поддерживает OLE, что дает возможность автоматизировать работу браузера Microsoft Internet Explorer (существуют также версии библиотеки для работы с другими браузерами, такими как Firefox и Safari и операционными системами Windows, MacOS и Linux). В отличие от инструментов тестирования, основанных на протоколе HTTP, которые симулируют запросы браузера, WATIR симулирует работу пользователя с браузером с помощью протокола OLE, который создан на основе архитектуры COM.

Возможности языка библиотеки WATIR:

- Управление работой браузера – открытие браузера с загрузкой нужного URL и закрытие браузера.

- Проверка доступа к элементам на странице.
- Работа с элементами управления.
- Проверка контента.

3. Графическая среда CubicTest [4].

CubicTest предназначен для разработки тестов веб-приложений с использованием платформ Selenium и Watir и представляет собой графическую среду, реализованную в виде плагина к Eclipse. Использование платформ Selenium и Watir позволяет выполнять тесты непосредственно в браузерах (Internet Explorer, Firefox, Opera, Safari), что дает точную картину результатов выполнения операций.

Тесты в CubicTest состоят из последовательностей «страница/состояние» и переходов между ними.

Тест может быть использован для:

- Тестирования требований;
- Описания требований;
- Сообщения об ошибках;
- Запросов на изменения.

Каждая страница/состояние могут иметь несколько элементов, которые рассматриваются как проверки, которые будут выполнены на странице. Примеры элементов страницы вклю-

чают текст, ссылки, текстовые поля, кнопки и т. д. Также эти элементы могут быть задействованы пользователем (быть объектами действия пользователя). Элементы страниц используются для проверок на наличие элементов на странице/состоянии и как основа для пользовательских взаимодействий.

Каждый элемент страницы имеет набор идентификаторов, которые определяют этот элемент на странице. Также элементы имеют набор пользовательских взаимодействий, которые могут быть применены к ним.

Контексты позволяют легко идентифицировать элементы путем указания соседних элементов и используются, когда элементы трудно идентифицировать самостоятельно.

Элемент страницы или контекст может быть идентифицирован по набору идентификаторов в HTML коде.

CubicTest поддерживает множественные идентификаторы, связанные между собой логическим «И». Элемент также может быть помещен в контекст, чтобы идентифицировать его (или же он может быть в контексте и иметь несколько идентификаторов).

Веб-приложение состоит из множества страниц/состояний. В CubicTest страница/состояние – стабильное представление приложения, с которым может взаимодействовать пользователь.

Переходы с одной страницы на другую выполняет так называемый «тестовый пользователь». Переход может состоять из нескольких пользовательских действий с элементами страницы. Все пользовательские действия формируют последующий переход.

В то время как страница/состояние представляют собой стабильные состояния приложения, переходы определяют изменение страницы/состояния при переходе к следующей странице или следующему состоянию.

Переход может иметь одно или больше пользовательских взаимодействий, которые последовательно выполняются для получения новой страницы/состояния. Пользовательские взаимодействия могут быть следующими:

- Заполнение текстового поля;
- Клик на ссылку;
- Клик на кнопку;
- Помещение курсора мыши над элементом;
- Выбор опции в SELECT.

Помимо возможностей, доступных в Selenium-IDE и Watir, CubicTest позволяет не только вы-

полнять, но и производить экспорт тестов для выполнения в данных средах тестирования.

Рассмотренные средства тестирования веб-приложений обладают следующими свойствами:

1. Наличие языка разработки тестов.
2. Интеграция в браузеры.

Видна также иерархия в технологиях взаимодействия с веб-приложениями и программами стороны клиента:

1. Интеграция в браузер (Selenium-IDE и Watir для IE) – в этом случае имеет место взаимодействие между программой стороны клиента и тестируемым веб-приложением по протоколу HTTP.

2. Интеграция в операционную систему (Watir) – установленная библиотека к языку программирования позволяет управлять не только тестированием, но и использует средства языка программирования для управления браузером.

3. Клиент-серверное приложение (Selenium-RC) – тесты выполняются на сервере и результат выполнения посылается в программу стороны клиента.

Недостатки рассмотренных средств автоматизированного тестирования:

1. Невозможность работы в полностью автоматическом режиме на веб-ресурсах с авторизацией.

2. Некорректность работы на некоторых веб-ресурсах, использующих cookie, так как данные программы не могут корректно эмулировать браузер, работающий с cookie.

3. Кроме Selenium-RC остальные средства представляют собой программы стороны клиента, что заведомо снижает функциональность средства автоматизированного тестирования, так как присутствуют две различные операционные среды – стороны клиента и стороны сервера.

Учитывая вышеизложенное, можно сказать о том, что выбор каждого из описанных средств зависит от сложности проекта, квалификации и пользовательских предпочтений тестировщика, а свободное распространение данных средств служит залогом их популярности.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Selenium Documentation. Release 1.0.
[http:// http://seleniumhq.org/docs/](http://seleniumhq.org/docs/)
2. Web Application Testing in Ruby. Documentation.
<http://watir.com/documentation/>
3. WATIR
<http://ru.wikipedia.org/wiki/WATIR>
4. CubicTest Wiki.
<http://boss.bekk.no/display/BOSS/CubicTest>

УДК 004.8

И. Г. Жукова, М. Б. Кульцова, А. В. Аникин

МОДЕЛИ И ИНСТРУМЕНТАЛЬНЫЕ СРЕДСТВА ИНТЕГРАЦИИ РАСПРЕДЕЛЕННЫХ РЕСУРСОВ В ОТКРЫТЫХ ОБРАЗОВАТЕЛЬНЫХ СЕТЯХ

Волгоградский государственный технический университет
dvam@vstu.ru

В статье рассмотрены онтологические модели и инструментальные средства для аннотирования электронных образовательных ресурсов и создания персонализированных образовательных коллекций.

Ключевые слова: Semantic Web, E-Learning, дистанционное обучение.

I. G. Zhukova, M. B. Kultsova, A. V. Anikin

MODELS AND SOFTWARE TOOLS FOR INTEGRATION OF THE DISTRIBUTED RESOURCES IN THE EDUCATIONAL NETWORKS

Volgograd State Technical University

In this paper, proposed ontology-based models and software tools for annotation of electronic educational resources and creating of the personal educational collections.

Key words: Semantic Web, E-Learning, distance learning.

Модернизация высшего образования в России в рамках Болонского процесса ставит задачи перехода на кредитно-модульную систему обучения с использованием компетентностного подхода, обеспечивающую академическую мо-

бильность студентов, международное признание дипломов (единое приложение к диплому), сокращение разрыва между профессиональной подготовкой выпускников вузов и реальными требованиями рынка труда, персонафикацию

процесса обучения. При этом необходима разработка соответствующих образовательных программ (учебных планов, программ дисциплин, индивидуальных учебных планов), а также обеспечение учебного процесса информационно-образовательными ресурсами и их эффективного использования.

В качестве таких ресурсов используются как традиционные, так и электронные образовательные ресурсы. Традиционные ресурсы, используемые в образовательном процессе, наряду с их достоинствами обладают рядом недостатков, среди которых: бумажный носитель информации; большой срок издания и распространения; быстрое устаревание информации, особенно в технических областях; невозможность организации эффективного поиска информации; ограниченность средств представления информации. Хранилищами таких ресурсов являются традиционные библиотечные фонды, базирующиеся на использовании библиотечно-библиографических классификаций (УДК, ББК). При использовании таких ресурсов затруднены адаптация материалов под индивидуальные характеристики обучаемых, изучение материалов по персональным траекториям ввиду отсутствия вариативных компонент и средств описания характеристик адаптации таких ресурсов.

Согласно ГОСТ Р 52653-2006, электронные образовательные ресурсы (ЭОР) – это образовательные ресурсы, представленные в электронно-цифровой форме и включающие в себя структуру, предметное содержание и метаданные о них. Такие ресурсы, помимо преимуществ традиционных, отличаются компактностью хранения, мобильностью, тиражируемостью, возможностью оперативного внесения изменений и дополнений, использования при автоматизации образовательного процесса и др., что позволяет решить многие проблемы, возникающие при использовании традиционных ресурсов.

ЭОР могут быть простыми и составными, которые могут быть декомпозированы на элементы, такие как информационные фрагменты (фрагменты текста, иллюстрации и т. п.), объекты данных (совокупность информационных фрагментов), образовательные объекты более низких уровней (образованные объектами данных и логическими связями и имеющие образовательные цели) [1]. Модульность ЭОР на основе их декомпозиции позволяет формиро-

вать на их базе более сложные структуры – электронные образовательные коллекции. Под электронной образовательной коллекцией будем понимать множество ЭОР, обладающих однотипными формальными признаками (семантика, образовательные цели обучаемого, его требования к представлению образовательного контента, уровня подготовки в данной предметной области и др.), и логических связей между ними, определяющих рекомендуемую стратегию обучения. Таким образом, электронная образовательная коллекция должна быть построена на основе ЭОР; учитывать требования образовательной программы по подготовке по специальности, образовательные цели обучаемого, параметры адаптации; задавать рекомендуемый порядок изучения входящих в нее ЭОР; адаптироваться к изменяющимся в процессе обучения характеристикам обучаемого. Требования к подготовке специалистов и содержанию образовательного процесса при получении высшего профессионального образования регламентируются федеральными государственными образовательными стандартами (ФГОС ВПО) по соответствующим направлениям подготовки и детализируются в рабочих программах дисциплин (учебных курсов). Учебный курс является одним из базовых понятий в образовательном процессе и должен поддерживаться учебно-методическим комплексом дисциплины (УМКД) – изданием, объединяющим программно-технические и учебно-методические средства обучения, выполняющие определенные дидактические задачи и регламентирующие основные виды учебной деятельности, включающим рабочую программу дисциплины, конспекты лекций, учебно-методическое обеспечение лабораторных работ, практических занятий, курсового проектирования, контрольные задания и вопросы, экзаменационные билеты и другие методические ресурсы.

Типовой процесс использования ЭОР в качестве таких ресурсов в образовательном процессе с точки зрения обучаемого может быть представлен как итеративный процесс поиска, изучения ресурсов и применения полученных знаний для формирования умений и навыков. Условием окончания изучения отдельного курса является достижение знаний, умений, навыков, компетенций обучаемого уровня, определенного рабочей программой дисциплины, разработанной на основе требований образовательного стандарта. Ресурсы, изучаемые на

каждом этапе, регламентируются требованиями и ограничениями рабочей программы, целями и предпочтениями обучаемого, а минимально необходимые для изучения каждого ресурса знания, умения, компетенции должны соответствовать текущим характеристикам обучаемого.

Таким образом, при поиске ЭОР должны учитываться как характеристики предметной области (в частности, описываемые в рабочих программах дисциплин), так и характеристики обучаемого. Для обеспечения таких возможностей ЭОР должны быть соответствующим образом описаны. Процесс подготовки преподавателем ЭОР для использования в учебном процессе состоит из этапов подготовки предметного содержания ЭОР, аннотирования их метаданными и размещения ЭОР для дальнейшего использования. Для описания предметной области, ограничений образовательных стандартов и рабочих программ, характеристик обучаемых, ЭОР и персонализированных коллекций разработаны соответствующие модели на основе онтологии и метаонтология. Для описания онтологий выбран язык OWL, для описания правил логического вывода – язык SWRL.

В частности, метаонтология включает в себя концепты «человек», «обучаемый», «преподаватель», «ЭР», «ЭОР» и др., а также концепт «компетенция», связанный отношениями и описываемый концептами «знание» и «умение». Данный концепт используется в отношениях в онтологии ЭОР (включающей описание ЭОР и позволяющей описывать в том числе и образовательный стандарт, рабочую программу дисциплины как особый тип ЭОР) для описания входных требований для изучения отдельных ЭОР, получаемых в процессе изучения ЭОР знаний, требований образовательных документов к подготовке специалистов, а также в онтологии обучаемого для описания его целевого и текущего состояний. Концепт «знания» имеет отношения с концептами предметной области и позволяет описать их подмножество. Модель описания предметной области на основе онтологии представлена как:

$$DD_o = \langle C_{DD}, R_{DD}, Inst_{DD} \rangle,$$

где C_{DD} – множество концептов предметной области, организованных иерархически;

R_{DD} – множество отношений, описывающих отношения между концептами, а также язык представления;

$Inst_{DD}$ – множество концептов предметной области, представленных на естественных языках.

Модель электронного информационного ресурса описана в виде:

$$ER_o = \langle C_{ER}, R_{ER}, Inst_{ER} \rangle,$$

где C_{ER} – множество концептов онтологической модели электронного образовательного ресурса, представляющих собой типы ресурсов,

R_{ER} – множество отношений, задающих тип ЭОР, образовательную функцию ресурса, язык представления, уровень сложности изложения материала, входные требования для изучения ресурса и результаты изучения ресурса в виде компетентностной оценки, информация, идентифицирующая ресурс;

$Inst_{ER}$ – множество ресурсов, образующих репозиторий ресурсов данной предметной области.

Онтологическая модель обучаемого имеет вид:

$$LM_o = \langle C_{LM}, R_{LM}, Inst_{LM} \rangle,$$

где C_{LM} – концепты онтологической модели пользователя (информация, идентифицирующая обучаемого; уровень подготовки в предметной области и др.);

R_{LM} – множество отношений, задающих уровень подготовки по предметной области, язык, текущее и целевое поле знаний обучаемого как подмножество элементов предметной онтологии;

$Inst_{LM}$ – множество описаний различных обучаемых.

Реализована предметная онтология для дисциплины «Программирование на языке высокого уровня», описаны ЭОР для данной предметной области. Разработаны правила на языке SWRL для поиска ЭОР и их интеграции в персонализированные коллекции. В частности, правила для многокритериального поиска информационных ресурсов по целевому полю знаний пользователя с учетом синонимии терминологии предметной области, уровню сложности изложения материала и языку имеют вид:

```
hasResourceByLanguage(?c,?r) ^ hasResourceByExpirience(?c,?r) ^ hasResourceByConcepts(?c,?r)
→ hasResource(?c,?r); hasUser(?c,?user) ^
LM:hasObjectiveKnowledge(?u,?concept1) ^
ER:hasConcept(?r,?concept2) ^ concept:isSynonymOf(?concept1,?n1) ^ concept:isSynonymOf(?concept2,?n2) ^ sameAs(?n1,?n2) → hasResourceByConcepts(?c,?r);
```


УДК 004.896

*И. В. Ковалев***ПРОГРАММНЫЙ КОМПЛЕКС АВТОМАТИЗАЦИИ ЭКОЛОГИЧЕСКОГО МОНИТОРИНГА
ЗАГРЯЗНЕНИЯ АТМОСФЕРЫ****МГТУ им. Н. Э. Баумана (Калужский филиал)**
ikovalew@gmail.com

Предложен программный комплекс автоматизации экологического мониторинга загрязнения атмосферы, основанный на статистической обработке данных расчетного мониторинга и на интеллектуальном анализе данных инструментального мониторинга. Описанный программный комплекс позволяет решить задачу повышения эффективности принятия административных решений в области экологической безопасности.

Ключевые слова: экологический мониторинг, автоматизация, интеллектуальный анализ данных, нейронные сети, генетические алгоритмы.

*I. V. Kovalev***PROGRAM COMPLEX OF AUTOMATION ECOLOGICAL MONITORING OF POLLUTION
OF ATMOSPHERE****Moscow State Technical University n.a. N.E. Bauman (Kaluga Branch)**

The program complex of automation of ecological monitoring of pollution of the atmosphere, based on statistical data processing of settlement monitoring and on the intelligence data analysis of tool monitoring is offered. The described program complex allows to solve a problem of increase of efficiency of acceptance of administrative decisions in the field of ecological safety.

Key words: ecological monitoring, automation, intelligence data analysis, neural networks, genetic algorithms.

Специфической особенностью большинства городов средней полосы России является то, что они имеют исторически сложившуюся плотную жилую застройку, в которую включено большинство промышленных предприятий, расположенных в черте городов. В подобной ситуации для получения полной картины состояния атмосферного воздуха в городе крайне важно учитывать взаимное влияние выбросов предприятий, а также городского транспорта. Разумным выходом представляется разработка и ведение автоматизированной системы расчетного (прогнозного) и фактического (инструментального) мониторинга атмосферного воздуха, основанного на базе данных выбросов предприятий и транспорта, проведение сводных расчетов рассеивания вредных веществ и инструментальном мониторинге загрязнения атмосферы.

Для решения задачи автоматизации экологического мониторинга загрязнения атмосферы разработан программно – аналитический комплекс (ПАК) «Воздух – Город». Укрупнено совокупность задач, решаемых ПАК, можно разделить на два класса:

Статистическая обработка расчетных данных. Результатом обработки является пакет нормативных документов (разрешающих, предписывающих, информационных и т. д.);

Интеллектуальная обработка данных фактического мониторинга. Данная обработка

необходима как для сопоставления расчетных данных с реальными замерами, так и для прогнозирования будущего развития обстановки в области загрязнения атмосферы.

Рассмотрим эти два класса задач и их реализацию в ПАК «Воздух – Город».

Статистическая обработка расчетных данных

ПАК «Воздух – Город» позволяет реализовать следующие приоритетные цели системы управления экологической безопасностью города:

минимизировать количество выбросов в атмосферу, производимых природопользователями; повысить эффективность управления экологической безопасностью города с целью выполнения требований действующего законодательства.

Функции, реализуемые комплексом, сводятся к следующему:

формирование и ведение базы данных выбросов предприятий и городского автотранспорта на основе инвентаризаций источников вредных веществ, выбрасываемых предприятиями, а также ежегодного исследования транспортных потоков на улицах города;

проведение сводных расчетов полей приземных концентраций загрязняющих веществ с выявлением вкладов предприятий и автотранспорта в загрязнение атмосферы;

нормирование выбросов предприятий с автоматическим формированием разрешающих документов на выброс для всех предприятий города;

разработка индивидуальных томов Предельно Допустимых Выбросов (ПДВ) для предприятий города в виде автоматических извлечений из Сводного тома ПДВ, включающих комплексную оценку качества атмосферного воздуха в районе расположения предприятий, вклады предприятий в загрязнение воздуха, мероприятия по снижению выбросов в атмосферу и прочее;

выявление виновников сверхнормативных загрязнений атмосферного воздуха и рекомендации по снижению сверхнормативных выбросов;

разработка предприятиями мероприятий по достижению нормативов ПДВ;

аналитическая обработка результатов общегородского инструментального мониторинга атмосферного воздуха;

получение картографического материала, характеризующего фактическое и прогнозное качество атмосферного воздуха в городе;

составление ежегодного аналитического обзора качества атмосферного воздуха города.

Разработанный ПАК позволяет автоматизировать работу в части управления качеством атмосферного воздуха на муниципальном уровне. Автоматизация приема и подготовки данных позволяет анализировать проблему в целом по городу и тем самым облегчает принятие управленческих решений.

Обработка данных фактического мониторинга

ПАК «Воздух – Город» позволяет вести базу данных инструментального (фактического) мониторинга загрязнения атмосферного воздуха в городе. Отличительной особенностью этих данных является неполнота (могут присутствовать отчетные периоды, за которые нет данных), неточность (обусловлена погрешностями измерительных приборов) и противоречивость (обусловлена тем, что за каждый отчетный период присутствуют данные, предоставляемые различными лабораториями, которые могут существенно отличаться).

Основные задачи, решаемые подсистемой интеллектуальной обработки данных фактического мониторинга, сводятся к следующему:

1) получение выборки данных для сопоставления с расчетными данными. Данное сопоставление позволяет выявить следующее:

возможные неточности (неполноту) в данных инвентаризаций, предоставляемых предприятиями, что может послужить причиной проверок по отношению к таким предприятиям;

неточности в статистических исследованиях влияния автотранспорта на загрязнение атмосферы;

дополнительные, неучтенные источники загрязнения (например, дорожно-ремонтные работы и т. д.).

2) прогнозирование развития обстановки в сфере загрязнения атмосферы.

Эти прогнозы являются информацией для внутреннего пользования природоохранной компании. Прогнозы служат дополнительной информацией при выработке стратегии действий по предотвращению возможного ухудшения экологической обстановки, либо, напротив, подтверждением правоты выбранной ранее стратегии.

Реализация программного комплекса

Для решения задачи обработки данных фактического мониторинга в ПАК используются нейросетевые технологии, на основе которых выполняются следующие функции:

1) обобщение данных. Поскольку за один отчетный период, как правило, накапливается множество данных, которые в общем случае отличаются друг от друга, то перед тем, как использовать эти данные, необходимо их обобщить. Для решения этой задачи применяется алгоритм нейросетевой кластеризации, разработанный в [4, 5]. Экспериментально подтверждено, что предложенный алгоритм обобщения данных дает решение, которое превосходит по качеству решение, получаемое традиционными математическими методами усреднения.

2) восстановление отсутствующих значений. Поскольку входными данными для задачи получения выборки данных для сопоставления с расчетными данными являются данные фактического (инструментального) мониторинга, который проводится лабораториями непосредственно в различных районах города, то может случиться так, что за отчетный период некоторые данные могут отсутствовать. Для получения «качественной» выборки данных необходимо восстановить отсутствующие данные. С этой целью в ПАК «Воздух – Город» применяется многослойный перцептрон, обучающийся на основе генетического алгоритма [4, 5]. Входными данными для нейросети являются предшест-

вующие (и по возможности последующие) данные инструментального мониторинга.

Для решения задачи прогнозирования будущего развития обстановки в сфере загрязнения атмосферы выполняется динамический подбор топологии нейронной сети (НС) с использованием другого генетического алгоритма. Задача подбора топологии НС решается в совокупности с задачей обучения НС. Для обучения НС применяются генетические алгоритмы, разработанные в [6], поскольку дают более оптимальное с точки зрения затрат машинного времени решение.

На рис. 1 представлена структурная схема ПАК.

Новизной ПАК «Воздух – Город» в первую очередь является создание комплексного инструмента управления качеством атмосферного воздуха города на основе прогнозного и фактического мониторинга, включающего оперативный контроль загрязнения атмосферы, выявление виновников загрязнения и получение оперативной информации для разработки природоохранных мероприятий.

ПАК имеет клиент – серверную архитектуру. Это позволяет организовать многопользовательский доступ к данным и дает возможность нескольким пользователям системы параллельно просматривать различные отчеты, строить карты загрязнения, подготавливать документы в области природоохранных мероприятий.

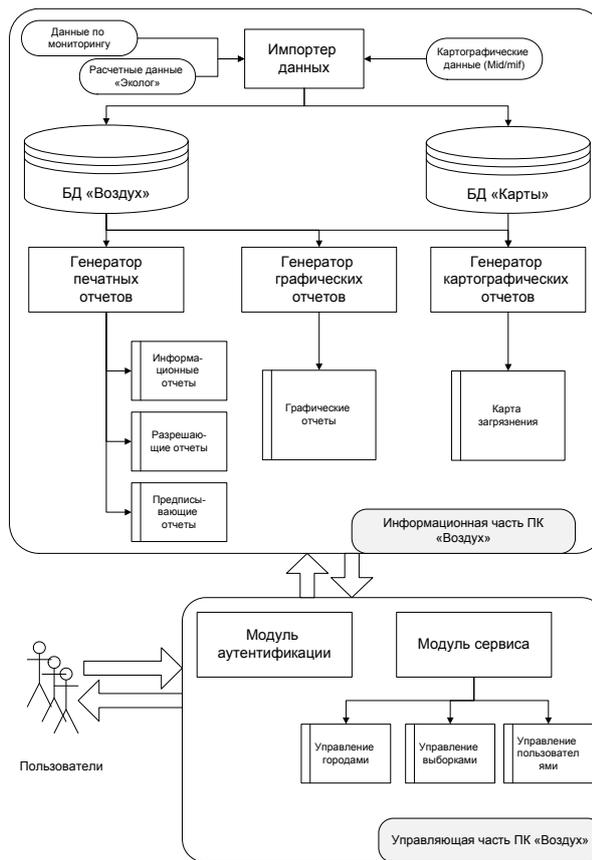


Рис.1. Структурная схема ПАК «Воздух – Город»

На рис. 2 представлена карта загрязнения атмосферы г. Калуги по веществу «Азота диоксид» с наложенными точками источников загрязнения.

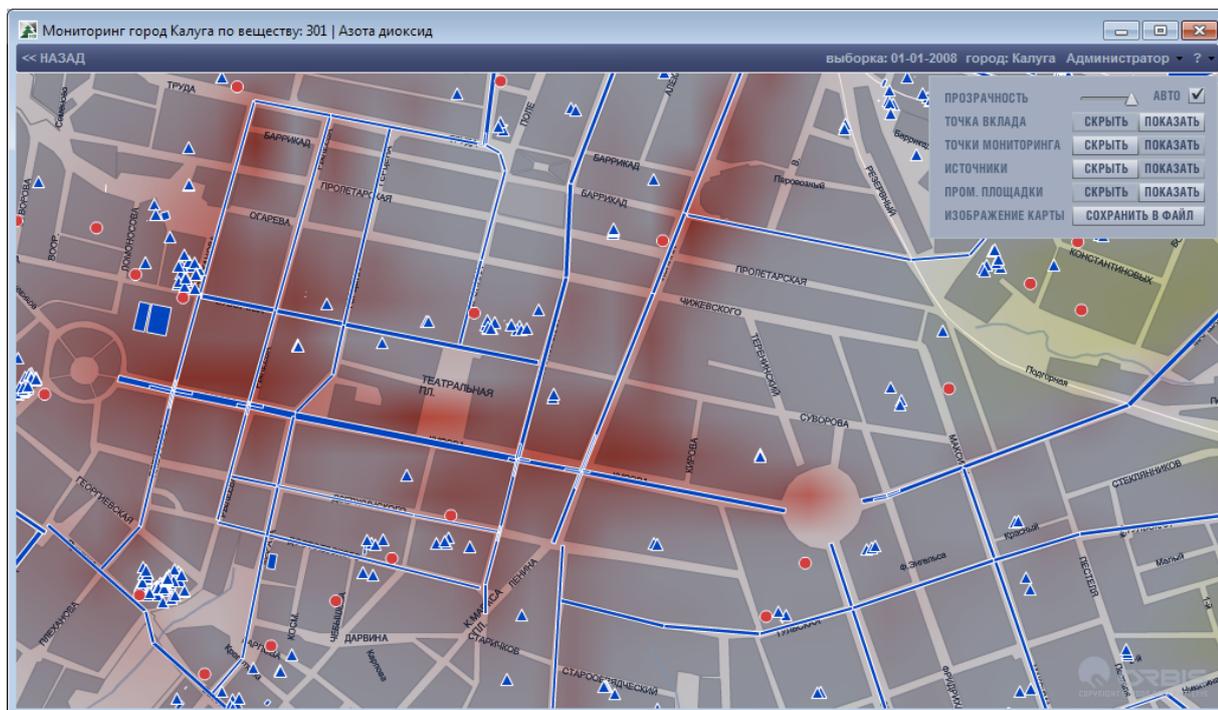


Рис. 2. Модуль картографии

Представленный комплекс построен по модульному принципу с контролем уровня доступа пользователя. Это позволяет автоматически настраивать ПАК под конкретный тип пользователя с дискретизацией до модуля. Кроме того, имеется возможность работы с несколькими городами, что избавляет от необходимости установки нескольких копий программы и базы данных для разных городов.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Головкин В. А. Нейронные сети: обучение, организация и применение. Кн.4: Учеб. Пособие для вузов / Общ. ред. А.И. Галушкина. – М.: ИПРЖР, 2001.
2. Рутковская Д., Пилинский М., Рутковский Л. Нейронные сети, генетические алгоритмы и нечеткие системы. – М.: Горячая линия – Телеком, 2007.

3. Комарцова Л. Г., Максимов А. В. Нейрокомпьютеры. – М.: Изд-во МГТУ им. Баумана, 2002.

4. Ковалев, И. В. Применение нейронных сетей для обобщения входных данных в системах прогнозирования / И. В. Ковалев // Рег. студ. н-т. конф. Ч. 1. Применение кибернетических методов в решении проблем общества XXI века. Тезисы докладов. Обнинск, 24–26 апреля 2009 г., С. 24–27.

5. Ковалев, И. В. Нейросетевой подход к решению задачи снижения размерности входного множества системы прогнозирования загрязнения атмосферы / И. В. Ковалев // Научн. сессия МИФИ-2009. XI Всерос. н-т. конф. «Нейроинформатика 2009». – М.: МИФИ, 2009, С. 66–68.

6. Ковалев И. В. Эволюционный подход к обучению нейронных сетей в интеллектуальной системе управления мобильного робота / И. В. Ковалев // Интеллектуальные системы: Труды Восьмого международного симпозиума / под ред. К. А. Пупкова. – М.: РУСАКИ, 2008, С. 128–131.

УДК 378.145

Н. Ю. Кожевникова

ВЛИЯНИЕ БОЛОНСКОГО ПРОЦЕССА НА ПРЕОБРАЗОВАНИЕ УЧЕБНЫХ БИЗНЕС-ПРОЦЕССОВ И РАСПРЕДЕЛЕНИЕ РЕСУРСОВ ВУЗА

Астраханский государственный университет
nina.kozhevnikova@aspu.ru

В статье рассматривается влияние Болонского процесса и перехода к ECTS на учебные бизнес-процессы вуза, описываются бизнес-процессы линейной и нелинейной форм организации учебного процесса, выделяются требующие решения задачи и методы их решения.

Ключевые слова: Болонский процесс, учебный бизнес-процесс, ECTS, линейная и нелинейная форма организации.

N. Y. Kozhevnikova

BOLOGNA PROCESS IMPACT ON THE EDUCATIONAL BUSINESS-PROCESSES REORGANIZATION AND UNIVERSITY RESOURCES DISTRIBUTION

Astrakhan state university

The article examines the impact of the Bologna Process and transfer to ECTS on university' educational business-processes, describes linear and nonlinear forms of educational business-processes, defines problems and methods, required for their solutions.

Key words: Bologna Process, educational business-process, ECTS, linear and nonlinear forms of organization.

19 сентября 2003 года на встрече европейских министров образования в Берлине Россия присоединилась к Болонской декларации и стала участницей Болонского процесса.

Целями Болонского процесса являются: расширение доступа к европейскому образованию и повышение его качества, расширение мобильности студентов и преподавателей посредством принятия сопоставимой системы ступеней высшего образования, применения системы учебных кредитов (ECTS), выдачи выпускникам вузов общеевропейского Приложения к диплому (Diploma Supplement) [1].

Одними из основных задач, требуемых решения в рамках реализации концепций Болонского процесса являются:

Внедрение Европейской системы учета и перевода кредитов.

«Асинхронная» или «Нелинейная» форма организации учебного процесса как средство повышения академической мобильности студентов и профессорско-преподавательского состава вуза.

Решение данных задач в рамках реализации Болонского процесса неизбежно приведет к реорганизации учебного процесса и реструктуризации его содержания.

Традиционная, «линейная» форма организации учебного процесса обладает следующими характеристиками:

1. Учебный план специальности формируется один раз при зачислении студентов на данную специальность и обозначает фиксированное место и трудоемкость для каждой дисциплины, не подлежащие изменению. Основным документом, регламентирующим учебный план, является государственный стандарт специальности.

2. Рабочий учебный план составляется на каждый учебный год, имеет фиксированную структуру и уточняет распределение учебной нагрузки, заявленной в учебном плане на текущий учебный год.

3. Нагрузка преподавателей определяется, исходя из рабочих учебных планов, и является постоянной в течение каждого учебного семестра.

4. Расписание занятий имеет фиксированную структуру и характеризуется равномерностью распределения в течение всего учебного семестра.

Общий унифицированный бизнес-процесс «линейной» организации учебного процесса представлен на рис. 1.

Ежегодно совместными усилиями учебного отдела, деканатов и кафедр на основе государственных стандартов, в соответствии с планом приема в вуз, а также нормами и приказами, формируется учебный план специальности, который представляет собой распределение учебных дисциплин по учебным семестрам с указанием трудоемкости и формы контроля. Перед началом учебного года на основе учебного пла-

на специальности в соответствии с нормами и распоряжениями для каждого набора специальности составляется рабочий учебный план. Рабочий план содержит распределение учебных дисциплин в рамках текущего учебного года с указанием семестра, итоговой формы контроля, ведущей кафедры, а также количества студентов, групп и подгрупп.

В соответствии с рабочими учебными планами кафедры проводят распределение учебной нагрузки и формируют учебные поручения. На основе рабочих учебных планов и учебных поручений формируется расписание учебных занятий. При «линейной» организации учебного процесса расписание занятий характеризуется равномерным распределением учебных занятий в течение учебного семестра для фиксированного набора групп каждой специальности.

В рамках реализации задач Болонского процесса по поддержке академической мобильности и внедрения системы зачетных единиц необходима перестройка существующего учебного бизнес-процесса, в виду того, что вуз должен перейти на несвойственную ему модель студентоцентрированного обучения. Этот переход возможен путем преобразования «линейной» формы организации учебного процесса в «асинхронную». Данные преобразования приведут к прозрачности учебного процесса и, как следствие, возникновению у студента возможности выбора индивидуальной образовательной траектории, определяемой индивидуальным учебным планом.

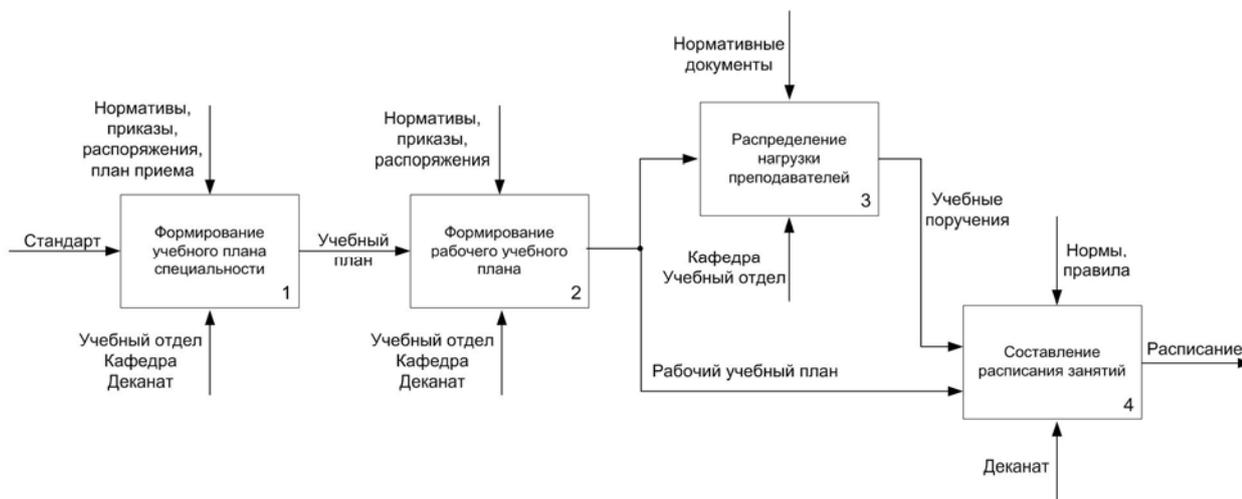


Рис. 1. Бизнес-процесс «линейной» формы организации учебного процесса

Согласно положению об организации учебного процесса в высшем учебном заведении с использованием системы зачетных единиц для реализации системы зачетных единиц целесообразно использовать три формы учебного плана по каждому направлению подготовки (специальности) [2]:

рабочие учебные планы по направлению подготовки (специальности), служащие для определения трудоемкости учебной работы студентов;

индивидуальные учебные планы студентов, определяющие их образовательную траекторию;

учебные планы, служащие для расчета трудоемкости учебной работы преподавателей.

По степени обязательности и последовательности усвоения содержания образования рабочий учебный план по направлению подготовки (специальности) должен включать три группы дисциплин: группа дисциплин, изучаемых обязательно и строго последовательно во времени, группа дисциплин, изучаемых обязательно, но, возможно, не последовательно, дисциплины, которые студент изучает по своему выбору. Первая группа дисциплин является базовой для определения курса (года обучения) студента, его учебного потока и учебной группы. Дисциплины второй и третьей групп создают предпосылки для «нелинейной» организации учебного процесса.

Таким образом, происходит переориентация учебного процесса с преподавателя на студен-

та, что неизбежно приводит к реинжинирингу существующих бизнес-процессов (рис. 2).

Очевидно, что в определенной степени неизменным останется только процесс формирования учебного плана специальности в соответствии с государственными стандартами третьего поколения. На основании учебного плана специальности формируются базовый рабочий учебный план и типовой индивидуальный учебный план студента. На основании типового индивидуального учебного плана студент составляет свой индивидуальный учебный план на следующий учебный год и сдает его в соответствующее подразделение вуза до установленного срока. План формируется по установленной форме, при необходимости, после консультации с тьютором. Тьютор – академический консультант, основная функция которого заключается в помощи студенту с выбором его индивидуальной образовательной траектории. Для этого тьютор должен обладать четким представлением обо всех возможных вариантах академической свободы студента, что требует серьезной подготовки.

Изменения в индивидуальный учебный план могут быть внесены студентом в срок до установленной даты. По каждой дисциплине вузом устанавливается минимальное число студентов, необходимое для открытия дисциплины, а для каждого преподавателя – максимальное число студентов в учебном потоке (группе).

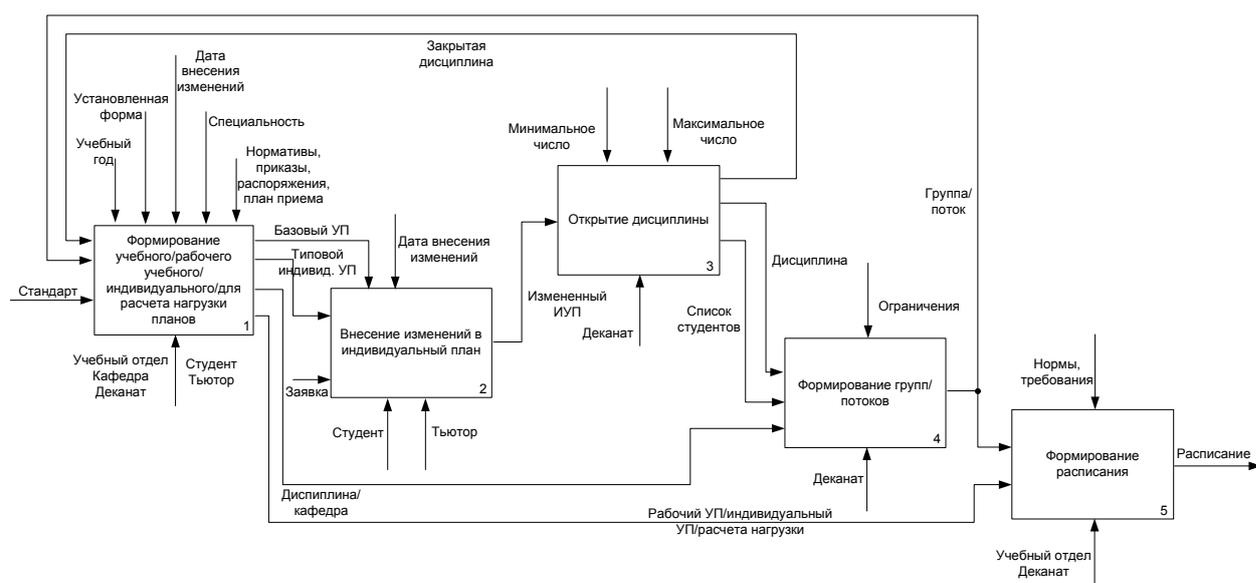


Рис. 2. Бизнес-процесс «нелинейной» формы организации учебного процесса

В случае если на данную дисциплину записалось число студентов, меньше минимально установленного, то дисциплина не открывается.

В случае если к данному преподавателю записалось число студентов, больше максимально установленного, то вуз формирует по этой дисциплине второй (при необходимости – больше) учебный поток (учебную группу) и назначает в него преподавателя не меньшей квалификации. При этом студенты распределяются по потокам (учебным группам) согласно порядку очередности записи. При этом в поток могут попасть студенты разных курсов.

По результатам анализа индивидуальных планов вузом составляются рабочие планы и расписание занятий на следующий учебный год. При этом процесс формирования планов является достаточно сложным и состоит из нескольких подпроцессов. Таким образом, блок 1 на диаграмме бизнес-процесса «нелинейной» формы организации учебного процесса можно представить как совокупность 5 процессов (рис. 3).

Учебный план, служащий для расчета учебной нагрузки преподавателей, составляется на основе рабочего учебного плана по направлению подготовки (специальности) с учетом анализа индивидуальных планов студентов данного направления подготовки (специальности).

По результатам анализа индивидуальных планов на основе базового рабочего учебного плана вузом составляются итоговые рабочие планы.

Очевидно, что переход к «нелинейной» форме организации учебного процесса приведет к усложнению существующих учебных бизнес-процессов и, как следствию, увеличению объема и сложности структуры набора исходных данных для принятия решения о распределении ресурсов вуза при составлении расписания занятий. Основная сложность состоит в двойственности задач, возникающих в течение всего переходного периода к стандартам третьего поколения, т. к. с одной стороны необходима модернизация всего учебного процесса в виду необходимости перехода к студентоцентрированной форме обучения, а с другой стороны, невозможно допустить снижение качества образовательного процесса традиционной формы, который не потеряет своих позиций до конца переходного этапа.

На основании вышесказанного можно выделить следующие условия функционирования вуза на время переходного периода:

1. Переход на образовательные стандарты третьего поколения.
2. Параллельное использование стандартов второго поколения.
3. Переход к студентоцентрированной форме обучения за счет применения европейской системы перевода и накопления кредитов (ECTS) и введения индивидуальных учебных планов.

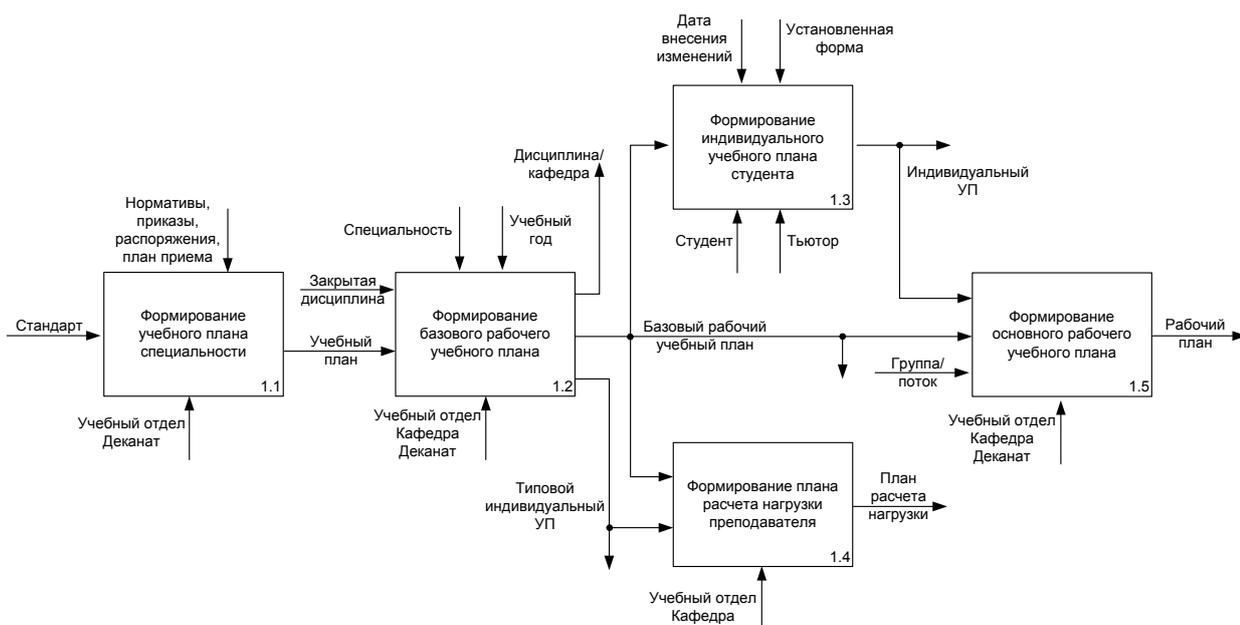


Рис. 3. Процесс формирования планов

В качестве методов реорганизации учебного процесса с учетом перечисленных условий можно выделить:

1. Реинжиниринг учебных бизнес-процессов.
2. Создание службы академических консультантов (тьюторов).
3. Перевод учебных планов в ECTS.
4. Переход к новой системе расчета нагрузки профессорско-преподавательского состава.
5. Изменение формы и правил составления расписания учебных занятий.
6. Разработка правил одновременного функционирования линейной (традиционной) и нелинейной (асинхронной) форм организации образовательного процесса.

Необходимо отметить, что последний пункт накладывает дополнительные условия на все этапы организации учебного процесса и требует дополнительных ресурсов. Но с другой стороны это состояние для вуза временное, следовательно, все принимаемые решения должны оставаться актуальными при полном переходе к стандартам третьего поколения.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Справка к заседанию коллегии Минобрнауки России 16 декабря 2004 года по вопросу «О реализации положений Болонской декларации в системе высшего профессионального образования Российской Федерации»
2. Примерное положение об организации учебного процесса в высшем учебном заведении с использованием системы зачетных единиц, министерство образования РФ.

УДК 004.588 + 519.21

В. В. Лантев, В. И. Сербин

УЧЕТ ВРЕМЕНИ ПРИ ОЦЕНИВАНИИ РЕЗУЛЬТАТОВ АВТОМАТИЗИРОВАННОГО КОНТРОЛЯ

Астраханский государственный технический университет

Астраханский государственный университет

serpentvv@mail.ru, viserbina@mail.ru

В статье рассматривается проблема учета времени при оценивании результатов выполнения контрольных заданий в автоматизированных обучающих системах. Полученная функция позволяет корректировать оценку в зависимости от времени решения и трудности задачи. Вероятностная трактовка функции оценки позволяет прогнозировать вероятность решения задания за нормативное время.

Ключевые слова: Автоматизированное тестирование, время решения, функция оценки, коэффициент трудности задания, вероятность решения, экспоненциальное распределение.

V. V. Laptev V. I. Serbin

USING TIMING INFORMATION DURING TESTS CHECKING IN AUTOMATED TRAINING SYSTEMS

Astrakhan State Technical University

Astrakhan State University

The article devoted to the problem of using timing information during tests checking in automated training systems. Proposed function allows to adjust the results of the tests depending on the time and the difficulty of the problem. The probabilistic interpretation of the evaluation function allows to predict the probability of solving tasks in normative time.

Key words: computer-aided testing, solving problem time, estimation function, task difficulty factor, solving problem probability, exponential distribution.

При автоматизации обучения основной проблемой является автоматизация контроля и оценивания знаний. В статье [1] приводится достаточно полный обзор моделей и методов проведения контроля и оценивания знаний, используемых в автоматизированных обучающих системах. Хорошие обзоры моделей и методов контроля можно найти в диссертациях [2] и [3], но эти модели практически не учитывают время выполнения заданий. Обычно оценивается доля выполненных контрольных заданий от

общего их количества и правильность выполнения каждого задания. Процесс контроля осуществляется в течение некоторого фиксированного интервала времени, по истечении которого выполнение заданий прекращается. Иногда ограничивается время выполнения отдельного задания – при превышении лимита времени задание не засчитывается. Время, выделяемое для выполнения отдельного задания, может быть либо фиксировано, либо вычисляется в зависимости от сложности задания. Подобные режи-

мы реализованы в системах УСАТИК [4], SunRav TestOfficePro [5], СИИТеЗ [6].

Отсутствие учета времени при оценивании заданий в процессе контроля объясняется в первую очередь тем, что контроль осуществляется посредством тестирования. Отдельное задание среди множества контрольных тестовых заданий имеет небольшой вес, поэтому нецелесообразно усложнять алгоритм оценивания за счет учета времени решения одного задания. Однако полное время выполнения контрольного теста достаточно велико и может служить дополнительной характеристикой при уточнении общей оценки за тест – это показывает следующий пример.

Пусть нормативное время выполнения контрольного задания составляет полчаса. Предположим, один обучаемый справился за 40 минут, а второй – за час. При традиционной методике оценивания оба получают одинаковую оценку – неудовлетворительную. Однако очевидно, что первый ученик лучше владеет материалом, чем второй. Поэтому естественно было бы оценить работу первого более высокой (хотя и не максимальной, так как ученик не уложился в норматив) оценкой, чем работу второго.

В отдельных системах учет времени решения все же ведется. Например, в Конструкторе Тестов [7] оценка выполнения отдельного задания оценивается в долях от 0 до 1. Если ответ неправильный, то оценка равна нулю. Если же ответ верный, то оценка вычисляется по формуле:

$$R = 1 - (t_b - t_n) / 100, \quad (1)$$

где t_b – время выполнения задания обучаемым, t_n – нормативное время выполнения задания. При этом t_n может либо задаваться как константа, либо вычисляться в зависимости от сложности задания. Если время ответа значительно превосходит нормативное, то $R < 1$. В этом случае задание считается не выполненным, и оценка равна нулю. Если же ученик выполнил задание быстрее, чем требуется по нормативу, то $R > 1$, и оценка равна 1. При $0 < R \leq 1$ оценка равна R . Таким образом, оценка снижается пропорционально времени, затраченному сверх лимита.

Формула (1) представляет собой линейную функцию вида $y = kx + b$:

$$R = -0,01t_b + (0,01t_n + 1).$$

Очевидно, что формула (1) является эмпирической, обоснование ее линейности в документации не приводится. Однако при показан-

ных ниже условиях возможно получить обоснованную формулу снижения оценки в зависимости от превышения лимита времени на решение задания.

Постановка задачи

Обозначим как t_n нормативное время решения задачи. Если задача решена за время $t \leq t_n$, то ученик получает максимальную оценку. Если же ученик не уложился в нормативное время, то оценка должна быть снижена. При отсутствии лимита времени правильное решение задачи оценивается некоторым минимальным количеством баллов. Обозначим:

R_{\max} – максимальная оценка решения задачи за время $t \leq t_n$;

R_{\min} – минимальная оценка решения задачи за неограниченное время.

Требуется найти $R(t)$ – неизвестную функцию оценки решения задачи за время $t \geq t_n$.

Функция оценки

Отметим, что $R(t) = R_{\max}$ при $t \leq t_n$, а далее оценка начинает снижаться. Поэтому момент превышения нормативного времени можно считать нулевым моментом времени, с которого начинается снижение оценки. Таким образом, функции $R(t)$ определена для $t \in [0, \infty)$. Очевидно, функция $R(t)$ непрерывна и убывает на всей области определения:

$$R(t_1) \geq R(t_2) \text{ при } t_1 \leq t_2,$$

$$\text{причем } R(0) = R_{\max}, R(\infty) = R_{\min}.$$

Отметим, что данным условиям удовлетворяют многие функции как линейные, подобные (1), так и нелинейные, например:

$$R(t) = (R_{\max} - R_{\min})/(t + 1) + R_{\min} - \text{гипербола};$$

$$R(t) = (R_{\max} - R_{\min}) \times 2 / (1 + e^t) + R_{\min} - \text{сигмоида}.$$

Это говорит о том, что необходимо наложить дополнительные ограничения на искомую функцию. Обозначим:

$$p(t) = (R(t) - R_{\min}) / (R_{\max} - R_{\min}).$$

Функция $p(t)$ – это нормированная функция оценки за время $t \geq 0$, причем $p_{\max} = p(1) = 1$, $p_{\min} = p(\infty) = 0$. Очевидно, $p(t)$ тоже является непрерывной убывающей функцией на всей области определения.

Пусть $g(t) = 1 - p(t)$ – это потери в оценке, полученной обучаемым, выполнившим задание за время $t \geq 0$. Для нахождения $p(t)$ сделаем следующие допущения:

1. Потери в оценке за приращение времени Δt пропорциональны Δt с точностью до бесконечно малых высших порядков относительно Δt .

2. Степень ухудшения оценки за приращение времени Δt зависит только от Δt и не зависит от начального момента времени t .

При данных ограничениях в работе [8] получена формула функции оценки:

$$p(t) = e^{-\lambda t}. \quad (2)$$

Переходя к искомой функции $R(t)$, имеем:

$$R(t) = (R_{\max} - R_{\min}) \times e^{-\lambda t} + R_{\min}. \quad (3)$$

При фиксированном значении λ значение $R(t)$ полностью определяется значениями R_{\min} и R_{\max} . На практике очень часто устанавливают $R_{\min} = 0$. В этом случае формула (3) принимает более простой вид:

$$R(t) = R_{\max} \times e^{-\lambda t}. \quad (4)$$

Таким образом, оценка решения задачи за время, превышающее норматив, составляет некоторую долю от максимальной.

Некоторые свойства оценки

Если обучаемый затратил на решение задачи время t_b , то фактическое время, превышающее нормативное время решения, вычисляется как $t = t_b - t_n$. Поэтому формула (4) принимает вид:

$$R(t) = R_{\max} \times e^{-\lambda(t_b - t_n)}. \quad (5)$$

Если ученик решает задачу быстрее норматива, то $t = t_b - t_n < 0$, и формула (5) позволяет начислить ему дополнительные баллы сверх R_{\max} .

Рассмотрим теперь, какой смысл может иметь коэффициент λ . Выразим время t через оценку p и коэффициент λ :

$$t = -\ln(p) / \lambda. \quad (6)$$

Обозначим $d = 1/\lambda$. Так как $\lambda > 0$, то и $d > 0$. Тогда формула (6) имеет вид:

$$t = -\ln(p) \times d. \quad (7)$$

Очевидно, что время решения задачи пропорционально ее трудности. Поэтому d можно считать коэффициентом трудности задачи. Подставив d в формулу (2) для вычисления $p(t)$, имеем:

$$p(t) = e^{-t/d}. \quad (8)$$

Коэффициент d может вычисляться автоматизированной системой или устанавливаться экспертом. Время же может отслеживать обучающая среда, в которой студент выполняет контрольную работу.

Вероятностная трактовка функции оценки

Функция оценки $p(t)$ принимает значения от 1 до 0, поэтому ее можно интерпретировать как вероятность появления некоторого случайного события.

Пусть ученик начинает решать задачу в момент времени $t_0 = 0$ и решает ее за некоторое время t . Рассмотрим непрерывную случайную вели-

чину T – время решения задачи. Функция распределения $F(t)$ этой случайной величины представляет собой вероятность того, что ученик потратит на решение задачи время $T < t$: $F(t) = P(T < t)$.

Функция оценки (8) представляет собой вероятность того, что ученик затратит на решение время $T \geq t$: $p(t) = P(T \geq t)$. Причем $p(0) = P(T \geq 0) = 1$, поскольку любая задача требует некоторого времени $t > 0$ для решения, и $p(\infty) = P(T \geq \infty) = 0$, так как любая задача решается за конечное время $t < \infty$. Тогда $P(T < t) = 1 - p(t) = q(t)$. Функция $q(t)$ – это вероятность того, что на решение задачи будет затрачено время $T < t$, причем $q(0) = 0$, так как нельзя решить никакую задачу за нулевое время. Для любой конкретной задачи трудности d с увеличением времени t вероятность ее решения возрастает, причем $q(t) \rightarrow 1$ при $t \rightarrow \infty$. Таким образом, функция $q(t)$ представляет собой функцию распределения $F(t)$ случайной величины T :

$$F(t) = q(t) = 1 - e^{-t/d}.$$

Очевидно, распределение непрерывной случайной величины T подчиняется экспоненциальному закону.

Заметим, что вероятность $P(T < t)$ является условной вероятностью. Рассматривать случайную величину T и вычислять вероятность $P(T < t)$ можно только после того, как задача решена. Пусть событие X – «выполнение задания» без учета времени его решения, обозначим его вероятность $P(X)$. Вероятность $P(T < t)$ является вероятностью случайного события Y – «время выполнения задания не превысило t » при условии, что событие X произошло: $P(T < t) = P(Y|X)$. Событие «выполнение задания за время, не превышающее t » – это произведение событий XY , и его вероятность вычисляется по правилу умножения вероятностей: $P(XY) = P(X) \times P(Y|X)$. Противоположное событие $\sim Y$ означает, что «время выполнения задания превысило t » при условии, что событие X произошло. Вероятность события $\sim Y$ равна $P(T \geq t) = P(\sim Y|X)$. Вероятность успешного решения задания за время $T \geq t$ вычисляется как произведение вероятностей: $P(X \sim Y) = P(X) \times P(\sim Y|X)$. Заметим, что эта вероятность отличается от вероятности события $\sim X$ – «задача не решена»: $P(\sim X) = 1 - P(X)$.

Искомая безусловная вероятность решения задачи $P(X)$ определяется в современной теории тестирования (Item Response Theory, IRT) [9]. В рамках этой теории строятся вероятностные модели успешного выполнения задания, зависящие от двух латентных параметров: уровня

подготовленности студента и трудности задания. Пусть d , как обычно, – коэффициент трудности задания, и r – уровень обученности студента. В двухпараметрической логистической модели Раша-Бирнбаума вероятность успешного выполнения задания определяется следующим образом:

$$P(r) = 1/(1 + e^{-a(r-d)}), \quad (9)$$

где a – дифференцирующий коэффициент. Вероятность P как функция от r при некоторой фиксированной трудности $d = d_0$ характеризует возможности обучаемого при решении задания трудности d_0 . Эта вероятность тем выше, чем выше уровень подготовленности ученика. Коэффициент a определяет «крутизну» логистической кривой, что позволяет хорошо различать шансы на успешное выполнение задания трудности d_0 учениками с уровнем подготовленности $r < d_0$ и $r > d_0$.

Таким образом, вероятность успешного выполнения задания трудности d_0 за время $T < t$ можно вычислить с учетом формулы (9) по правилу умножения вероятностей:

$$P(XY) = P(r) \times P(T < t) = (1 - e^{-td_0}) / (1 + e^{-a(r-d_0)}). \quad (10)$$

Некоторые следствия вероятностной трактовки

Формула (10) позволяет вычислять вероятности успешного выполнения нескольких заданий за нормативное время. Пусть требуется решить 2 задачи с коэффициентами трудности d_1 и d_2 . Это события совместные, но независимые, поэтому вероятность решения обеих задач равна:

$$P(AB) = P(A) \times P(B) = (1 - e^{-td_1}) / (1 + e^{-a(r-d_1)}) \times (1 - e^{-td_2}) / (1 + e^{-a(r-d_2)}).$$

Вероятность решения хотя бы одной из двух задач, вычисляется как вероятность суммы совместных событий, то есть $P(A + B) = P(A) + P(B) - P(AB)$:

$$P(A + B) = (1 - e^{-t/d_1}) / (1 + e^{-a(r-d_1)}) + (1 - e^{-t/d_2}) / (1 + e^{-a(r-d_2)}) - (1 - e^{-t/d_1}) / (1 + e^{-a(r-d_1)}) \times (1 - e^{-t/d_2}) / (1 + e^{-a(r-d_2)}).$$

Подобным же образом можно определить вероятность решения студентом N задач, или хотя бы одной из тех же N задач.

Заключение и выводы

Функцию оценки (формулы (2) – (5)) можно применять для оценивания результатов выполнения заданий в автоматизированной обучающей или тестирующей системе. Например, при обучении программированию выполнение задания означает написание соответствующей программы, что требует значительного времени. Учет времени решения в подобных случаях представляется весьма полезным.

Вероятностная оценка (формула (10)) позволяет прогнозировать успешное выполнение задания учеником с известным уровнем подготовленности. Этот прогноз автоматизированная обучающая система может использовать для адаптивного управления подбором заданий, адекватных уровню подготовки для данного ученика.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Зайцева Л. В., Прокофьева Н. О. Модели и методы адаптивного контроля знаний // Educational Technology & Society. – 2004, № 7(4). – С. 265–277.
2. Данилова С. Д. Адаптивная нечеткая модель оценивания результатов автоматизированного тестирования с разделением заданий по уровням усвоения: дис. ... канд. техн. наук. – Улан-Удэ: ВСГТУ, 2005. – 122 с.
3. Ульянов Д. А. Марковская модель адаптивного тестирования и ее программная реализация в условиях дистанционного обучения: дис. ... канд. техн. наук. – Иркутск: ИГТУ, 2005. – 119 с.
4. <http://usatic.narod.ru/>
5. <http://www.sunrav.ru/testofficepro.html>
6. <http://www.sintest.ru/>
7. <http://www.keepsoft.ru/simulator.htm>
8. Сербин В. И. Методы оценки знаний в подсистеме тренинга обучающей системы // Вестник Астраханского государственного технического университета, № 1(36)/2007. – С. 247–251.
9. Нейман Ю. М., Хлебников В. А. Введение в теорию моделирования и параметризацию педагогических тестов. – М.: Прометей, 2000. – 169 с.

УДК 004.052.42

*Д. В. Литовкин, О. А. Сычев***БИБЛИОТЕКА МОДУЛЬНОГО ТЕСТИРОВАНИЯ, ИСПОЛЬЗУЕМАЯ
ПРИ ОБУЧЕНИИ ПРОГРАММИРОВАНИЮ****Волгоградский государственный технический университет**
litd@mail.ru, oasychev@gmail.com

Рассматривается библиотека модульного тестирования, позволяющая проверять работоспособность функций, разрабатываемых студентами. В отличие от существующих библиотек она не требует написания дополнительного тестирующего кода и защищена от просмотра и модификации студентами.

Ключевые слова: библиотека модульного тестирования, обучение программированию, функция языка программирования.

*D. V. Litovkin, O. A. Sychev***UNIT-TESTING LIBRARY FOR USE IN TEACHING PROGRAMMING****Volgograd State Technical University**

The paper describes unit-testing library developed to test functions, implemented by students in order to learn programming. Unlike typical unit-testing libraries for professional programming it is adapted to learning purposes, don't requiring students to write any additional code and is protected from interference from the students.

Key words: unit-testing library, learning, programming education, programming language function.

Согласно государственному образовательному стандарту по направлению «552800 Информатика и вычислительная техника» в рамках дисциплины «Программирование на языке высокого уровня» студенты должны ознакомиться с понятием функции языка программирования. С этой целью студенты выполняют лабораторную работу, в которой требуется реализовать не менее двух функций на языке Си и использовать (вызвать) их для решения общей задачи. При этом для каждой функции задается ее заголовок [1], назначение и описание аргументов (см. рис. 1). Тем самым студент концентрируется на реализации функций, а не на их проектировании. Предполагается, что функция выполняет только преобразование входных данных в выходные, и они явно заданы в ее заголовке. Таким образом, функции не могут использовать глобальные переменные, осуществлять ввод/вывод данных и т. п.

Проверка правильности решения подобной задачи, создаваемого студентом, является трудоемким процессом и имеет ряд сложностей [7, 8, 9].

– Помимо функционального тестирования программы в виде «черного ящика», требуется ее модульное тестирование или unit-тестирование [2, 3], т. к. довольно часто программа работает в целом, но при этом оказывается, что функции либо не выполняют свое назначение, либо программа реализована без функций. Кроме того, нередко функции являются более универсальными, чем общая задача, поэтому тестирование программы в виде «черного ящика» не эквивалентно тестированию самих функций.

– Ручной процесс проверки функций является чрезвычайно субъективным, т. к. проводится путем визуального анализа кода функции и во многом зависит от квалификации преподавателя.

Формулировка

Дан многострочный текст (`char text[20][81]`). Верните слово по его индексу (индекс первого слова равен 0). Слово – это непрерывная последовательность букв и/или цифр. Разделителями слов могут быть любые другие символы. Исходный текст не модифицировать.

```
int strToWords (const char str[81], int words[40][21]);
```

Входные данные:

– `str` – исходная строка

Выходные данные:

– `words` – массив слов, из которых состоит исходная строка;
– возвращаемое значение – кол-во слов в массиве

Реализовать и использовать функцию...

Рис. 1. Фрагмент задания, включающего реализацию и использование функций

– Студент не должен иметь доступ к модульным тестам, т. к. он самостоятельно должен выделить ситуации использования функций и разработать алгоритмы для каждой такой ситуации. Кроме того, необходимо обеспечить защиту тестирующего кода от несанкционированной правки с целью получения завышенных результатов тестирования.

В связи с этим актуальной задачей является разработка библиотеки модульного тестирования, которая бы удовлетворяла следующим требованиям.

– Библиотека должна содержать модульные тесты для всех функций, которые встречаются в заданиях. Однако за один «прогон» должно тестироваться только некоторое подмножество функций, соответствующих одному варианту задания.

– Набор модульных тестов для функций должен корректироваться без изменения кода библиотеки. Изменение кода библиотеки допустимо при добавлении новых тестируемых функций или изменении заголовков уже существующих.

– Библиотека должна интегрироваться с тестируемой программой студента без создания дополнительного или модификации существующего кода. Так же недопустимо создание новых версий программы, специально предназначенных для тестирования.

– Модульные тесты должны быть закрыты для студентов, т. к. они предназначены для проверки работоспособности функций преподавателем. При необходимости преподаватель может показать студенту название модульного теста, а так же набор его входных и выходных данных.

– Результатом тестирования должен быть отчет, содержащий тесты, на которых тестируемые функции показали неверные результаты. Для каждого такого теста должно отображаться смысловое название и расхождения между эталонными и реальными выходными значениями.

– Должен обеспечиваться «безопасный» вызов тестируемых функций [4], который бы не приводил к нарушению процесса тестирования вследствие отсутствия тестируемой функции, неправильного набора ее аргументов, бесконечного выполнения, появлении исключительных ситуаций и т. д.

Анализ существующих каркасов для написания библиотек для unit-тестирования CppUnit, Boost.Test, CppUnitLite, NanoCppUnit, Unit++, CxxTest и др. [5] показал, что они не удовле-

творяют всем указанным требованиям и предназначены для промышленного тестирования, которое не требует поддержки избыточного набора функций, скрытия модульных тестов и т. д. В результате была спроектирована и реализована собственная библиотека модульного тестирования, учитывающая особенности тестирования функций в учебном процессе, а также система, используемая для проведения модульных тестов (см. рис. 2).

Типовой процесс модульного тестирования выглядит следующим образом.

Студенту выдается библиотека модульного тестирования в виде статической библиотеки (файл `str_unit.lib`) и заголовочный файл `str_unit.h`, содержащий заголовки всех тестируемых функций (точнее указателей на эти функции). Библиотека для каждой тестируемой функции содержит тестирующую функцию (ее работа будет рассмотрена ниже), а также функцию-заглушку. Функции-заглушки используются для тех функций, которые отсутствуют в тестируемой программе, т. к. язык Си при компоновке требует наличия кода всех функций, вызовы которых встречаются в программе. Кроме того, функция-заглушка вызывается, если студент неверно объявил функцию, что автоматически приводит выводу в протокол тестирования сообщения об отсутствии функции.

Для использования библиотеки тестирования студенту достаточно включить в проект тестирования, учитывающая особенности не требует заголовочный файл `str_unit.h` и поместить `str_unit.lib` в папку с проектом. Кроме того, с целью компоновки нескольких вариантов реализации одной и той же функции (функции, реализованной студентом, и функции-заглушки) в проекте должна быть задана специальная настройка компилятора `/FORCE:MULTIPLE`, игнорирующая множественное объявление функций. После компиляции и компоновки программы студент получает исполняемый файл тестируемой программы, который он отдает на проверку преподавателю.

Используя специальную программу «Автотестер» [6], преподаватель указывает вариант задания и путь к исполняемому файлу тестируемой программы. Программа «Автотестер» записывает в переменную окружения номер варианта задания и запускает тестируемую программу. В результате управление передается библиотеке модульных тестов, которая по номеру варианта запускает соответствующую тестирующую функцию.

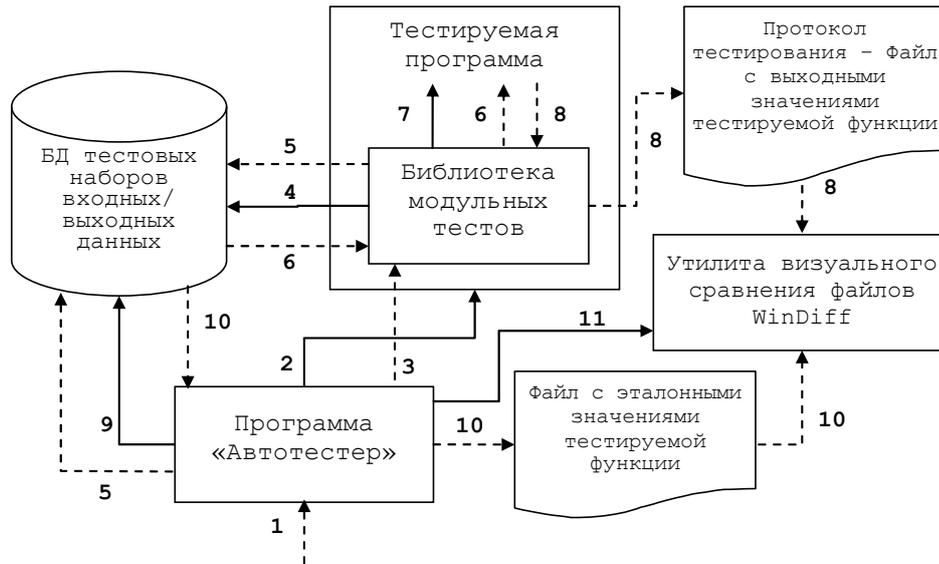


Рис. 2. Структура системы для проведения модульного тестирования:

1 – номер варианта задания и путь к исполняемому файлу с тестируемой программой – задается преподавателем; 2 – вызов «автотестером» тестируемой программы; 3 – переменная окружения с номером варианта задания; 4 – получение набора входных данных для модульного тестирования; 5 – номер варианта задания; 6 – набор входных данных для модульного тестирования; 7 – вызов тестируемой функции; 8 – выходные значения тестируемой функции; 9 – получение эталонных значений тестируемой функции; 10 – эталонные значения тестируемой функции; 11 – вызов утилиты визуального сравнения файлов

Тестирующая функция запускается многократно для каждого тестового набора входных/выходных данных. Такой подход позволяет завершить некорректно исполняемый набор данных и перейти к следующему набору. Типовой алгоритм тестирующей функции состоит из нескольких шагов:

получение на основе номера варианта задания набора входных данных из базы данных. Для чтения данных используются параметризованные функции [1], позволяющие считывать данные произвольного типа (реализованы как часть библиотеки);

«безопасный» вызов тестируемой функции с заданным набором входных данных («безопасный» вызов функции будет рассмотрен ниже);

анализ значений, полученных от тестируемой функции, на корректность – проверка диапазонов чисел, валидности указателей, завершения строк нуль-символом и т. д.;

запись полученных значений в протокол тестирования. Для записи значений используются параметризованные функции, позволяющие записывать данные произвольного типа (реализованы как часть библиотеки). Помимо выходных значений тестируемой функции в протокол тестирования добавляется название тестового набора.

По окончании работы тестирующей функции управление возвращается программе «Автотестер», которая, на основе базы данных тестовых наборов данных формирует файл с эталонными значениями. Затем этот файл и файл с выходными значениями тестируемой функции передается утилите визуального сравнения текстовых файлов – WinDiff. Как результат преподаватель видит различие между эталонными значениями и реальными значениями, которые вернула тестируемая функция (см. рис. 3).

```

int strToWords (const char str[81], char words[40][21])
{
    5. В строке несколько слов
    [out]>>words
    I
    |> don't
    | have
    | email
    [out]>>return
    <| 3
    |> 4
}

```

Рис. 3. Результат модульного тестирования функции strToWords, которая выделяет слова в строке – окно программы WinDiff

Поскольку тестируемая функция запускается в контексте тестирующей функции (точнее тестирующей программы), то при создании библиотеки особое внимание было уделено мерам «безопасного» вызова тестируемой функ-

ции и обеспечения стабильности тестирования в целом (никакие ошибки в тестируемой функции не должны приводить к сбою процесса тестирования). Для этого использовались следующие подходы:

входные данные из тестовых наборов проверяются на соответствие аргументам тестируемой функции (по названию и типу). Если обнаруживается несоответствие, то в протокол тестирования выводится предупреждение;

перед записью в протокол тестирования значений, возвращаемых тестируемой функцией, они обязательно проверяются на корректность (см. выше). При обнаружении неверных значений в протокол выводится сообщение с указанием деталей проблемы;

с помощью механизма обработки исключений «перехватываются» ошибки исполнения тестируемой функции [1]. При этом в протокол тестирования выводится соответствующее сообщение, а тестирование продолжается со следующего теста;

каждый тест запускается в отдельном потоке (нити) управления. При превышении им предельного времени тестирования («зацикливании» тестируемой функции) его выполнение прерывается с выдачей сообщения в протокол тестирования;

в случае если тестируемая функция не найдена (чаще всего она не обнаруживается из-за несоответствия типов данных в заголовке функции), то вызывается функция-заглушка, которая печатает в протокол тестирования сообщение об отсутствии тестируемой функции;

детектируются случаи некорректного использования библиотечных функций `strcat`/`strcpy` внутри тестируемой функции. Это достигается путем анализа фактических аргументов, передаваемых в функции `strcat`/`strcpy`. Сообщение об обнаруженной ошибке выводится в протокол тестирования.

Таким образом, удалось добиться полной стабильности процесса тестирования, а также вывода осмысленных сообщений об этих ошибках, что облегчает проверку правильности программ.

Созданная библиотека модульного тестирования на текущий момент содержит код для

тестирования более 120 функций; для половины из них составлены тестовые наборы входных/выходных данных. Апробация библиотеки проводилась на занятиях у 6 групп студентов первого курса. Не смотря на экспериментальную версию библиотеки, она позволила выявить неверную реализацию функций более чем у 30 % студентов. После уточнения заголовков тестируемых функций, исправления ошибок в библиотеке и расширения набора тестовых данных ожидается повышение качества проверки функций, уменьшение влияния человеческого фактора и сокращение времени тестирования.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Павловская, Т. А. C/C++. Программирование на языке высокого уровня. – СПб.: Питер, 2008. – 460 с.
2. Лайза Кристин. Гибкое тестирование. Практическое руководство для тестировщиков ПО и гибких команд / Лайза Кристин, Джанет Грегори. – М.: Вильямс, 2010. – 464 с.
3. Джерард Месарош. Шаблоны тестирования xUnit. Рефакторинг кода тестов. – М.: Вильямс, 2009. – 832 с.
4. Майкл Говард. Защита кода с помощью средств защиты Visual C++ [Электронный ресурс] / электронная версия журнала MSDN Magazine, – 2008. – Режим доступа: <http://msdn.microsoft.com/ru-ru/magazine/cc337897.aspx>
5. Noel. Exploring the C++Unit Testing Framework Jungle [Электронный ресурс] / Noel, – 2004. – Режим доступа: <http://gamesfromwithin.com/exploring-the-c-unit-testing-framework-jungle>
6. Литовкин, Д. В. Назначение и область применения системы тестирования заданий студентов, выполняемых в рамках дисциплины «Программирование на языке высокого уровня» / Д. В. Литовкин, О. А. Сычев, В. В. Криворучко // Информационные технологии в образовании, технике и медицине: матер. междунар. конф., 21–24 сент. 2009 / ВолгГТУ [и др.]. – Волгоград, 2009. – С. 37.
7. Дворянkin, А. М. Применение автоматизированного тестирования на лабораторных работах / А. М. Дворянkin, И. Г. Жукова, О. А. Сычев // Известия ВолгГТУ. Серия «Новые образовательные системы и технологии обучения в вузе»: межвуз. сб. науч. ст. / ВолгГТУ. – Волгоград, 2007. – Вып. 4, № 7. – С. 166–168.
8. Воробкалов, П. Н. Quality Estimation of e-Learning Systems / П. Н. Воробкалов, В. А. Камаев // Methodologies and Tools of the Modern (e-) Learning: suppl. to Int. Journal «Information Technologies and Knowledge». – 2008. – Vol. 2, [Int. Book Series «Inform. Science & Comput.»; № 6]. – С. 25–30. – Англ.
9. Кудряшов П. П., Фоменков С. А. Автоматизация проверки решения задач по механике // Педагогическая информатика, 2005, № 5 (спец. выпуск), С. 96–101.

УДК 004.89

*А. Ю. Ужва, И. Г. Жукова, М. Б. Кульцова***ОНТОЛОГИЧЕСКАЯ МОДЕЛЬ ПРЕДМЕТНОЙ ОБЛАСТИ,
ОБЕСПЕЧИВАЮЩАЯ ПОДДЕРЖКУ РАССУЖДЕНИЙ ПО ПРЕЦЕДЕНТАМ
ДЛЯ ПОИСКА ОБРАЗОВАТЕЛЬНЫХ РЕСУРСОВ****Волгоградский государственный технический университет**
alexey@uzhva.ru, zhukova@vstu.ru

Рассматривается онтологическая модель предметной области «электронные образовательные ресурсы», обеспечивающая поддержку алгоритма рассуждения по прецедентам для поиска электронных образовательных ресурсов за счет онтологической модели прецедента.

Ключевые слова: онтологическая модель представления знаний, рассуждения по прецедентам, образовательные ресурсы.

*A. Y. Uzhva, I. G. Zhukova, M. B. Kultsova***ONTOLOGY MODEL FOR CASE-BASED REASONING IN SEARCH OF EDUCATIONAL RESOURCES****Volgograd State Technical University**

The ontology model of educational resources domain with support of case-based reasoning considered. The described model supports adaptive search algorithm with respect to student personal.

Key words: ontology, case-based reasoning, educational resources.

Введение

В настоящее время создано большое количество электронных образовательных ресурсов, применимых в образовательном процессе. Показано, что электронное образование способно повысить эффективность обучения, понизив при этом стоимость образовательного процесса [2].

Электронные образовательные ресурсы могут быть использованы в образовательном процессе как при помощи самостоятельного выбора обучаемым необходимых материалов, так и путем получения готовых образовательных коллекций от преподавателя.

Проведенный анализ позволяет утверждать, что поиск образовательных ресурсов преподавателем, и создание им персонализированных образовательных коллекций, является наиболее эффективным способом структурирования образовательной информации для использования ее в образовательном процессе. Т. к. неавтоматизированный процесс поиска образовательных коллекций является трудоемким и затратным, то актуальным направлением исследований является автоматизация процесса поиска персонализированных образовательных коллекций.

Анализ подходов на основе нейронных сетей, решающих деревьев, онтологии, рассуждений по прецедентам, а так же других подходов [1], показал, что наиболее перспективным является подход, сочетающий в себе рассуждения на онтологии [4] и рассуждения по прецедентам [3].

Рассуждения по прецедентам позволяют учесть опыт предыдущих решений, увеличить количество используемых при адаптации параметров, повысить скорость работы системы, а онтологическая модель – обеспечить соответствие создаваемой коллекции требованиям учебного процесса, задействовать не только числовые параметры, но и знания о предметной области, учебном курсе, образовательном ресурсе.

Для реализации данного подхода должна быть создана онтологическая модель представления знаний, обеспечивающая интеграцию представленных на ней знаний на всех этапах цикла рассуждений по прецедентам.

Требования к структуре и функциям модели предметной области

Для реализации поставленной задачи было принято решение построить онтологическую модель предметной области, включающей в себя следующие структурные компоненты:

1. Модель образовательного курса, содержащая знания об образовательных дисциплинах, включая состав входящих в рабочую программу дисциплины вопросов (тем), их объемами в учебных часах, а так же знания об отношениях и связях между этими вопросами.

2. Модель обучаемого, содержащая знания об обучаемом, включая его персональные характеристики и предпочтения, а так же модель текущего поля знаний студента.

3. Модель образовательного ресурса, содержащая знания об образовательных ресурсах,

включая их библиографические данные, набор охватываемых ресурсом тем, а так же качественные характеристики ресурса:

- а) эффективность изложения материала;
- б) понятность изложения;
- в) практическая направленность.

4. Модель образовательной коллекции, которая должна содержать список включенных в коллекцию образовательных ресурсов, а так же последовательность их изучения.

5. Модель прецедента, обеспечивающую поддержку выбранных методов и алгоритмов построения персонифицированных образовательных коллекций.

6. Базы данных прецедентов, образовательных ресурсов, образовательных коллекций.

На данной модели решена задача поиска образовательных ресурсов при помощи составления персонифицированных образовательных коллекций в соответствии с требованиями рабочей программы и образовательного стандарта, а так же персональными предпочтениями и характеристиками пользователя, с использованием методов рассуждений по прецедентам.

Разработка модели

Разработка и верификация онтологических моделей представления знаний является затратной, поэтому была проведена адаптация одной из существующих онтологических моделей, наиболее близко соответствующей предъявляемым требованиям.

Разработанная онтологическая модель предметной области М состоит из следующих компонент:

$M = \langle \text{MetadataO}, \text{CourseO}, \text{ResourceO}, \text{SturentO}, \text{CollO} \rangle$, где

MetadataO – онтологическая модель мета-описания,

CourseO – онтологическая модель образовательного курса,

ResourceO – онтологическая модель образовательного ресурса,

SturentO – онтологическая модель обучающегося,

CollO – онтологическая модель образовательной коллекции.

Онтологическая модель мета-описания MetadataO определяет структуру мета-описания, используемую в других онтологических моделях данной предметной области. Модель позволяет ассоциировать экземпляры классов с набором пар «ключ-значение», при помощи которых мо-

гут быть сохранены данные о сущностях предметной области.

Онтологическая модель образовательного курса CourseO включает в себя описание образовательных курсов и их элементов, а так же взаимосвязей между ними, определяющих принадлежность образовательных единиц к образовательному курсу, а так же зависимость между образовательными единицами и междисциплинарные связи.

Модель описания образовательных ресурсов ResourceO обеспечивает поддержку аннотирования электронных образовательных ресурсов и задания соответствия образовательного ресурса и концепта предметной области. Модифицированная модель образовательного ресурса представляет собой множество:

$$CResource = \langle \text{ContentType}, \text{Type}, \text{Verbosity}, \text{Role}, \text{Language}, \text{Complexity}, \text{Prerequisites}, \text{Results} \rangle,$$

где:

ContentType – тип ресурса,

Type = {Practice, Theory} – характеристика теоретической, либо практической направленности данного образовательного ресурса,

Verbosity = {High, Medium, Low} – характеристика подробности изложения материала,

Language – язык материала,

Complexity – сложность изложения,

Prerequisites – требуемые для изучения знания,

Results – знания, получаемые в процессе обучения.

Онтологическая модель обучающегося SturentO обеспечивает поддержку описания свойств и характеристик обучающегося, таких как его идентификационные данные (ФИО, группа, курс), текущее поле знаний, персональные характеристики и предпочтения. Модель пользователя Student имеет вид:

$$\text{Student} = \langle \text{MetaProperties}, \text{Knowledge}, \text{Preferences} \rangle,$$
 где

MetaProperties = {MetaProperty₁, ..., MetaProperty_n} – набор метаданных пользователя, включающих ФИО, группу обучения и прочую информацию;

Knowledge – модель текущего поля знаний студента;

Preferences – модель персональных предпочтений пользователя.

Онтологическая модель образовательной коллекции CollO обеспечивает поддержку агрегации образовательных ресурсов в коллекции, соответствующие целям персональным характеристикам обучаемых.

Для обеспечения модульности и расширяемости, данные о конкретных образовательных курсах, образовательных ресурсах, пользователях и образовательных коллекциях размещены в специальных репозиториях:

$RepoS = \{ResourceRepo, CollRepo, StudentRepo, CourseRepo\}$, где

$ResourceRepo$ – репозиторий образовательных ресурсов,

$CollRepo$ – репозиторий образовательных коллекций,

$StudentRepo$ – репозиторий профилей обучающихся,

$CourseRepo$ – репозиторий образовательных курсов.

На описанной онтологической модели формализованы знания о первой части курса «Алгоритмические языки программирования» и вводной части курса «Интеллектуальные системы». Описаны 92 простых и составных образовательных ресурсов, 2 рабочих программы, состоящих из 52 элементов, 30 пользователей, 20 образовательных коллекций.

Модель прецедента

Прецедент C в общем виде представляет собой структуру следующего вида:

$$C = \langle Ind, Sol \rangle,$$

где: Ind – описание задачи (индекс прецедента),

Sol – описание решения данной задачи.

Для поддержки построения адаптивных образовательных коллекций, индекс прецедента расширен, и в его состав включена информация не только о решаемой задаче, но и о персональных свойствах обучаемого:

$$Ind = \langle Task, UserDescr \rangle,$$

где: $Task$ – формальное описание образовательной задачи;

$UserDescr$ – формальное описание характеристик обучаемого, для которого проводится адаптация.

Таким образом, прецедент поиска персонализированной образовательной коллекции представлен в виде трехкомпонентной структуры, включающей в себя формальное описание пользователя, для которого создается образовательная коллекция, формализацию поставленной перед ним образовательной задачи, а также найденном решении в виде модели персонализированной образовательной коллекции.

Формально описание прецедента $Case$ имеет вид:

$$Case = \langle Task, UserDescr, Sol \rangle,$$

где: $Task$ – описание образовательной задачи;

$UserDescr$ – описание характеристик обучаемого, для которого проводится адаптация;

$Sol \subset Coll$ – описание принятого решения в виде образовательной коллекции, составленной для данной образовательной задачи $Task$ и текущих персональных характеристик $UserDescr$.

При этом под образовательной задачей $Task$ понимается набор теоретических и практических знаний, которые обучаемый должен получить в процессе обучения:

$$Task = \{KnowledgeItem_1, \dots, KnowledgeItem_n\},$$

где $KnowledgeItem_i$ – i -й элемент целевого поля знаний студента, который необходимо изучить в процессе обучения при помощи составляемой электронной коллекции.

Описание характеристик пользователя в свою очередь состоит из двух компонент:

$$UserDescr = \langle UserPreferences, UserKnowledge \rangle,$$

где: $UserPreferences$ – описание предпочтений обучаемого,

$UserKnowledge$ – описание текущего поля знаний обучаемого.

На данной модели составлено описание 63 прецедентов поиска образовательных коллекций. Поставлена задача разработки алгоритма выбора близких прецедентов с использованием онтологической функции оценки близости, которая должны учитывать онтологические знания об обучаемых, рабочих программах и образовательных ресурсах.

Интеграция представленных моделей в цикл рассуждений по прецедентам

Обеспечена интеграция онтологической модели представления знаний во все этапы проведения рассуждения по прецедентам (рис. 1):

- Онтология описывает структуру задания прецедента. Онтология содержит информацию о возможных целях обучения, текущих знаниях студента и его индивидуальных особенностях. Онтология задает основу для поддержки анализа полноты предоставленных сведений о знаниях студента.

- Онтология может быть использована на этапе поиска прецедента для доопределения индекса прецедента (если пользователь его не полностью задал) на основе содержащихся в онтологии знаний. Также алгоритмы поиска и оценки близости прецедентов сильно зависят от модели представления прецедента, заданной онтологией.



Рис. 1. Схема интеграции онтологий и CBR
 Цифрами на схеме обозначены: 1 – описание модели прецедента; 2 – формализованные знания предметной области; 3 – сохраненный прецедент; 4 – новый прецедент

- На этапе адаптации используются знания, содержащиеся в онтологии в виде правил сопоставления одних параметров описания с другими параметрами для доопределения параметров, для которых наиболее близкие к прецеденту-запросу прецеденты дают противоречивые значения.

- На этапе оценки эффективности адаптации, заложенные в онтологию правила используются для определения качества адаптации. При помощи онтологических знаний созданная коллекция может быть проверена на соответствие требованиям рабочей программы и знаниям студента.

На данной модели поставлена задача реализации алгоритма поиска образовательных ресурсов на основе рассуждения по прецедентам, реализующего процедуру оценки близости фраг-

ментов онтологии, соответствующих компонентам описания прецедента Task и UserDescr.

Выводы

Таким образом, разработанная онтологическая модель представления знаний о предметной области «электронные образовательные ресурсы» обеспечивает поддержку алгоритма рассуждения по прецедентам для поиска электронных образовательных ресурсов за счет онтологической модели прецедента.

На данной онтологической модели формализованы знания о первой части курса «Алгоритмические языки программирования» и вводной части курса «Интеллектуальные системы». Описаны 92 простых и составных образовательных ресурсов, покрывающих 45 формализованных частей образовательных курсов.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Интеграция онтологических моделей представления знаний и рассуждений по прецедентам для генерации персонализированных образовательных коллекций / А. М. Дворянkin, А. Ю. Ужва, И. Г. Жукова, М. Б. Кульцова // Инновационные технологии в обучении и производстве. – сб. науч. тр. всеросс. конф. – Камышин, 2009. – С. 59.
2. Учебно-методические комплексы в системе ресурсов информатизации образования: опыт проектирования / В. А. Камаев, Д. А. Давыдов, А. М. Дворянkin, А. В. Кизим // Педагогическая информатика. – 2005. – № 5 (спец. выпуск). – С. 132–134.
3. *Aamodt, A. Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches/ A. Aamodt, E. Plaza // Communications. – 1994. – № 7. – С. 39–59.*
4. *Zhou, L. Ontology learning: state of the art and open issues // Information Technology and Management. – 2007. – т. 8, № 3. – С. 241–252.*

УДК 004.42
X 20

И. М. Харитонов, Е. Н. Скрипченко

ПРИМЕНЕНИЕ КОНТЕНТ-АНАЛИЗА ДЛЯ ОЦЕНКИ ТЕРМИНОЛОГИЧЕСКОЙ СВЯЗИ МЕЖДУ УЧЕБНЫМИ ДИСЦИПЛИНАМИ

**Камышинский технологический институт (филиал)
Волгоградского государственного технического университета
Wisdom_monk@mail.ru, Ekateina_skripch@mail.ru**

Предлагается методика нахождения и количественной оценки терминологических взаимосвязей между учебными дисциплинами, направленная на совершенствование планирования учебного процесса. Получение оценок основано на известных методах контент-анализа. В качестве примера рассмотрены оценки терминологической связи некоторых естественнонаучных дисциплин со специальными дисциплинами, изучаемыми при подготовке инженеров-системотехников.

Ключевые слова: управление учебным процессом, взаимосвязи между дисциплинами, контент-анализ.

I. M. Kharitonov, E. N. Skripchenko

CONTENT-ANALYSIS METHODS APPLICATION FOR THE BRANCHES OF STUDY TERMINOLOGY INTERCONNECTIONS ESTIMATION

**Kamishin Technological Institute (branch)
Vologorad State Technical University**

The designation and quantitative estimation facilities are proposed for the branches of study terminology interconnections levels purposes. The estimates are based on the known content-analysis methods. The terminology interconnections estimations of the several natural-sciences and special branches of systems engineering studies are presented as the example.

Key words: study process control, branches of study interconnections, content-analysis.

Введение

Развивающиеся наука и техника предъявляют новые требования к содержанию высшего образования, вытекающие из изменений содержания труда, развития структуры рынка труда. При этом повышение требований экономики к качеству подготовки специалистов не сопровождается адекватным улучшением содержания образования.

По сравнению со средней школой, где номенклатура предметов относительно стабильна, в высшей школе вместе с появлением новых отраслей знаний возникают и новые специальные предметы. Так как время на непосредственную их апробацию в вузах весьма ограничено, то многие специалисты предполагают изучать различные аспекты содержания образования, пользуясь методами моделирования. Такая структурная модель может отражать весь процесс обучения [1]. Попытки моделировать учебный процесс помогают подойти к нему как к сложной динамической системе и учесть те широкие возможности, которые открывает применение технических средств переработки информации. В исследовании учебного процесса модель выступает как важнейшее средство наглядного представления связей и отношений его компонентов. Соответственно для организации и научного исследования учебного про-

цесса высшей школы моделирование становится все более насущно необходимым [2]. Особую значимость моделирование учебного процесса приобретает сейчас, когда происходит переход к новым государственным образовательным стандартам. Использование данного подхода поможет спрогнозировать последствия интеграции и дифференциации учебных дисциплин в результате текущего реформирования учебного процесса.

Сущность междисциплинарных связей

Модели позволяют находить оптимальные структуры процесса обучения исходя из поставленной цели. Средствами оптимизации обучения являются: отбор содержания обучения и установление последовательности при изучении учебных дисциплин, прочных связей и взаимоотношений между предметами и видами обучения. Чем теснее и глубже эта связь (в частности, изучение одного предмета на основе знаний, полученных в другом), тем выше уровень научной и профессиональной подготовки специалистов. Нахождение связей между учебными дисциплинами наиболее важный из компонентов управления учебным процессом [2].

Междисциплинарные связи в обучении являются конкретным выражением интеграционных процессов, происходящих сегодня в науке

и в жизни общества. Эти связи играют важную роль в повышении практической и научно-теоретической подготовки будущих специалистов. С помощью многосторонних междисциплинарных связей не только на качественно новом уровне решаются задачи обучения и развития учащихся, но также закладывается фундамент для комплексного видения, подхода и решения сложных задач будущей профессиональной деятельности.

Наиболее подходящим способом для нахождения междисциплинарных связей в процессе обучения, является методика контент-анализа [3]. Контент-анализ – формализованный метод изучения текстовой и графической информации, заключающийся в переводе изучаемой информации в количественные показатели и ее статистической обработке, который характеризуется большой строгостью, систематичностью.

В контент-аналитических исследованиях используются следующие методы:

1. Частотная оценка. В контент-анализе самыми бедными по содержанию и в то же время самыми фундаментальными являются простые оценки частот. $f(c,t)$ – частота встречаемости характеристики c в тексте t .

2. Условно-частотная оценка. Простые частоты являются не самой подходящей оценкой текстов. Проблемы с ними могут возникнуть в том случае, если сравниваются разные по длине тексты. Вычисляется она как частота встречаемости $f(c,t)$ деленная на длину текста $L(t)$.

3. Нормированная оценка. Применяется для определения характера какого-либо отдельного текста. Для нахождения нормированной оценки следует сформировать категорию определенных терминов по какой-либо тематике Km и сопоставить ей условную частоту встречаемости в обычной речи $pr(Km, \text{речь})$, которая называется нормой для категории Km . При анализе конкретного текста t подсчитывается условная частота $pr(Km,t)$. Если она существенно больше нормы $pr(Km, \text{речь})$, то текст t относят к заданной тематике.

4. Контекстная оценка. Основная идея контекстного анализа заключается в том, что анализу подвергается не весь текст, а лишь некоторая выборка из него, являющаяся контекстом употребления характеристики c .

5. Контент-мониторинг. Если анализу подвергается массив упорядоченных во времени текстов, поступивших из одного источника, речь идет уже не о простом контент-анализе, а о кон-

тент-мониторинге текстовой информации. В этом случае, появляется дополнительная возможность применить математический аппарат многомерно-регрессионного анализа, аппарат анализа временных рядов, методы технического анализа.

В данной работе автором использовался метод условно-частотной оценки, так как происходил анализ множества текстов различного объема с целью выявления в них определенных терминологических понятий.

Использование методики контент-анализа позволяет найти и количественно оценить взаимосвязи между учебными дисциплинами. Для этого авторами был разработан программный комплекс, позволяющий производить поиск ключевых понятий или термов заданных учебных дисциплин в общем изучаемом материале других учебных дисциплин.

Алгоритм разработанной методики

Для нахождения и оценки взаимосвязей между учебными дисциплинами авторами была разработана и опробована следующая методика:

1. Сначала осуществляется формирование списка ключевых понятий или термов всех учебных дисциплин учебного плана по заданной специальности. Такой список будет являться исходными данными для выполнения анализа. Эта работа выполняется группой преподавателей читающих данные дисциплины на основании коллективной экспертной оценки.

2. Далее происходит формирование базы учебного материала, излагаемого преподавателем в процессе преподавания дисциплины. В такую базу вносится материал, не только лекционных занятий, но и материал практических, лабораторных и семестровых работ по заданной дисциплине. Вместе со списком термов такая база тоже является исходными данными для анализа.

3. После формирования всех исходных данных для анализа, с помощью разработанного программного комплекса происходит непосредственный процесс контент-анализа. То есть происходит поиск всех термов учебных дисциплин в общей базе учебного материала, сформированного всеми учебными дисциплинами учебного плана. В результате такой обработки получатся данные о том, сколько раз термин одной учебной дисциплины встретился в учебном материале других учебных дисциплин. Пример анализа с помощью разработанного программного комплекса представлен на рис. 1.

Список терминов		Количество терминов							
1ое слово	2ое слово	1	2	3	4	5	6	7	8
систем	управл	0	5	9	20	66	19	69	1
замкнут	систем	0	0	1	1	2	1	0	0
разомкнут	систем	0	0	1	0	21	0	0	0
сау		0	5	0	0	298	0	4	0
крив	разгон	0	0	0	0	61	0	0	0
автомат	стабилиз	0	0	0	0	6	0	0	0
порядок	статизм	0	0	0	0	0	0	0	0
следящ	систем	0	0	0	0	17	0	0	0
экстрем	управл	0	1	0	0	0	0	0	0
адаптив	систем	0	0	0	0	1	0	3	0
импульс	систем	0	0	0	0	0	0	0	0
импульс	сау	0	0	0	0	0	0	0	0
непрерывн	сау	0	0	0	0	0	0	0	0
непрерывн	систем	1	0	4	0	0	0	0	0
статист	сау	0	0	0	0	0	0	0	0
динам	сау	0	0	0	0	2	0	0	0
линейар		0	0	1	7	0	0	0	1
модель	сау	0	0	0	0	1	0	0	0
преобраз	лаплас	0	0	0	0	7	0	0	0
передат	функц	0	0	0	0	129	0	0	0
весов	функц	0	0	1	4	0	0	0	0
контур	сау	0	0	0	0	12	0	0	0
закон	управлен	0	4	0	4	176	0	3	0
элементар	звен	0	0	0	0	9	0	0	0
интегрир	звен	0	0	0	0	10	0	0	0
дифференц	звен	0	0	0	0	2	0	0	0
апериодич	звен	0	0	0	0	14	0	0	0
колебат	звен	0	0	0	0	0	0	0	0
управляем	сау	0	0	0	0	0	0	0	0
наблюдаем	сау	0	0	0	0	0	0	0	0
устойчив	сау	0	0	0	0	6	0	0	0
гурвиц		0	0	0	6	0	0	0	3
годограф		0	0	0	0	0	0	0	0
критер	михайлова	0	0	0	0	0	0	0	0
критер	найкриста	0	0	0	0	0	0	0	0
показател	качеств	0	5	1	4	8	0	2	1
инвариантн		0	0	1	0	0	0	4	0

Рис. 1. Контент-анализ терминологической связи между учебными дисциплинами

4. На заключительном этапе происходит обработка результатов контент-анализа. На основании частоты встречаемости термов можно сделать выводы о степени терминологической связности учебных дисциплин между собой. В случае недостаточной связности близких дисциплин возможен выход не только в наполнении их общим терминологическим словарем, но и в наполнении общим материалом, что позволит сблизить дисциплины между собой. Например, при изложении лекционного или практического материала дисциплины «Математический анализ» предлагается использовать в качестве абстрактных примеров графиков, решений или формул конкретные понятия, изучаемые в других дисциплинах, например «Моделирование систем» или «Основы теории управления», что позволит не только сблизить сами дисциплины, но и облегчить восприятие материала студентами при прослушивании и той и другой дисциплины. В качестве примера полученных результатов обработки, представленных на рис. 2, была взята оценка взаимосвязи между учебной дисциплиной «Основы теории управления» и некоторыми другими специальными дисциплинами направления «Ин-

форматика и вычислительная техника». Анализ показал, что терминологическая связь между дисциплинами «Основы теории управления» с одной стороны и «Математические основы описания процессов управления в пространстве состояний» и «Теоретические основы автоматизации управления» с другой, наиболее сильная (14,24 % и 10,91 % терминов одной дисциплины встречаются в лекционном материале другой соответственно).

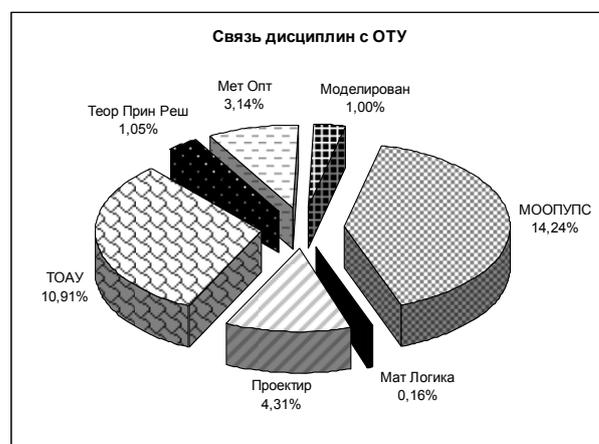


Рис. 2. Анализ взаимосвязи между дисциплиной «Основы теории управления» и некоторыми другими специальными дисциплинами

В то же время связь между дисциплиной «Основы теории управления» и дисциплиной «Математическая логика» практически отсутствует (всего 0,16 % встречаемости терминов).

Вывод

Методы контент-анализа, естественно, не могут дать исчерпывающую оценку степени взаимосвязи между учебными дисциплинами, в частности они не позволяют производить анализ содержимого математических формул в тексте. Но контент-анализ может служить дополнением к другим методикам управления учебным процессом, в частности к методике

экспертных оценок, с целью исключения некоторой субъективности в оценках экспертов.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Когнитивный анализ качества подготовки специалистов в вузах / В.А. Камаев [и др.] // Современные наукоемкие технологии. – 2005. – № 6. – С. 25–26.
2. Трофимова, О. К. Автоматизация процесса составления учебных планов вузов: автореф. дис. ... канд. техн. наук / О. К. Трофимова. – М., 1999. – 25 с.
3. Скрипченко, Е. Н. Контент-анализ учебно-методических комплексов с целью совершенствования междисциплинарных связей при подготовке инженеров-системотехников / Е.Н. Скрипченко, И.М. Харитонов // Информационные технологии в образовании, технике и медицине: матер. Междунар. конф., г. Волгоград, 21–24 сентября 2009 г. / ВолгГТУ [и др.]. – Волгоград, 2009. – С. 122.

УДК 004.55, 378.147

О. А. Шабалина, А. В. Катаев, П. Н. Воробкалов

ОБУЧЕНИЕ РАЗРАБОТЧИКОВ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ: ПРИМЕНЕНИЕ КОМПЬЮТЕРНЫХ ИГР И ПРОЦЕССА ИХ РАЗРАБОТКИ

Волгоградский государственный технический университет

O.A.Shabalina@gmail.com, alexander.kataev@gmail.com, pavor84@gmail.com

В данной статье авторы рассматривают различные возможности применения компьютерных игр для обучения разработке программного обеспечения, в том числе не только для подачи обучающего материала в компьютерных играх, но и в качестве заданий студентам на разработку. Описан опыт применения данного подхода в процессе подготовки студентов в ВТУЗе.

Ключевые слова: электронное обучение, компьютерная игра, разработка программного обеспечения.

O. A. Shabalina, A. V. Kataev, P. N. Vorobkalov

TEACHING SOFTWARE ENGINEERS: USING GAMES AND GAME DEVELOPMENT PROCESS

Volgograd state technical university

In the paper authors consider different possibilities of using computer games for teaching software development, not only presenting learning material through computer games, but also as giving computer games development as assignments for students. Practical experience of using of this approach for teaching students in technical university is described.

Key words: eLearning, computer game, software development.

Уровень развития информационных технологий (ИТ) сегодня определяет статус страны. В настоящее время невозможно найти сферу деятельности, в которой они так или иначе не применяются. Решение задач, связанных с разработкой и внедрением информационных систем требует привлечения высококвалифицированных ИТ специалистов. Обучение ИТ специалистов имеет свою специфику. Информационные технологии развиваются очень быстрыми темпами, существующие решения быстро устаревают и теряют актуальность, появляются новые задачи, решение которых требует разработки новых методов, программных средств и технологий. Студенты должны усваивать и пере-

рабатывать огромный и постоянно растущий объем информации, быстро обучаться новым продуктам и технологиям.

Анализ состояния ИТ рынка [1, 2] показывает несоответствие структуры спроса и предложения на рынке труда. Причем проблемой является не только нехватка ИТ специалистов, но также и недостаточный уровень их подготовки. Работодатели, представляющие различные сектора ИТ-индустрии, в качестве основных недостатков в подготовке ИТ специалистов отмечают слабое знание современных технологий и языков программирования, неумение применять теоретические знания для создания реальных проектов, неумение работать в команде,

отсутствие навыков самообучения, технического и организационного лидерства [3]. Ключевую роль в информационных технологиях играет программное обеспечение (ПО).

В последние годы заметной становится тенденция к снижению эффективности методов, традиционно применяемых при обучении студентов. Современные студенты не ощущают интереса к учебе, находят ее сложной и скучной, они никак не мотивированы. В частности, при обучении будущих специалистов в области информационных технологий собственно информационные технологии используются недостаточно, предпочтение отдается классическим методам обучения. Решение этих проблем требует разработки и внедрения новых эффективных подходов к обучению разработчиков ПО, использующих современные возможности информационных технологий.

Существуют различные способы повышения эффективности обучения, и один из наиболее перспективных подходов – это использование в обучении компьютерных игр (Digital Game Based Learning, DGBL) [4]. Обучающие игры применяются в различных областях, таких, как экономика, маркетинг, иностранные языки, разработка программного обеспечения для обучения студентов, повышения квалификации сотрудников компаний, самообразования.

Анализ существующих игр для разработчиков ПО показывает, что в большинстве своем они предназначены либо для обучения технологиям и языкам программирования, либо для обучения процессу разработки ПО. Игры для обучения языкам программирования можно разделить на две группы: игры, которые ориентированы в первую очередь на изучение объектной технологии программирования, и игры, в которых необходимо реализовывать алгоритмы на языке программирования. Игры для изучения технологий разработки программных систем разрабатываются в университетах, и предназначены для образовательных целей. В основе всех игр по изучению технологий программирования лежит визуализация решений по разработке программного кода, что помогает обучаемым понять абстракции объектно-ориентированного подхода. К таким играм относятся игры на основе библиотеки Turtle graphics, различные модификации игры Karel The Robot, среды разработки и визуализации кода BlueJ, DrJava, Alice и др. Рассмотренные игры представляют собой игровые платформы (frameworks)

для создания объектных миров или среды разработки (IDE), или их комбинации. Игровые платформы имеют хорошие возможности для визуализации объектной парадигмы. Среда разработки, в свою очередь, предоставляют богатые возможности для изучения синтаксиса и семантики программных языков, позволяют экспериментировать с программными концептами (создание объектов, вызов методов), и также тренируют навыки работы собственно со средами разработки.

Известные игры для обучения программированию в большинстве своем представляют собой игры стратегии, в которых необходимо разрабатывать алгоритмы для управления игровыми объектами. Наиболее известными этого жанра являются игры Colobot, и ее развитие игра Ceebot, RoboCode, Robot Battle и др. С точки зрения обучения программированию игры для программистов можно разделить на две категории – это игры для изучения основ программирования и соревновательные игры уже для состоявшихся программистов. Игры для начинающих программистов используют свой собственный, упрощенный язык программирования, обладающий упрощенным синтаксисом и ограниченными возможностями, что позволяет понять основы программирования. Вторая группа игр предназначена в большей степени для соревнования, чем для обучения. В качестве языка обучения в большинстве игр используется язык Java или язык платформы .NET, так как они широко распространены и программы на этих языках могут быть легко скомпилированы динамически и результаты работы могут быть проверены. Для изучения технологий программирования эти игры явно не предназначены, хотя в процессе написания кода игрок реализует соответствующие алгоритмы управления объектами. Игры для программистов также не включают в себя теоретический материал в полном объеме. Цель таких игр – либо помочь начинающим программистам понять базовые принципы программирования, либо дать возможность опытным программистам развить свои навыки и продемонстрировать их в соревновании с другими участниками.

Практически все игры для обучения процессу разработки ПО создаются преподавателями, научными сотрудниками и аспирантами университетов. В качестве примера можно привести такие игры, как SESAM, SimSE, RPG SE, SimVBSE, SimjavaSP и др. Объектом изучения

во всех рассмотренных играх являются процессы, соответствующие определенным этапам жизненного цикла ПО. Этим определяется выбор способов интеграции обучающего контента в игровой процесс – это или симуляция процессов или обучение на примерах. Обучение на примерах – это известный «case study» подход, который реализуется с использованием возможностей компьютерных игр (организация диалогов, визуализация персонажей, событий и среды обитания и т. д.). Эти игры, как правило, реализуются в ролевом (role-playing) или приключенческом (adventure) жанре. Игрок помещается в виртуальную среду, созданную на основе описаний ситуаций из реальной действительности. Игрок действует в соответствии с заложенным в игру сценарием (планом действий), ведет предварительно предусмотренные в игре диалоги, выбирает свои решения из заданных наборов и в итоге обучается на примерах правильных решений. Обучение на примерах показывает свою эффективность при отработке выполнения некоторой последовательности действий в определенных ситуациях. Но при этом, возможности игрока ограничиваются ситуациями, которые предусмотрены и описаны разработчиками. Процесс проектирования таких игр включает разработку описания, игрового сюжета на основе сценария, разработку виртуальной среды и способа ее отображения.

Другой подход – имитация действительности (simulation) используется в широком спектре обучающих игр. К этому классу игр можно отнести симуляторы управления различными транспортными средствами, медицинские, тактические, социальные и бизнес-симуляторы. Игры-симуляторы моделируют реальные условия профессиональной деятельности специалиста в различных областях. Большинство рассмотренных игр для обучения разработке ПО используют именно этот подход. Имитационный подход предоставляет игроку большую свободу действий. Используемые в таких играх модели позволяют отработать профессиональные навыки, при этом отсутствуют риски, связанные с работой в реальных условиях. Проектирование таких игр включает разработку модели процесса работы в соответствующей предметной области, которая потом реализуется в игре. Ограничения этого подхода связаны с тем, что для некоторых дисциплин построение подобной модели является чрезвычайно трудоемким.

Игровая компонента в обоих случаях заключается в основном именно в использовании богатых возможностей компьютера (графика, звук, эффекты, анимация) для организации процесса обучения. Основной задачей проектировщиков таких обучающих игр является выбор и реализация способов наиболее привлекательного с игровой точки зрения воспроизведения сценария (для случая case study) или модели (для симулятора).

Применение существующих игр в учебном процессе показывает, что игры, реализующие case-study подход, и игры-симуляторы, позволяют успешно развивать практические навыки, необходимые для специалистов в различных предметных областях. При этом знания, необходимые для развития навыков, игрок должен либо извлекать самостоятельно (симуляторы), либо получать уже готовые правильные решения (case-study), которые он может потом использовать для решения подобных задач. Необходимый объем теоретического материала, составляющего содержание изучаемого курса, в игровом контексте явно не присутствует (он может присутствовать в виде отдельных блоков контекстной помощи или в виде учебника, встроенного в игру также в виде помощи). И, главное, – конечной целью реального проекта является его реализация и внедрение, а данные стадии разработки ПО не могут быть полностью воспроизведены в играх.

Для обучения будущих разработчиков ПО компьютерные игры также используются в качестве проектов для разработки ПО на реальных задачах. Анализ результатов обучения, в котором используется разработка игр, показывает, что такой подход повышает мотивацию студентов и позволяет повысить уровень подготовки разработчиков ПО.

Изучение базовых дисциплин по направлению «552800 Информатика и вычислительная техника» непосредственно связано с разработкой программного обеспечения. Дисциплина «Программирование на языке высокого уровня» связана с изучением синтаксиса программных языков, алгоритмизации и реализации базовых алгоритмов на языке высокого уровня. Компьютерная графика включает изучение способов представления виртуального пространства, математических основ и методов обработки графики на компьютере, типовых архитектур графических систем, графических прикладных программных интерфейсов, и раз-

работку программ, реализующих алгоритмы компьютерной графики. Изучение операционных систем подразумевает изучение базовых принципов построения операционных систем, управления выполнением задач, функционирования драйверов, разработку программ системного уровня. Дисциплина «Базы данных» требует изучения реляционной алгебры, способов хранения, индексирования и поиска данных, программирования на языке SQL, и заканчивается курсовым проектом по разработке баз данных. «Сети ЭВМ и телекоммуникации» – это изучение современных архитектур сетей и средств их построения, принципов их организации, уровней и протоколов взаимодействия в сети, разработка программ для проектирования сетей и решения коммуникационных задач. «Методы и средства защиты компьютерной информации» включают изучение схем шифрования, протоколов защищенного обмена сообщениями, аутентификации и авторизации, а также программной реализации криптографических алгоритмов и схем защищенного взаимодействия. Таким образом, изучение всех базовых курсов связано с освоением теоретического материала, а также разработкой и реализацией программ, в которых необходимо использовать полученные знания.

Причем для всех дисциплин характерно отсутствие четкой границы между теорией и практикой (они имеют явно выраженный прикладной характер). Поэтому наилучшего результата обучения можно добиться, если студенты будут иметь возможность сразу же после изучения нового материала попробовать применить полученные знания. Для закрепления навыков необходимо многократно решать различные практические задачи, в том числе самостоятельно. Но студенты не всегда готовы тратить достаточно много времени на освоение навыков, они находят это занятие скучным. В итоге они могут успешно сдать экзамен, запомнив нужный объем информации, но оказываются не готовы к применению полученных знаний в реальной жизни.

Для того, чтобы мотивировать студентов не останавливаться после изучения и запоминания теоретического материала, а прилагать усилия к осмыслению этого материала, сфер его применения для решения реальных задач, и тренировки в решении таких задач, нужно дать им понять, зачем изучается тот или иной раздел курса. Среди известных парадигм обу-

чения этой задаче в наибольшей степени отвечает деятельностная парадигма («learning by doing»), в основе которой лежит связь теории и практики в обучении [5]. Деятельностная парадигма ориентирована на овладение навыками определенных видов (или одного вида) деятельности. Последовательность действий в процессе обучения на основе этой парадигмы должна обосновывать необходимость изучения теории для того, чтобы решить некоторую задачу. Но сама необходимость решения задачи при этом в большой степени постулируется, предьявляется как данность, и это может быть препятствием для студента, он не мотивирован решать эту задачу.

В реальном мире разработчик программного обеспечения должен определять задачи самостоятельно, для этого необходимо обладать навыками анализа потребностей, выявления проблем и изучения способов решения этих проблем. Для такого обучения студента нужно погрузить в виртуальную ситуацию, в которой он будет понимать необходимость в решении этой задачи, в получении необходимых для этого знаний, осознавать возможные последствия решения задачи. Такую виртуальную ситуацию можно реализовать в виде игры, в которой игровой результат, и даже сама возможность продолжать игру будут зависеть от результатов процесса обучения в игре. Для этого обучающие материалы должны быть, таким образом, интегрированы в игровой контекст, чтобы игрок воспринимал их как необходимые элементы игры, и осознавал необходимость их изучения. Таким образом, для обучения базовым дисциплинам с использованием игрового подхода необходима разработка игр, которые учитывают специфику дисциплин, входящих в программу подготовки разработчиков ПО. Комплекс игр должен включать в себя игры для изучения базовых дисциплин и тренировки навыков программирования в рамках содержания этих дисциплин.

После изучения базовых дисциплин студенты изучают процесс проектирования программных систем, и применяют полученные знания и навыки для разработки собственных программных проектов в рамках выпускной работы бакалавра. Процесс проектирования ПО включает в себя определенную последовательность стадий и этапов, регламентированных российскими и международными стандартами на проектирование автоматизированных систем. Обуче-

ние процессу разработки программного обеспечения (ПО) является сложной задачей, понимание процесса проектирования программных систем и технологий программирования требует высокого уровня абстракции, без прямой связи с проблемами реального мира. Также требуются значительные усилия для того, чтобы приобрести практические навыки. Подавляющая часть современных программных систем разрабатывается в группах, и именно командная работа представляет наибольшие трудности для начинающих инженеров. Но работа над реальными проектами может потребовать нереально большого времени, кроме того, сбор и анализ требований к программной системе также является сложной задачей. Для этого необходимо организовывать реальные связи с заказчиками, и не всегда можно найти нужное количество таких заказчиков, которые готовы доверить работу студенту.

Студенты, будущие разработчики ПО, вряд ли смогут довести свою работу до приемлемого уровня, так они еще не обладают достаточными навыками. Поэтому, в качестве заказчиков выступают сами преподаватели, и проекты становятся условно реальными. При этом снижается эффективность работы студентов на этапе сбора и анализа требований, и мотивация такой работы часто оказывается недостаточной высокой.

Еще одной проблемой обучения специалистов в области разработки ПО является то, что задания по проектированию, которые выполняют студенты в ходе обучения, как правило, уже формализованы и значительно менее сложны, чем реальные задачи, в качестве объектов проектирования в них предлагаются упрощенные (адаптированные к учебному процессу) системы. Для разработки современных программных систем требуется гораздо более высокий уровень квалификации, и только те студенты, которые много занимаются дополнительно и/или приобретают практический опыт, работая над реальными заказами, достигают такого уровня во время обучения в вузах.

Использование игр в качестве объекта проектирования позволяет в той или иной мере решить рассмотренные проблемы. Компьютерные игры представляют собой программные системы, и процесс разработки игр включает все этапы проектирования ПО. Однако процесс разработки игр должен учитывать специфику игр, как класса программных систем, так как

проектировщики игр привыкли оперировать такими понятиями, как game-концепт (общее описание игры), feature-лист (список функций), design-документ (техническое задание), целевая аудитория (потребители игрового программного продукта). Анализ процесса проектирования игр позволяет соотнести стадии и этапы проектирования игр и соответствующими шагами процесса проектирования ПО. Проектирование игры позволяет изучить все этапы процесса проектирования ПО на реальных задачах и довести этот процесс до стадии реализации системы.

Использование процесса разработки игр в обучении разработчиков ПО имеет ряд важных преимуществ в сравнении с другими подходами. Во-первых, цель разработки такого ПО понятна и близка большинству студентов. Во-вторых, многие из них хорошо знакомы с этой областью и могут сформулировать адекватные требования к таким программным системам. И также очень важно то, что разработка компьютерных игр требует командной работы, что в настоящее время является очень важным для разработчика ПО. Разработка компьютерных игр развивает такие необходимые в работе навыки, управление командной разработкой, выставление приоритетов и планирование, решение конфликтных ситуаций. Использование обучающих игр в качестве объектов проектирования позволяет организовать такие важные этапы процесса разработки ПО как тестирование на целевых группах и внедрение игр в процесс обучения студентов, будущих разработчиков ПО.

Однако применение игр и в качестве средства обучения, и в качестве объекта проектирования требует решения ряда проблем. Для обучения разработчиков ПО необходимы игры, обучающий контент которых соответствует образовательным стандартам. Для разработки обучающей игры требуется модификация процесса проектирования обычной игры с учетом разработки обучающей компоненты и механизмов ее интеграции в игру, и также разработки методов интеграции обучающего курса в игру, развития навыков и организации тестирования в игровом контексте, а также оценки уровня знаний обучаемых. Кроме того, чтобы успешно конкурировать по привлекательности с обычными играми, обучающие игры не должны уступать им по уровню графических и звуковых эффектов, проработанности сюжета и ка-

честву реализации. Поэтому для разработки обучающих игр необходимо привлекать специалистов – разработчиков программного обеспечения, знакомых с современными технологиями в области разработки игр, а так же специалистов, имеющих непосредственное отношение к обучению студентов. С другой стороны при использовании игр в качестве объектов проектирования при обучении студентов необходимо найти потенциальных потребителей, которые помогут сформулировать требования к игре и обеспечат тестирование игры.

Для решения этих проблем предлагается в рамках обучения студентов процессу разработки ПО разрабатывать обучающие игры, предназначенные для обучения таким дисциплинам, как «Компьютерная графика», «Программирование», «Базы данных» и др. В этом случае потребителями таких игр будут преподаватели и студенты младших курсов. Преподаватели знают требования соответствующих образовательных стандартов к содержанию дисциплин, они могут выступать в качестве разработчиков контента. Студенты младших курсов будут привлекаться для тестирования разработанных игр.

Те студенты, которые будут заниматься разработкой обучающих игр, получают возможность изучить все этапы процесса проектирования ПО на реальных проектах, работать с реальными заказчиками, тестировать свои игры на будущих потребителях. Кроме того, разработка игр требует координированной работы различных специалистов, навыков работы в команде и управления проектами, и сту-

денты смогут получить реальный опыт такой работы.

Компьютерные игры относятся к классу программных систем высокой сложности. Разработка игр требует высокой квалификации. Проектирование игр, как программных систем, требует умений анализа требований к программной системе, знания методологий проектирования ПО, навыков программирования, особенностей технических средств и сред разработки. Кроме того, реализация компьютерных игр требует знаний в области разработки интерактивных приложений, двумерной и трехмерной компьютерной графики, методов лексического и синтаксического анализа выражений на формальном языке, алгоритмов поиска пути в виртуальном пространстве, кросс-платформенного программирования. В той или иной степени, соответствующие разделы включены в образовательный стандарт и входят в программы изучения дисциплин «Программирование на языках высокого уровня», «Базы данных», «Искусственный интеллект», «Компьютерная графика», но этого объема недостаточно для разработки игр. Поэтому студенты, выполняющие проекты по разработке игр, самостоятельно осваивают дополнительный материал и, таким образом, существенно повышают свою квалификацию.

Таким образом, предложенный подход основан на использовании обучающих игр для разработчиков ПО как в качестве объекта проектирования при обучении процессу разработки ПО, так и в качестве средства обучения (базовым дисциплинам) (рисунок).

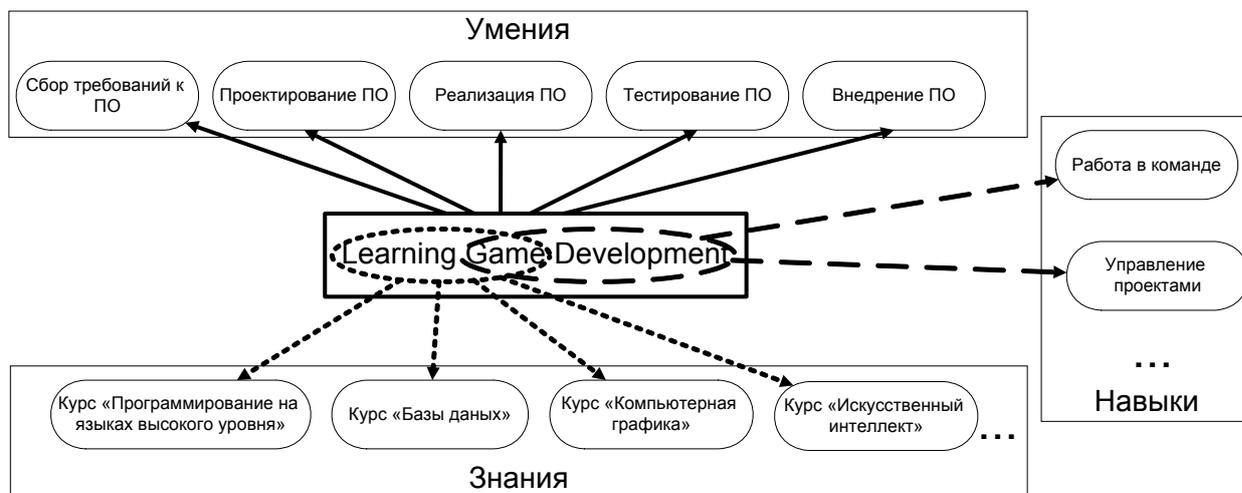


Рис. Способы применения компьютерных обучающих игр для разработчиков ПО

На кафедре САПР и ПК в течение нескольких лет для обучения разработке ПО используются обучающие компьютерные игры. В начале процесса обучения компьютерные игры применяются для обучения студентов базовым дисциплинам и тренировки навыков. Такой подход помогает вовлечь студентов в процесс обучения, и повысить их мотивацию к учебе. Затем, используя полученные знания, студенты получают возможность разрабатывать свои собственные обучающие компьютерные игры в рамках курсовых проектов и бакалаврских работ. Разработанные обучающие игры тестируются на студентах младших курсов, дорабатываются и внедряются в учебный процесс.

В настоящее время разработаны игра для изучения объектно-ориентированной технологии проектирования и языку C# [6,7], несколько игровых тренажеров для тренировки навыков программирования на языке C++ [8,9, 10], ряд игр под мобильные платформы [11, 12]. Трехлетний опыт работы в области создания обучающих игр позволил накопить достаточный опыт проектирования и создать базу компонентов, пригодную для использования в последующих разработках.

Анализ результатов применения разработки игр для обучения разработчиков ПО показал, что студенты, участвующие в проектах, получили знания и навыки, выходящие за рамки учебной программы, и значительно повысили уровень квалификации (см. таблицу).

Студенты, принимавшие участие в проектах, отмечают также, что в процессе работы они приобрели такие важные профессиональные навыки, как работа над реальными задачами, проектирование, тестирование, отладка и реализация программных систем, работа с чужим кодом, в том числе с открытыми библиотеками, работа с системами контроля версий и другими современными средствами разработки. В процессе разработки прототипы игр тестировались на целевых группах студентов кафедры САПР и ВолгГТУ, результаты апробации учитывались для следующих итераций разработки обучающих игр.

Не менее важным результатом выполнения проектов студенты считают такие необходимые в работе навыки, как умение работать в команде, управление командной разработкой, выставление приоритетов и планирование, решение конфликтных ситуаций.

Таблица

Знания и навыки, полученные студентами, участвовавшими в разработке игр

	Изучаемые в рамках учебной программы	Полученные в результате разработки игр
Знания по разделам	Программирование на языках высокого уровня (C++) Базовые понятия компьютерной графики Основы трансляции Физика, численные методы Теория графов Методы искусственного интеллекта Разработка баз данных	Программирование с использованием языка C#. Работа с трехмерными графическими API DirectX и OpenGL Использование скриптовых языков программирования и динамической компиляции кода Численное моделирование физического взаимодействия объектов в виртуальном мире. Поиск пути в трехмерном пространстве Разработка искусственного интеллекта персонажей игры Разработка структуры хранения данных об игровых объектах для описания игровых уровней, загрузки и сохранения игры
Языки	C++ SQL	C# LINQ to SQL Lua High Level Shader Language GL Shader Language
Технологии	Процедурное программирование Объектно-ориентированное программирование	Паттерны проектирования COM-технология .NET

Опыт разработки проектов по созданию игровых приложений под различные платформы показал, что применение компьютерных игр для обучения разработчиков ПО является очень эффективным как с точки зрения повышения качества подготовки, так и повышения мотивации студентов к обучению.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Специфика обучения ИТ-специалистов в России. Режим доступа: <http://www.academy.ru/t.asp?id=115&checked=yes> – 2010 г.
2. Holliss, M. A study on the IT labour market in the UK / M. Holliss, M. Cadby. – Режим доступа: <http://www.cphc.ac.uk/docs/reports/cphc-itlabourmarket.pdf>. – 2008.

3. *Левкович-Маслюк, Л.* ИТ-образование: индустриальный угол / Л. Левкович-Маслюк // Журнал «Компьютерра», № 23. – 27 июня 2007 г.
4. *Prensky, M.* Digital Game-Based Learning: Paragon House Ed. – Paragon House Publishers, 2007. – 442 с.
5. *Charles, M.* Reigeluth Instructional-design Theories and Models: A new paradigm of instructional theory New York: HarperCollins. – November. – 1995.
6. *Shabalina, O.A.* et al. “Educational Games for Learning Programming Languages”, Methodologies and Tools of the Modern (e-) Learning: suppl. to Int. Journal «Information Technologies and Knowledge», Vol. 2, Int. Book Series «Inform. Science & Comput.», No 6, pp 79–83. – 2008.
7. *Shabalina, O.* 3I-Approach for IT Educational Games Development / Olga Shabalina, Pavel Vorobkalov, Alexander Kataev, Alexey Tarasenko // Proceedings of the 3rd European Conference on Games-Based Learning, Graz, Austria, 12–13 October 2009 / FH JOANNEUM University of Applied Science. – [UK], 2009. – P. 339-344.
8. *Шабалина О. А.* Разработка средств обучения кодированию на языках высокого уровня с помощью игровых тренажеров / А.С. Соловьев, А.А. Алимов, О.А. Шабалина // Городу Камышину – творческую молодежь (посвящается 15-летию Камышинского технол. ин-та (филиала) ВолгГТУ): матер. III регион. н.-практ. студ. конф., 22–23 апр. 2009 г. / ВолгГТУ, КТИ (филиал) ВолгГТУ. – Камышин, 2009. – Т. 2. – С. 96–98.
9. *Соловьев, А.С.* Разработка игровых тренажеров для обучения программированию / А. С. Соловьев, А. А. Алимов // Третья международная научно-техническая конференция «Инфокоммуникационные технологии в науке, производстве и образовании (ИНФОКОМ-3)», г. Кисловодск, 1–5 мая 2008 г./ СевКавГТУ, филиал Ставропольского государственного технического университета – С. 255–267.
10. *Шабалина О.А.* Разработка комплекса средств для обучения алгоритмизации и языкам программирования с использованием игрового подхода / А. С. Соловьев, Д. А. Ересько, А. А. Алимов, О. А. Шабалина // Единое образовательное пространство славянских государств в XXI веке: проблемы и перспективы: мат. III Международной научно-практической конференции в г. Брянск 2–3 апр. 2009. – Брянск, 2009.
11. *Shabalina, O.* Game for learning logical design / Dmitry Eresko, Olga Shabalina // Proceedings of IADIS International Conference Mobile Learning 2010 – Porto, Portugal, 2010.
12. *Shabalina, O.* Mobile Learning Games for Primary Education / Olga Shabalina, Pavel Vorobkalov, Alexander Kataev, Alexey Tarasenko and Alexander Arsentiev // Proceedings of the 4th European Conference on Games-Based Learning, Copenhagen, Denmark, 21–22 October 2010 / The Danish School of Education. – [UK], 2010. – P. 350–359.

АВТОМАТИЗИРОВАННОЕ ПРОЕКТИРОВАНИЕ

УДК 62-52

М. В. Голубятникова

ПРИМЕНЕНИЕ АЛГОРИТМА ВЫБОРА ОПТИМАЛЬНОГО ТЕХНОЛОГИЧЕСКОГО РЕЖИМА ПРИ НЕЧЕТКО ВЫРАЖЕННЫХ ЭКСПЕРТНЫХ ОЦЕНКАХ

Астраханский государственный технический университет
m.golubyatnikova@astu.org

Рассматривается применение метода анализа иерархий в задачах выбора оптимального технологического режима при нечетко выраженных экспертных оценках. Выполнена модификация метода, пригодная для обработки оценок парных сравнений, выраженных в лингвистической форме. Приведен пример выбора оптимального технологического режима для процесса получения экстракта солодкового корня *Extractum Glycyrrhizae*.

Ключевые слова: метод анализа иерархий, лингвистические описания, экспертные оценки, процесс экстракции.

M. V. Golubyatnikova

APPLICATION OF OPTIMUM TECHNOLOGICAL MODE CHOICE ALGORITHM FOR FUZZY EXPERT ESTIMATIONS

Astrakhan state technical university

Application of a method of the analysis of hierarchies in problems of a choice of an optimum technological mode is considered at indistinctly expressed expert estimations. Method updating, suitable for processing of estimations of the pair comparisons expressed in the linguistic form is executed. The example of a choice of an optimum technological mode for *Extractum Glycyrrhizae* extraction process is resulted.

Key words: method of hierarchies analysis, linguistic descriptions, expert estimations, process extraction.

При использовании в системах автоматического управления математических моделей, способных обрабатывать как количественную, так и качественную информацию о процессе, возникает ряд трудностей. Обработка качественной информации в таких моделях строится, чаще всего, с использованием математического аппарата нечетких множеств. Наличие верхней и нижней грани нечеткого отношения приводит к невозможности применения традиционных методов поисковой оптимизации. Тем не менее, эта задача является актуальной для автоматизации целого ряда различных технологических процессов.

Для решения этой задачи необходимо определить алгоритм нахождения вектора оптимальных технологических параметров режима R^* , являющегося оптимальным по произвольно выбранным критериям сравнения, в случае

представления оценок сравнения в виде лингвистических описаний.

При этом в случае количественных оценок традиционно применяется метод анализа иерархий [1], в котором используются парные оценки альтернативных режимов R_1, R_2, \dots, R_m по априорно выбранным факторам F_1, F_2, \dots, F_n . Основной проблемой в этом случае является получение экспертных оценок парных сравнений. В большинстве случаев эксперты затрудняются при сравнении выразить приоритет доминирования в числовой форме, что приводит к необъективным оценкам.

В связи с этим предлагается выражать оценки приоритетов в лингвистической форме, например, «более предпочтительно», «значительно более предпочтительно», «менее предпочтительно» и т. д. Для обработки такой информации используется математический аппа-

рат нечетких множеств. В качестве универсального множества может использоваться непрерывный числовой диапазон, например, $U \in [0,9]$. При этом $U=0$ соответствует абсолютному доминированию второй альтернативы над первой, $U=9$ – абсолютному доминированию первой альтернативы над второй, $U=4,5$ соответствует полной равнозначности альтернатив. В качестве примера покажем представление лингвистической оценки «менее предпочтительно» (рис. 1).

Для выполнения требования транзитивности и обратной симметричности матриц парных сравнений, полярные оценки типа «более предпочтительно» и «менее предпочтительно» должны быть симметричными относительно центра множества U (рис. 2).

Рассмотрим применение метода анализа иерархий для выбора оптимального технологического режима при лингвистических оценках парных сравнений альтернатив. Для этого структурируем проблему выбора оптимального технологического режима в виде иерархии (рис. 3).

На первом уровне расположена главная цель, на втором уровне находятся факторы сравнения F_1, F_2, \dots, F_n , на третьем уровне – альтернативы технологических режимов R_1, R_2, \dots, R_m , каждая из которых характеризуется соответствующей совокупностью режимных параметров.

В табл. 1 приведен вид матрицы парных сравнений факторов относительно главной цели.

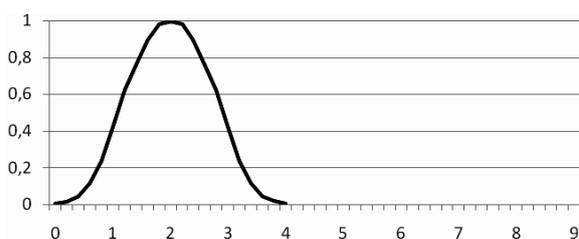


Рис. 1. Графическое представление лингвистической оценки

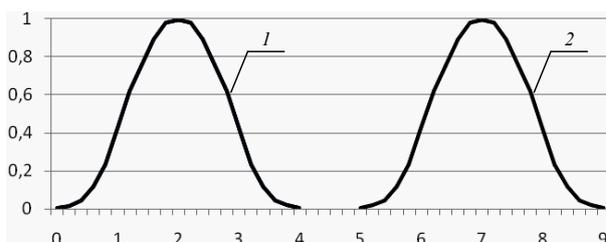


Рис. 2. Графическое представление полярных оценок: 1 – «менее предпочтительно», 2 – «более предпочтительно»

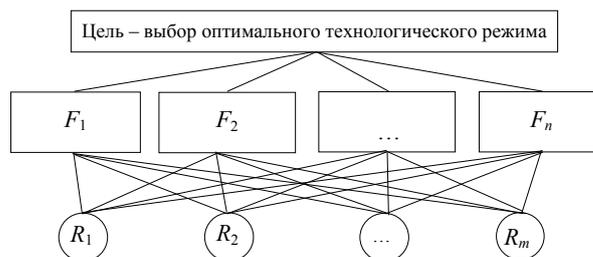


Рис. 3. Иерархия задачи о выборе оптимального технологического режима:

F_1, F_2, \dots, F_n – факторы сравнения, R_1, R_2, \dots, R_m – альтернативы технологических режимов

Таблица 1

Матрица парных сравнений факторов относительно цели

Факторы	F_1	F_2	...	F_n	Вектор приоритетов
F_1	–	X_{F_1/F_2}	...	X_{F_1/F_n}	w_1
F_2	X_{F_2/F_1}	–		X_{F_2/F_n}	w_2
...	–
F_n	X_{F_n/F_1}	X_{F_n/F_2}	...	–	w_n

где X_{F_i/F_j} – приоритет доминирования фактора F_i по отношению к F_j .

Полученная матрица сравнений должна являться диагональной и обладать свойствами транзитивности и обратной симметричности, то есть нечеткое множество X_{F_i/F_j} должно являться симметричным к множеству X_{F_j/F_i} относительно центра U .

Для нахождения вектора приоритетов введем понятие момента нечеткого множества.

Для нечеткого множества X , заданного в непрерывной форме на диапазоне U , момент $M(X)$ определяется как:

$$M(X) = \int_0^u \mu/u \cdot u dU \tag{1}$$

или в дискретной форме

$$M(X) = \sum_{i=0}^u \mu/u_i \cdot u_i \tag{2}$$

Для определения вектора приоритетов вычислим нормированные суммарные моменты лингвистических оценок доминирования факторов $M^*(X_{F_i/F_j})$ в виде

$$M^*(X_{F_i/F_j}) = \sum_{j=1}^n M(X_{F_i/F_j}) / \sum_{i=1}^n \sum_{j=1}^n M(X_{F_i/F_j}) \tag{3}$$

Вектор нормированных суммарных моментов доминирования будет являться вектором приоритетов w_1, w_2, \dots, w_n .

Следует отметить, что оценка приоритета доминирования X_{F_i/F_j} при $i = j$ выражается в виде лингвистического описания «безразлично» и имеет момент относительно центра $M(X_{F_i/F_j}) = 0$, поэтому включение таких оценок в матрицу не имеет смысла и лишь снижает чувствительность метода.

На следующем этапе выполняются парные сравнения альтернатив технологических режимов и вычисление их глобальных приоритетов, что выполняется аналогично.

Рассмотрим представленный алгоритм на примере выбора оптимального технологического режима для процесса экстракции корня солодки. В качестве факторов сравнения альтернативных режимов были выбраны: F_1 – цена экстракта; F_2 – длительность процесса; F_3 – остаточное содержание целевого компонента в шроте; F_4 – качество продукта.

В качестве альтернатив используются три различных технологических режима, которые характеризуются следующим образом:

R_1 – цена получаемого экстракта высокая, длительность технологического процесса большая в связи с низкой температурой экстрагента, остаточное содержание целевого компонента в шроте высокое в связи с низкой интенсивностью процесса, качество продукта среднее;

R_2 – цена получаемого экстракта низкая, длительность технологического процесса малая в связи с высокой температурой экстрагента, остаточное содержание целевого компонента в шроте низкое, качество продукта низкое в связи с разложением высокомолекулярных соединений при высокой температуре;

R_3 – цена получаемого экстракта средняя, длительность технологического процесса средняя, так как процесс проводится при средних температурах, остаточное содержание целевого компонента в шроте низкое, качество продукта высокое.

Даже в столь ограниченном пространстве альтернатив определить оптимальный режим на основе интуитивных оценок достаточно затруднительно, так как субъективные оценки альтернатив входят в противоречие друг с другом.

Матрица парных сравнений факторов относительно главной цели представлена в табл. 2.

Значения лингвистических переменных формализуются на множестве U с использованием в качестве порождающей функции нормального распределения с соответствующим сдвигом среднего значения, приведенное к диапазону $[0 \dots 1]$:

$$\begin{aligned} \text{«Более предпочтительно»} &= \\ &= \frac{2,5}{\sqrt{2\pi}} \int_{-\infty}^x e^{-\frac{(t-7,5)^2}{50}} dt, \end{aligned}$$

$$\begin{aligned} \text{«Менее предпочтительно»} &= \\ &= \frac{2,5}{\sqrt{2\pi}} \int_{-\infty}^x e^{-\frac{(t-1,5)^2}{50}} dt, \end{aligned}$$

$$\begin{aligned} \text{«Значительно более предпочтительно»} &= \\ &= \frac{2,5}{\sqrt{2\pi}} \int_{-\infty}^x e^{-\frac{(t-8,5)^2}{50}} dt, \end{aligned}$$

$$\begin{aligned} \text{«Значительно менее предпочтительно»} &= \\ &= \frac{2,5}{\sqrt{2\pi}} \int_{-\infty}^x e^{-\frac{(t-0,5)^2}{50}} dt, \end{aligned}$$

$$\text{«Безразлично»} = \frac{2,5}{\sqrt{2\pi}} \int_{-\infty}^x e^{-\frac{(t-4,5)^2}{50}} dt.$$

Тогда моменты соответствующих нечетких множеств определены как:

$$M(\text{«Более предпочтительно»}) = 16,5;$$

$$M(\text{«Менее предпочтительно»}) = 3,3;$$

$$M(\text{«Значительно более предпочтительно»}) = 18,7;$$

$$M(\text{«Значительно менее предпочтительно»}) = 1,1;$$

$$M(\text{«Безразлично»}) = 9,9.$$

Значения вектора приоритетов (вектора нормированных моментов) приведены в табл. 2. Видно, что наиболее значимым является фактор F_4 «качество продукта», характеризующий качество готового экстракта. Наименее значимым является фактор F_2 «длительность процесса», определяющий технологические особенности режима.

После расчета по математической модели [2] концентрационных кривых для выбранных режимов R_1, R_2, R_3 проводят сравнения альтернатив. В табл. 3 приведены матрицы парных сравнений режимов технологического процесса и их локальные приоритеты относительно факторов второго уровня иерархии.

Обработка матриц производится аналогично обработке матрицы парных сравнений факторов, результаты обработки в виде векторов нормальных приоритетов приведены в табл. 3.

Таблица 2

Матрица парных сравнений факторов относительно главной цели

Факторы	F_1	F_2	F_3	F_4	Вектор приоритетов
F_1	–	более предпочтительно	более предпочтительно	менее предпочтительно	0,306
F_2	менее предпочтительно	–	менее предпочтительно	значительно менее предпочтительно	0,065
F_3	менее предпочтительно	более предпочтительно	–	значительно менее предпочтительно	0,176
F_4	более предпочтительно	значительно более предпочтительно	значительно более предпочтительно	–	0,454

Таблица 3

Матрица парных сравнений альтернатив

Цена экстракта	R_1	R_2	R_3	Нормированные приоритеты
R_1	–	менее предпочтительно	менее предпочтительно	0,111
R_2	более предпочтительно	–	более предпочтительно	0,556
R_3	более предпочтительно	менее предпочтительно	–	0,333
Длительность процесса	R_1	R_2	R_3	Нормированные приоритеты
R_1	–	значительно менее предпочтительно	менее предпочтительно	0,074
R_2	значительно более предпочтительно	–	более предпочтительно	0,593
R_3	более предпочтительно	менее предпочтительно	–	0,333
Остаточное содержание целевого компонента в шроте	R_1	R_2	R_3	Нормированные приоритеты
R_1	–	значительно менее предпочтительно	менее предпочтительно	0,074
R_2	значительно более предпочтительно	–	безразлично	0,481
R_3	более предпочтительно	безразлично	–	0,444
Качество продукта	R_1	R_2	R_3	Нормированные приоритеты
R_1	–	более предпочтительно	безразлично	0,444
R_2	менее предпочтительно	–	значительно менее предпочтительно	0,074
R_3	безразлично	значительно более предпочтительно	–	0,481

Матрица глобальных приоритетов приведена в табл. 4. Расчет значений глобальных приоритетов осуществляется по Саати [1] традиционным образом.

Видно, что наиболее предпочтительным является режим R_3 с максимальным значением глобального приоритета. Этот режим характеризуется средней ценой получаемого экстракта и мягкостью температурного воздействия, что

предотвращает разрушение высокомолекулярных компонентов экстракта.

Относительно небольшой разброс оценок глобальных приоритетов является следствием разнопланового подхода к оценке оптимальности и объясняет невозможность интуитивного выбора оптимального режима на основе лингвистических описаний альтернатив.

Таблица 4

Глобальные приоритеты альтернатив

Веса критериев	$F_1(0,306)$	$F_2(0,065)$	$F_3(0,176)$	$F_4(0,454)$	Глобальный приоритет
R_1	0,111	0,074	0,074	0,444	0,254
R_2	0,556	0,593	0,481	0,074	0,327
R_3	0,333	0,333	0,444	0,481	0,420

Таким образом, предложенный алгоритм выбора оптимального технологического режима позволит повысить объективность выбора

наилучшего вектора управляющих переменных, при обработке нечетко выраженных экспертных оценок в лингвистической форме.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Саати Т. Л. Принятие решений при зависимостях и обратных связях: аналитические сети. Пер. с англ./ Науч. ред. А. В. Андрейчиков, О. Н. Андрейчикова. – М.: Изво ЛКИ, 2008 – 360 с.
2. Голубятникова М. В., Александян И. Ю., Антонов О. В. Использование комбинированной математической модели для повышения эффективности процесса экстракции растительного сырья. Наука: поиск-2009: сб. науч. ст. / Астрахан. гос. тех. ун-т Астрахань: Изд-во АГТУ, 2009, с. 27–30.

УДК 658.512.011.56:681.3.06

Е. А. Гопта, С. А. Фоменков, Г. А. Карачунова

АВТОМАТИЗАЦИЯ ПРОЦЕССА ЛИНЕЙНОГО СИНТЕЗА ФИЗИЧЕСКОГО ПРИНЦИПА ДЕЙСТВИЯ*

Волгоградский государственный технический университет
gopta.evgeniy@gmail.com, saf@vstu.ru, kar_gal@mail.ru

Рассматриваются аналоги и анализируются недостатки существующих автоматизированных систем синтеза физического принципа действия. Разработан и предложен более эффективный и быстродействующий метод для автоматизации линейного синтеза физического принципа действия.

Ключевые слова: физический эффект, физический принцип действия, условия совместимости, автоматизированная система, линейный синтез, матрица смежности.

E. A. Gupta, S. A. Fomenkov, G. A. Karachunova

AUTOMATION OF LINEAR SYNTHESIS OF PHYSICAL PRINCIPLE OF ACTION

Volgograd State Technical University

Article considered counterparts and examines the shortcomings of existing automated systems of synthesis of the physical principle of action. A more efficient and fast method was developed and proposed to automate the synthesis of linear physical principle of action.

Key words: physical effect, the physical principle of operation, compatibility conditions, automated system, linear synthesis, the adjacency matrix.

В настоящее время одной из важнейших задач является интенсификация создания новых высокоэффективных технических систем (ТС). Сроки от начала разработки до серийного выпуска, сумма затрат на переход к выпуску новой модели, функциональные и экономические характеристики, ряд других важнейших показателей жизнеспособности ТС во многом определяется результатами, достигнутыми на начальных этапах проектирования, когда принимаются основополагающие решения о принципе действия и структуре объекта проектирования.

Известен ряд подходов к реализации начальных этапов проектирования ТС, среди них одним из наиболее перспективных является подход, связанный с привлечением структури-

рованных физических знаний в форме физических эффектов (ФЭ) для автоматизированного формирования и выбора физического принципа действия разрабатываемой ТС [1]. Успешное решение задачи выбора ФПД дает большой экономический эффект, вызывает более заметный технический прогресс в рассматриваемой области и обеспечивает разработку изделий с большим сроком морального старения по сравнению с результатами, получаемыми на этапе выбора технического решения или параметров ТС.

В работах, известных в данном направлении, предполагаются различные модели представления физических знаний в форме ФЭ, разного уровня сложности и специализации алгоритма синтеза и анализа структур ФПД [1, 2].

*Работа выполнялась при поддержке РФФИ (проект № 10-01-00135-а).

Ряд методов включает автоматизированный синтез технических решений для выбранных структур ФПД. В то же время эффективное применение и широкое внедрение данных методов затруднено по ряду причин, главными из которых является:

малая информативность и узкая специализация моделей описания ФЭ;

сложности описания условий решаемой задачи сначала в рамках физических понятий, а затем на языке, определяемом рассматриваемым методом;

несовершенство методов и средств представления результатов синтеза;

сложность оценки адекватности синтезированных структур ФПД решаемой задачи;

недостаточная исследованность области применимости и свойств предлагаемых методов, особенностей получаемых с их помощью решений.

Таким образом, актуальной задачей является совершенствование методов и средств автоматизации синтеза ФПД ТС для повышения эффективности соответствующих проектных процедур начальных этапов проектирования.

На кафедре САПР и ПК ВолгГТУ на протяжении долгих лет также решается задача автоматизированного синтеза ФПД ТС. Существующие алгоритмы оперируют разработанной сотрудниками кафедры обобщенной моделью ФЭ [1, 3, 4].

Под физическим принципом действия понимается структура совместимых и объединенных ФЭ, обеспечивающих преобразование заданного входного воздействия в заданное выходное, при этом два последовательно расположенных ФЭ $W_i = (A_i, B_i, C_i)$ и $W_{i+1} = (A_{i+1}, B_{i+1}, C_{i+1})$ считаются совместимыми относительно базовой модели описания ФЭ, если выходное воздействие некоторого ФЭ цепочки C_i , эквивалентно входному воздействию последующего ФЭ A_{i+1} , то есть:

характер выхода C_i , совпадает с характером входа A_{i+1} , при этом если выход ФЭ направлен на другие (внешние относительно объекта ФЭ) объекты, то выход считается внешним выходом ФЭ (внешний – электрическое поле, электромагнитное излучение, поток нейтронов и т. д.), если выход ФЭ направлен на тот же самый объект, то такой выход считается внутренним (внутренний – изменение температуры, повышение электрического сопротивления и любого другого параметра, характеризующего объект).

Аналогично определяется характер входа произвольного ФЭ;

порядковый номер наименования выхода C_i , совпадает с порядковым номером наименования входа A_{i+1} в списке наименований воздействий;

качественные характеристики выхода C_i совпадают с качественными характеристиками входа A_{i+1} . При этом совпадение осуществляется путем наложения И-ИЛИ дерева входного воздействия на И-ИЛИ дерева выходного воздействия. Перечисленные условия совместимости входного и выходного воздействий реализованы в виде правил сравнения вектора кодировки выхода C_i и вектора кодировки входа A_i ;

если выход C_i , и вход A_{i+1} – внутренние (изменение параметров), то совпадают номера физических величин, характеризующие C_i , и A_{i+1} , представленных в списках физических величин по каждому разделу физики;

если выход C_i и вход A_i являются внутренними, то кроме ранее указанных условий совместимости производится анализ идентичности объектов $B_i(B_i^1, B_i^2)$ и $B_{i+1}(B_{i+1}^1, B_{i+1}^2)$, где $B_i^1, B_i^2(B_{i+1}^1, B_{i+1}^2)$ – соответственно начальное и конечное состояние объекта $B_i(B_{i+1})$ со структурными изменениями объекта, $B_i^1 = B_i^2 (B_{i+1}^1 = B_{i+1}^2)$ у объекта $B_i = B_{i+1}$ без структурных изменений. Объекты должны совпадать на определенном уровне иерархии свойств на и или дереве представления объектов. При этом конечное состояние объекта B_i должно совпадать с начальным состоянием объекта B_{i+1} . Условия совместимости объектов реализованы в виде правил сравнения векторов кодировок объектов $B_i(B_i^1, B_i^2)$ и $B_{i+1}(B_{i+1}^1, B_{i+1}^2)$;

Кроме того, используется следующее правило определения совместимости ФЭ: в случае, когда $(i+1)$ -й ФЭ имеет несколько входов (A_{i+1}^s), совмещение выхода C_i , происходит с любым из входов A_{i+1}^s , удовлетворяющим условию эквивалентности с C_i .

Рассмотрим математическую постановку задачи качественного синтеза ФПД.

Пусть имеется некоторое конечное множество $W = \{W_i | i = 1, \dots, N\}$ физических эффектов базы данных по ФЭ. Сопоставим этому множеству с ориентированным графом $G(V, E)$, где $V = \{v_i | i = 1, \dots, N\}$ – конечное непустое множество вершин графа (v_i , соответствует W_i); E – множество дуг графа:

$$E = \{e_{i,j} = (v_i, v_j) | 1 \leq i, j \leq N\}.$$

Ребро $e_{i,j}$ существует тогда и только тогда, когда эффект W_i совместим с W_j ($1 \leq i, j \leq N$).

ФПД – это путь $v_1, e_{1,2}, v_2, e_{2,3}, \dots, e_{k-1,k}, v_k$ (так как все ребра графа G имеют кратность 1, допустимо обозначение $v_1, v_2, v_3, \dots, v_k$) в графе G длины $k-1$ (k – налагаемое пользователем ограничение на длину искомого ФПД). Путь $v_1, v_2, v_3, \dots, v_k$ определяется заданием начального воздействия A_1 (воздействие A_1 эквивалентно входному воздействию эффекта W_1) и результирующего воздействия C_k (выход эффекта W_k эквивалентен воздействию C_k). Задача синтеза ФПД изделий состоит в нахождении всех таких путей в графе G , удовлетворяющих заданным ограничениям.

Предпринималась также попытка построения ФПД ТС как задачи вывода в системе продукций, что позволяло повысить степень формализации задачи синтеза и качества получаемых решений [5].

В настоящей работе в качестве ближайшего аналога была выбрана хорошо зарекомендовавшая себя на практике система САПФИТ [1]. Детальный анализ интерфейса и алгоритма работы данной системы, выявил ряд существенных недостатков:

любое изменение качественных характеристик входа или выхода приводит к необходимости построения нового графа перехода из входного воздействия в выходное, что существенно замедляет процесс синтеза ФПД.

интерфейс не удовлетворяющим современным стандартам эргономики вызывает неудобства работы с программой у большинства пользователей и делает невозможным использование системы вместе с современными стандартными процедурами.

система САПФИТ не предназначена для использования в сетевом варианте;

существующее математическое обеспечение не позволяет производить построения нелинейных структур ФПД.

В работе предлагается следующая модификация метода построения синтеза ФПД.

На начальном этапе загружается матрица смежности возможных переходов. Матрица смежности – это квадратная бинарная матрица, на пересечении i -й строки и j -го столбца которой стоит 1, если существует ФЭ, который может осуществить переход из i -го воздействия в j -е воздействие, и 0 в противном случае.

На основании введенных пользователем данных входные и выходные параметры, длина цепочки синтеза и матрица смежности, строится линейная структура ФПД. Рассмотрим этот процесс более подробно.

Вершиной линейной структуры ФПД является входное воздействие, которое задается пользователем. Далее, анализ матрицы смежности позволяет определить вершины, в которые возможен переход из одного воздействия в другое. Полученные результаты записываются в список путей перехода от входного воздействия в выходное. На следующем этапе происходит проверка, не является ли длина цепочки синтеза равной количеству переходов заданных пользователем. В случае положительного ответа, полученная цепочка добавляется в список переходов. В противном случае переходят к следующему воздействию и повторяют вышеизложенные процедуры. После того как все переходы между воздействиями будут проанализированы, сохраняется список путей преобразования входного воздействия в выходное.

Далее производится анализ получившихся путей на основе учета типа входов и выходов («внешний», «внутренний»), а также количественных и качественных характеристик. При этом отбрасываются пути перехода от начального воздействия A_1 в результирующее воздействие C_k , в которых не может реализоваться хотя бы один переход из одного воздействия в другое с помощью имеющихся в базе данных эффектов.

В случае удачного построения линейной структуры ФПД, получившаяся линейная структура ФПД отображается на экране. В противном случае, выводится сообщение о невозможности построения линейной структуры ФПД с заданными параметрами начального воздействия A_1 и результирующего воздействия C_k .

Блок-схема алгоритма синтеза ФПД представлена на рис. 1.

Используемые обозначения для описания синтеза линейной структуры воздействий представлены в табл. 1.

Таблица 1

Используемые обозначения

Recurs	Флаг успешного построения синтеза ФПД
Level	Уровень рекурсии
Length	Длина цепочки синтеза ФПД
Vход	Входное воздействие для синтеза ФПД
Vиход	Выходное воздействие для синтеза ФПД
Ways	Массив решений
Way	Массив, содержащий текущий ФПД
Graf	Массив, содержащий матрицу смежности

Более подробное описание синтеза линейной структуры воздействий представлено на рис. 2.

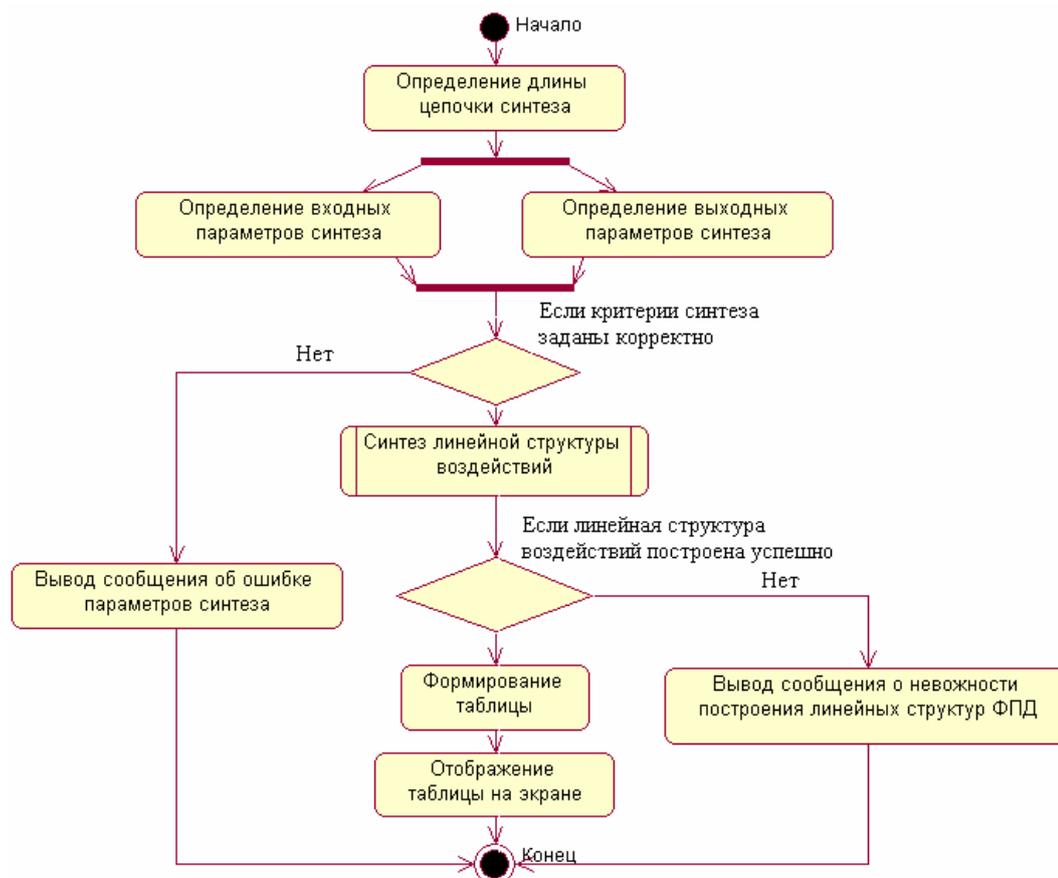


Рис. 1. Блок-схема алгоритма синтеза ФПД

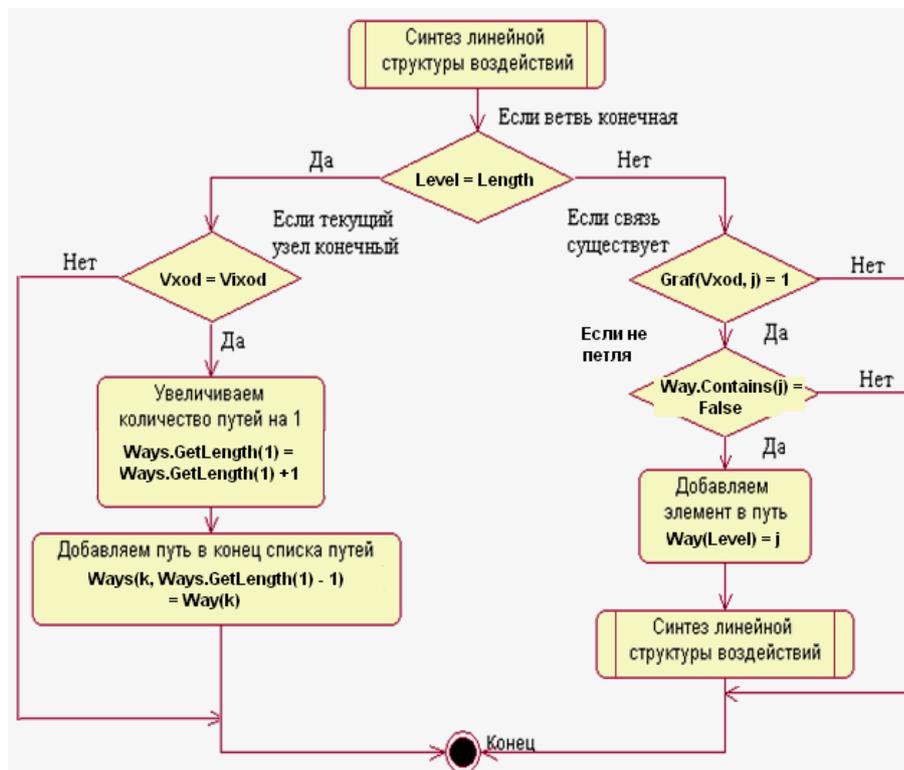


Рис. 2. Блок-схема алгоритма синтеза линейной структуры воздействий

Использование указанного подхода при автоматизации процесса формирования ФПД позволяет:

1) исследовать и модифицировать существующие модели описания и принципы совместности ФЭ, а так же методики синтеза структур ФПД;

2) разработать новые методики построения линейных структур ФПД;

3) разработать методики построения нелинейных (сетевых) структур ФПД.

В качестве направлений дальнейшего совершенствования процесса синтеза физического принципа действия можно назвать следующие:

1) расширение условий эквивалентности входов и выходов ФЭ за счет учета характера и области изменения физических величин; например за счет использования аппарата нечеткой математики;

2) модификация модели описания ФЭ для повышения полноты и адекватности отражения физических процессов, протекающих при функционировании ТС;

3) разработка и реализация методики построения сетевой структуры ФПД ТС;

4) проверка работоспособности автоматизированной системы ФПД на ряде тестовых и реальных технических задач.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Фоменков С. А., Давыдов Д. А., Камаев В. А.* Моделирование и автоматизированное использование структурированных физических знаний. Монография. – М.: Машиностроение – 1, 2004. – 278 с.

2. *Фоменков С. А., Петрухин А. В., Давыдов Д. А.* Автоматизированная система концептуального проектирования технических объектов и технологий // Информатика – машиностроение, 1999, № 3, С. 7–10.

3. *Фоменков С. А., Колесников С. Г.* Представление физических знаний в автоматизированном банке физических эффектов // Известия вузов. Машиностроение, 1998, № 1–3, С. 55–61.

4. *Фоменков С. А., Давыдов Д. А.* Формализованное представление физических знаний для компьютерной обработки информации // Техника машиностроения, 2001, № 6, С. 14–19.

5. *Камаев В. А., Фоменков С. А., Петрухин А. В., Давыдов Д. А.* Архитектура автоматизированной системы концептуального проектирования СОФИ // Программные продукты и системы, 1999, № 2, С. 30–34.

УДК 004.93'11

В. К. Гулаков, А. О. Трубаков

ПРОБЛЕМА БОЛЬШОГО ОБЪЕМА ВЕКТОРОВ ХАРАКТЕРИСТИК В ЗАДАЧЕ МНОГОМЕРНОГО ИНДЕКСИРОВАНИЯ ГРАФИЧЕСКОЙ ИНФОРМАЦИИ

Брянский государственный технический университет
gulakov@tu-bryansk.ru, trubakovao@mail.ru

В статье рассматриваются принципы индексирования многомерных данных для систем контекстного поиска изображений и, связанная с ней проблема большой размерности векторов характеристик. Подробно описаны проблемы перекрытия областей и арности дерева. Предложена структура, решающая данные проблемы за счет использования матриц сокращения.

Ключевые слова: контекстный поиск изображений, CBIR, структуры данных, многомерное индексирование.

V. K. Gulakov, A. O. Trubakov

PROBLEMS OF HIGH DIMENSIONAL INDEXES FOR CONTENT-BASED IMAGE RETRIEVAL

Bryansk State Technical University

In this paper we analyze the existing index structures for content-based image retrieval and problem of high dimensionality. It is in detail considered and described overlap between directory regions and storage utilization. We also propose a new structure, for indexing large amounts of point data in high dimensional space by help dimension reduction.

Key words: content-based image retrieval, CBIR, multidimensional data structures.

В связи с бурным ростом объемов графической информации остро встает вопрос автоматического анализа этой информации и организации поиска по ней. В данный момент исследователи отмечают эру развития web 2.0 технологий в сети internet. Одной из характеристик технологий web 2.0 является возросшая роль мультимедийного

контента в сети. Появляются большие объемы информации, которые никак не классифицируются, не оформляются и не снабжаются ключевыми словами. Это сильно усложняет работу классических поисковых систем. Часто единственным способом поиска по таким коллекциям является поиск по содержанию (*Content-Based Image*

Retrieval, CBIR) – поиск, основанный на характеристиках, полученных непосредственно из самого изображения, из значений тех пикселей, которые на нем присутствуют.

Многомерный подход к поиску графической информации

Одним из подходов решения задачи поиска изображений по содержанию является метод многомерного индексирования и поиск в пространстве характеристик. Для этого изображения сегментируют (выделяют на нем области со схожими цветовыми или текстурными характеристиками, которые необходимо классифицировать, распознать или найти на других изображениях). После выполнения этого шага изображение представляет собой множество $S = \{s_1, s_2, \dots, s_n\}$. Для каждой области s_i вычисляется набор характеристик. Данный набор можно представить в виде вектора числовых величин $s_i = \{k_{i,1}, k_{i,2}, \dots, k_{i,m}\}$, где $k_{i,j}$ – оценка i -го сегмента изображения по j -му критерию. В качестве критериев обычно выступают такие показатели, как гистограмма распределения цветов, текстурные показатели, характеристики аппроксимирующих фигур, коэффициенты дискретного преобразования Фурье (*DFT*) или вейвлет-преобразований.

Задача поиска похожих изображений в данной постановке сводится к сравнению сегментов эталонного изображения и изображений из коллекции и определение степени их схожести. Для сравнения двух сегментов на разных изображениях s_i и s'_i вводится специальная функция $Dist(s_i, s'_i)$ вычисляющая расстояние между двумя векторами. Чем больше сегментов одного изображения похоже на сегменты другого, тем большим сходством обладают данные изображения в целом.

Одной из особенностей индексирования векторов, полученных из графической информации, является их размерность. В практических задачах число компонент в векторе может достигать нескольких сотен. При реализации схем индексирования стараются сократить число компонент вектора, отбросив малозначимые или совпадающие критерии, но в универсальных поисковых системах выбрать претендентов на исключение часто бывает невозможно. К тому же, даже после сокращения числа компонент, размерность пространства остается слишком большой. Далее в статье рассматриваются основные проблемы, связанные с большой раз-

мерностью пространства, и предлагается новая структура, решающая их.

Проблема перекрытия регионов

Предлагаемая структура индексирования является древовидной с возможностью пересечения ограничивающих фигур внутренних узлов дерева. Разбиения пространства в узлах дерева происходит аналогично тем принципам, которые применяются в структурах, подобных *R*-деревьям [1, 4].

Из публикаций, посвященных *R*-деревьям, известно, что при индексировании часто могут появляться ситуации, при которых распределение дочерних узлов в группы без взаимного пересечения ограничивающих прямоугольников этих групп (*MBR*) будет просто невозможно. Взаимное пересечение *MBR* плохо влияет на производительность структур [4]. Если при поиске объектов в дереве регион поиска попадет в зону пересечения двух узлов, то для выполнения поиска появляется необходимость проверки обоих поддеревьев. В некоторых случаях такое поведение может привести к полной деградации алгоритмов обработки, и использование данной структуры станет нецелесообразным.

Положение осложняется тем, что рассматриваемое пространство имеет большую размерность. Как показано в работе [2], при увеличении числа измерений, пересечение внутренних узлов увеличивается нелинейно. Данная проблема широко рассматривается в литературе [6]. Основным выводом многих исследований является тот факт, что большинство структур плохо подходит для индексирования данных в пространствах большой размерности. С ростом числа измерений перекрытие внутренних регионов может приближаться к 100 % и использование структуры становится нецелесообразно [2]. Для получения приемлемой производительности нужно задачу сводить к пространствам с меньшим количеством измерений.

Проблема арности дерева

Еще одной проблемой, связанной с числом измерений, является размер одного элемента дерева (каждый элемент дерева должен содержать вектор чисел, равный размерности пространства). Объемы индексов, даже для небольших коллекций изображений, могут достигать огромных размеров. Поэтому практически всегда появляется необходимость использовать такие индексные структуры, которые эффек-

тивно работают во внешней памяти. Однако большая размерность пространства плохо влияет на арность дерева и эффективность структур в целом. Покажем, как размерность влияет на производительность.

Пусть необходимо построить две древовидные структуры для внешней памяти с размерностями $D_1 = d$ и $D_2 = k \cdot d$ соответственно. По аналогии с большинством современных структур, каждый узел дерева будет состоять из массива записей вида:

[*MBR_потомка, ссылка_на_потомка*].

Такая структура необходима для разбиения пространства на части и используется для организации эффективных процедур поиска. Обычно *MBR* задают в виде:

$$MBR = \{I_1, I_2, \dots, I_D\},$$

где D – число размерностей; I_i – интервал с закрытыми концами $[a, b]$, характеризующий размер объекта по соответствующей оси координат i . В этом случае объем памяти, необходимый для одной записи будет

$$V_{\text{записи}} = 2 \cdot D \cdot S_{el} + S_{ref},$$

где S_{el} – объем памяти, необходимый для хранения одной координаты измерения; S_{ref} – объем памяти, необходимый для хранения ссылки.

Одним из важных параметров древовидных структур является арность дерева – количество потомков, которые может иметь любой узел дерева. Арность дерева зависит от размера массива записей, структура которых была описана выше. При этом для структур внешней памяти арность рассчитывается как количество записей, которые могут поместиться в одном или нескольких блоках внешней памяти. В общем случае для вычисления арности получаем:

$$M = \left\lfloor \frac{S_{\text{page}}}{V_{\text{записи}}} \right\rfloor = \left\lfloor \frac{S_{\text{page}}}{2 \cdot D \cdot S_{el} + S_{ref}} \right\rfloor.$$

Если учесть, что указатель на потомка занимает достаточно мало памяти по сравнению с остальным вектором характеристик в многомерном пространстве, то для пространств с размерностями D_1 и D_2 получаем:

$$\frac{M_2}{M_1} = \frac{\left\lfloor \frac{S_{\text{page}}}{2 \cdot k \cdot d \cdot S_{el} + S_{ref}} \right\rfloor}{\left\lfloor \frac{S_{\text{page}}}{2 \cdot d \cdot S_{el} + S_{ref}} \right\rfloor} \approx \frac{1}{k}.$$

Т. е. увеличение размерности пространства влечет за собой пропорциональное уменьшение

арности дерева. При этом не трудно показать, что арность сильно влияет на высоту дерева. Дерево арности M в каждом узле делит пространство поиска на M частей. В идеальном случае его высота будет равна:

$$L = \log_M N.$$

Уменьшение арности приводит к изменению основания логарифма и, следовательно, к увеличению высоты дерева. Однако в структурах индексирования данных для внешней памяти высота дерева является показателем числа обращений к внешней памяти при выполнении поисковых запросов. Этот параметр считается одним из самых важных, сильнее всего влияющих на производительность структуры. Часто разработчики структур предпочитают пожертвовать любыми другими показателями для уменьшения объема операций с внешней памятью.

Структура предлагаемого индексного дерева

Как было показано выше, размерность пространства влияет на два самых важных параметра иерархических структур для внешней памяти – перекрытие внутренних узлов и количество обращений к внешней памяти. Поэтому проблема большой размерности становится одной из самых важных при индексировании векторов характеристик графической информации.

Решить проблему размерности можно с помощью сокращения пространства. Сокращение пространства позволяет уменьшить размер узла и улучшить перекрытие регионов. Однако трансформации осей и отсечение малозначащих измерений, часто применяемые для сокращения, не всегда пригодны. Очень часто они требуют априорного знания характера распределения и плохо подходят для случаев, в которых данные будут добавляться и изменяться динамически.

Интересным подходом является использование матриц сокращения. Измерения не отбрасываются полностью из рассмотрения, но при этом в каждом узле рассматривается только часть из них. Такой подход применяется в *TV*-дереве [3]. Однако использовать эту структуру для индексирования в данном случае не представляется возможным. При более глубоком анализе, можно увидеть, что новые критерии в дереве начинают рассматриваться только на тех уровнях, на которых все дочерние объекты имеют одинаковые значения первостепенных критериев. Этот

процесс похож на лексикографическую сортировку. Такое поведение не всегда приемлемо для систем поиска изображений по содержания. В подобных системах ни в одном запросе нельзя выстроить критерии так, что бы одни из них абсолютно доминировали над другими.

Предлагаемый в данной работе подход так же основан на сокращении пространства. Перед построением дерева формируется набор матриц сокращения $\{M_1, M_2, \dots, M_m\}$. Число матриц определяется разработчиком и не зависит от каких-либо других показателей структуры. Можно составить матрицы, которые выделяют отдельно группы критериев, связанных с гистограммой распределения цветов, статистическими показателями текстуры, критериями формы и положения объектов на изображении. Подобная группировка позволит применить разные формулы расстояний и мер подобия для разных групп в зависимости от потребностей.

Каждая матрица имеет высоту, равную размерности пространства и ширину, равную числу измерений, до которого она сокращает пространство. Каждая строка матрицы содержит 0 во всех положениях, кроме одного, соответствующего измерению, которое должно остаться при применении матрицы. Например, если рассматривается пространство из 5 измерений и необходимо построить правила, при которых на каких-то узлах дерева будут использоваться только первое и последнее измерение, а на других – три средних измерения, матрицы сокращения примут вид:

$$M_1 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}; \quad M_2 = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}.$$

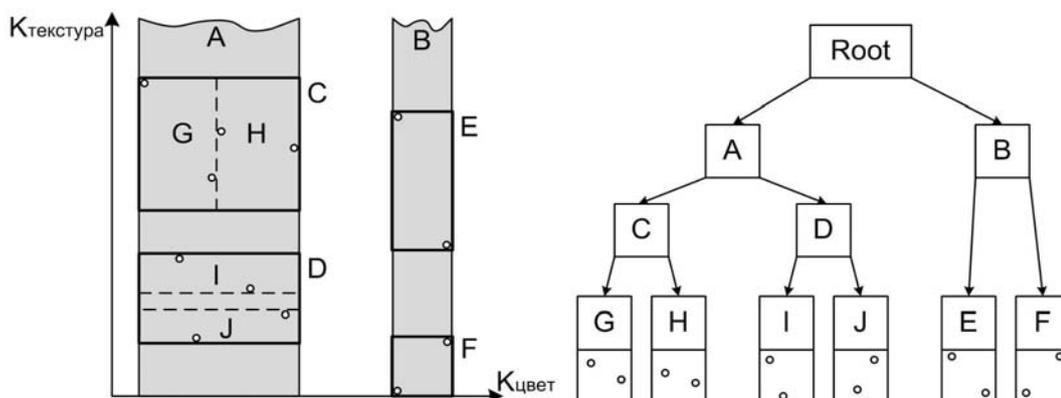


Рис. 1. Пример индексной структуры для 2-мерного случая

При построении дерева в каждом узле указывается номер матрицы сокращения, которая использовалась при его создании. Физический смысл матрицы сокращения заключается в определении набора компонент вектора характеристик, которые используются при построении узла и определении его *MBR*. Если имеется некоторый объект с вектором характеристик $X = \{x_1, x_2, \dots, x_n\}$ и узел дерева с матрицей сокращения M_i , то для выделения нужных компонент достаточно произвести перемножение вектора и заданной матрицы:

$$X' = X^T \cdot M_i.$$

В отличие от *TV*-дерева, разрабатываемый индекс не ограничивает порядок использования матриц и их применение в узлах дерева. Этот порядок может выбираться разработчиком самостоятельно в зависимости от его предпочтений, априорных знаний о системе, предполагаемых операциях поиска и распределении данных. Это дает некоторую свободу в использовании и проектировании системы.

На рис. 1 для наглядности показана структура дерева для двухмерного пространства (на нем в качестве ключей используется средняя яркость изображения и один из параметров текстуры). При этом выбраны матрицы сокращения до одного измерения:

$$X = \begin{bmatrix} \text{цвет} \\ \text{текстура} \end{bmatrix}; \quad M_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}; \quad M_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

Недостатки разработанного индекса. Использование произвольных матриц с произвольным порядком следования приводит к необходимости отказа от основных алгоритмов *TV*-дерева. В частности, деление узлов при переполнении и рост дерева производится не вверх (деление корня), а вниз (деление листьев).

Самым важным следствием такого построения является несбалансированность дерева. Однако произвольная группировка критериев и использование для индексирования пространства меньшей размерности превосходят негативный эффект от такого влияния. В работе [5] показано, что не во всех случаях балансировка является первостепенным фактором. Для некоторых применений несбалансированное дерево может оказаться более быстроедействующим и эффективным за счет применения алгоритмов кластеризации и более гибкой группировки объектов в узлах. К тому же разработанный индекс позволяет периодически проводить процедуры перебалансировки как по глубине, так и по распределению критериев, которые так же существенно улучшат структуру.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Гулаков, В. К. Многомерные структуры данных / В. К. Гулаков, А. О. Трубаков // Монография, Брянск, 2010. – 386 с.
2. Berchtold, S. The X-tree: An Index Structure for High-Dimensional / S. Berchtold, D. A. Keim, H.-P. Kriegel // Data. In Proc. of the 22nd International Conference on Very Large Data Bases (VLDB), Bombay, 1996. – P. 28–39.
3. Lin, K. I. The tv-tree: an index structure for highdimensional data / K. I. Lin, H. V. Jagadish, C. Faloutsos // The VLDB Journal, 3(4), 1994 – P. 517–542.
4. Manolopoulos, Y. R-Trees: Theory and Applications / Y. Manolopoulos, A. Nanopoulos, A.N. Papadopoulos, Y. Theodoridis // Springer-Verlag New York, 2005. – P. 194.
5. Ross, K. A. Cost-based Unbalanced R-Trees / K. A. Ross, I. Sitzmann, P. J. Stuckey // Proceedings of the 13th International Conference on Scientific and Statistical Database Management, 2001. – P. 203–212.
6. Wang, B. An Efficient Search Algorithm for High-Dimensional Indexing Using Cell Based MBR / B. Wang, B. Lee // Springer Berlin, Heidelberg, 2006 – P. 946–954.

УДК 002.6:004.91

Ю. А. Орлова

АНАЛИЗ МОДЕЛЕЙ И МЕТОДОВ ПОВЫШЕНИЯ ЭФФЕКТИВНОСТИ ПРОЕКТИРОВАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ*

Волгоградский государственный технический университет
yulia.orlova@gmail.com

В работе проведен анализ моделей и методов повышения эффективности проектирования программного обеспечения, на основе которого сформулирована необходимость создания автоматизированной системы семантического анализа текста технического задания. В процессе работы проводился анализ систем автоматизации проектирования программного обеспечения и систем, использующих семантический анализ текста.

Ключевые слова: модели и методы, повышение эффективности, программное обеспечение.

Y. A. Orlova

EXTENDED FUZZY ATTRIBUTE GRAMMAR ABOVE FRAME STRUCTURE OF TEXT OF A TECHNICAL SPECIFICATION

Volgograd State Technical University

In work the extended fuzzy attribute grammar above frame structure of text of a technical specification which allows to automate process of construction models of software based on technical specification is considered.

Key words: fuzzy attribute grammar, frame structure, text analysis, software models.

В настоящее время проектирование программного обеспечения представляет собой трудоемкий процесс, требующий от пользователя глубокого знания предметной области и навыков в проектировании.

Наиболее известные из коммерческих программных продуктов, используемых человеком для проектирования программного обеспечения, в основном предназначены для визуализации промежуточных и конечных резуль-

татов процесса проектирования. Некоторые из них позволяют полностью автоматизировать последние этапы проектирования: генерация кода, создание отчетной и сопровождающей документации и т. д. Начальный этап проектирования – анализ текста технического задания и построение модели программного обеспечения – выполняется аналитиком, т. е. задача автоматизации данного этапа остается открытой.

* Работа выполнена при поддержке грантов РФФИ 10-01-00135-а, 10-01-00165-а, 10-01-90012-Бел_а.

Современные CASE-средства охватывают обширную область поддержки многочисленных технологий проектирования ИС: от простых средств анализа и документирования до полномасштабных средств автоматизации, покрывающих весь жизненный цикл ПО.

В разряд CASE-средств попадают как относительно дешевые системы для персональных компьютеров с весьма ограниченными возможностями, так и дорогостоящие системы для неоднородных вычислительных платформ и операционных сред. Так, современный рынок программных средств насчитывает около 300 различных CASE-средств, наиболее мощные из которых так или иначе используются практически всеми ведущими западными фирмами.

Полный комплекс CASE-средств, обеспечивающий поддержку жизненного цикла ПО, содержит следующие компоненты: а) репозиторий, являющийся основой CASE-средства. Он должен обеспечивать хранение версий проекта и его отдельных компонентов, синхронизацию поступления информации от различных разработчиков при групповой разработке, контроль метаданных на полноту и непротиворечивость; б) графические средства анализа и проектирования, обеспечивающие создание и редактирование иерархически связанных диаграмм (поток данных, «сущность-связь» и др.), образующих модели ИС; в) средства разработки приложений, включая языки 4GL и генераторы кодов; г) средства конфигурационного управления; д) средства документирования; е) средства тестирования; ж) средства управления проектом; з) средства реинжиниринга.

Все современные CASE-средства могут быть классифицированы в основном по типам и категориям. Классификация по типам отражает функциональную ориентацию CASE-средств на те или иные процессы ЖЦ. Классификация по категориям определяет степень интегрированности по выполняемым функциям и включает отдельные локальные средства, решающие небольшие автономные задачи (tools), набор частично интегрированных средств, охватывающих большинство этапов жизненного цикла ИС (toolkit) и полностью интегрированные средства, поддерживающие весь ЖЦ ИС и связанные общим репозиторием. Помимо этого, CASE-средства можно классифицировать по следующим признакам: применяемым методологиям и моделям систем и БД; степени интегрированности с СУБД; доступным платформам.

На сегодняшний день Российский рынок программного обеспечения располагает следующими наиболее развитыми CASE-средствами: Vantage Team Builder (Westmount I-CASE); Designer/2000; Silverrun; ERwin+BPwin; S-Designor; CASE.Аналитик; Rational Rose.

Кроме того, на рынке постоянно появляются как новые для отечественных пользователей системы, так и новые версии и модификации перечисленных систем.

Проведенный анализ систем автоматизации проектирования показывает, что важное значение в процессе разработки ПО имеют средства спецификации проектов ПО. На начальных этапах проектирования ПО строится функциональная модель системы, которая описывает совокупность выполняемых системой функций. Результат представляется в виде диаграммы потоков данных и согласуется с заказчиком. Построение функциональных спецификаций осуществляется аналитиком на основе текста технического задания и является в настоящий момент не автоматизированным этапом, так как большую сложность вызывает автоматизация семантического анализа естественного языка.

Системы обработки связных текстов довольно разнообразны по структуре: интеллектуальные вопрос-ответные системы, системы общения с базами данных, диалоговые системы решения задач, системы обработки связных текстов и др. (Alex, AURA, GalaktikaZOOM, Link Grammar Parser, Гарант-Парк-Интернет, WordTabulator, Алхимик, Худломер, Рабочее Место Лингвиста, АБВУ Retrievall & Morphology (ARM), Следопыт, ВААЛ2000 и др.). Их общей чертой можно считать широкое использование технологий представления знаний. Функции систем такого рода заключаются в понимании текста и ответах на вопросы о его содержании. Понимание рассматривается не как универсальная категория, а как процесс извлечения информации из текста, определяемый конкретным коммуникативным намерением. Иными словами, текст «прочитывается» только с установкой на то, что именно потенциальный пользователь захочет узнать о нем. Тем самым и системы обработки связных текстов оказываются не универсальными, а проблемно-ориентированными.

Наиболее известные в настоящее время разработки в области анализа текста осуществляют семантический анализ на уровне предложения или слова.

По результатам анализа существующих систем, использующих семантический анализ текста (Galaktika-ZOOM, Link Grammar Parser, AURA, Алхимик, Рабочее Место Лингвиста, AuthorIT и др.), не было найдено системы, анализирующей текст технического задания.

Проведенный анализ существующих моделей естественно-языковой обработки текстов (трансформационные грамматики, генеративный лексикон, «Смысл↔Текст», HPSG, функциональные грамматики, системы семантических образцов, атрибутивные грамматики, SFG и др.) позволяет утверждать, что обработка текста сводится к методам синтаксического анализа, а проблема семантического анализа не решается.

Таким образом, на основе анализа систем автоматизации проектирования можно сказать, что автоматизируются конечные этапы проектирования программного обеспечения, такие как создание отчетной и сопровождающей документации, генерация кода и т. д. Начальный этап проектирования – анализ текста технического задания и построение модели программного обеспечения – выполняется аналитиком, т. е. задача автоматизации данного этапа остается открытой.

Для решения указанных проблем необходимо разработать автоматизированную систему семантического анализа текста технического задания. В результате разработки и внедрения предлагаемой системы повысится качество проектирования за счет автоматизации рутинного труда человека по извлечению полезной информации из стандартного документа и отображению ее в виде моделей программного обеспечения.

Проектирование программного обеспечения мало отличается от проектирования в других областях науки и техники, поэтому результаты данной работы можно расширить для применения в других областях человеческих знаний. Таким образом, открывающиеся перспективы повышают актуальность работы.

Данная система может использоваться как в высших учебных заведениях для обучения студентов, так и на всех организациях, составляющих техническое задание в качестве правового документа между заказчиком и исполнителем.

По результатам сравнительного анализа существующих систем, использующих семантический анализ текста, нами не было найдено системы, анализирующей текст техниче-

ского задания. Поэтому первостепенной задачей разрабатываемой системы является анализ текста с целью выявления функциональной структуры системы, описанной в техническом задании.

В целом можно сформулировать следующие общие требования, которые необходимо соблюдать для адекватной работы системы: текст технического задания создается человеком на естественном языке с соблюдением примерной структуры, требования к которой изложены в ГОСТах; на данном этапе развития методов анализа текста не представляется возможным анализировать естественный язык без каких-либо ограничений, поэтому техническое задание должно удовлетворять дополнительно сформулированным требованиям.

Общую архитектуру системы «СемантикаТЗ» можно представить схемой, показанной на рисунке. Автоматизированная система семантического анализа текста технического задания состоит из следующих подсистем: «Хранение документов», «Интерфейс», «Предварительная обработка текста», «Синтаксический анализ», «Семантический анализ», «Построение диаграмм потоков данных».

Подсистема «Хранение документов» должна обеспечивать возможность загрузки и сохранения текстовых документов, сохранения таблиц разделов, предложений и лексем, а также таблиц фреймов, в формате XML, сохранения отчетов о работе лексического и семантического анализатора.

Подсистема «Интерфейс» должна обеспечивать управление сеансами работы, визуальное представление XML-документов и отчетов, а также управление фоновыми операциями. Подсистема «Предварительная обработка текста» осуществляет загрузку структуры конечного автомата для разбора текста технического задания, построение таблиц разделов, предложений и лексем анализируемого технического задания.

Подсистема «Синтаксический анализ» осуществит поиск наилучшего синтаксического варианта, формирование списка синтаксических юнитов, синтаксических групп и синтаксических клауз.

Подсистема «Семантический анализ» загружает описание фреймовой структуры данных, осуществляет формирование автомата семантического разбора и построение семантического дерева технического задания.

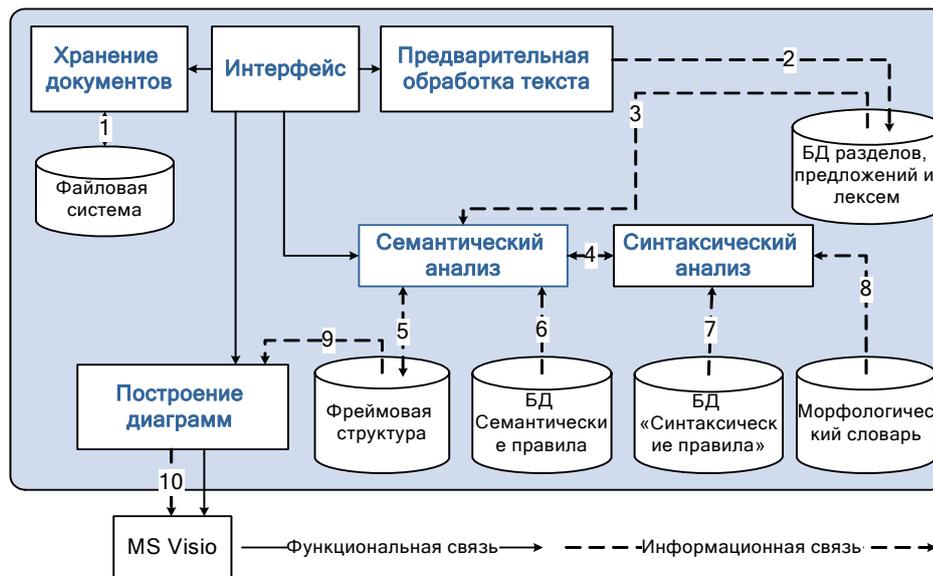


Рис. Архитектура системы «СемантикаТЗ»:

1 – файлы документов; 2, 3 – данные о разделах, предложениях и лексемах; 4 – синтаксические юниты, группы, клаузы; 5, 9 – фреймы; 6 – семантические правила; 7 – синтаксические правила; 8 – морфологические признаки; 10 – надписи и координаты фигур диаграмм

Подсистема «Построение диаграмм потоков данных» осуществляет построение и упорядочение графа потоков данных, а также создание фигур диаграмм потоков данных в Microsoft Office Visio.

Рассмотрим принцип функционирования автоматизированной системы.

Предварительная обработка текста необходима для того, чтобы разделить исходный текст технического задания на отдельные лексемы. Эта операция выполняется в три этапа: разделение на разделы, предложения и отдельные лексемы.

Уже после первого этапа производится работа не со всем текстом технического задания, а с его частями, представленными по разделам. По ходу работы текст технического задания дробится сначала на разделы все более низкого уровня, затем текст каждого раздела нижнего уровня делится на отдельные предложения, а каждое предложение делится на лексемы с указанием принадлежности к предложениям. Разделом называется фрагмент текста между двумя заголовками любого уровня, от первого до третьего (заголовки четвертого и дальнейших уровней по российским стандартам на текстовые документы не нумеруются).

Входной информацией подсистемы предварительной обработки текста является текст технического задания на ограниченном естественном языке, выходной информацией – таблицы разделов, предложений и лексем рассматри-

ваемого технического задания. Результаты могут быть представлены как в виде соответствующих таблиц, так и в виде дерева разделов. Это позволяет легко находить ошибки в оформлении документа, если он не соответствует ГОСТам.

Алгоритм предварительной обработки текста реализован следующим образом: сначала весь текст ТЗ делится на разделы первого уровня («1.», «2.», «3.» и т. д.), затем делится каждый раздел и так далее до третьего уровня. Предварительная обработка текста осуществляется с использованием аппарата конечных автоматов. В ходе работы конечного автомата символы, поступающие на его вход, накапливаются в буфере. В определенных состояниях конечного автомата осуществляется запись текущего содержимого буфера в одну из таблиц (разделов, предложений или лексем), после чего буфер опустошается. Работа автомата продолжается до достижения конечного состояния.

После этого полученные таблицы поступают на вход подсистемы синтаксического и семантического анализа. Семантический анализ текста производится на основе разработанной расширенной нечеткой атрибутивной грамматики текста технического задания. Если в правиле грамматики встретился терминал, имеющий синтаксический атрибут, то запускается механизм синтаксического анализа для текущего разбираемого предложения. Семантический анализатор предназначен для того,

чтобы на основе результата работы лексического анализа сформировать фреймовую структуру в виде базы данных. Выполняется в два этапа: построение грамматической таблицы на основе грамматики ТЗ и дерева лингвистических переменных $\beta_{k,i}$. Второй этап имеет два специальных блока: расчет степени принадлежности $\mu_{k,i}$ лингвистической переменной $\beta_{k,i}$ и создание фреймов.

После создания дерева лингвистических переменных начинается построение фреймового описания технического задания. Для этого используется информация о фреймах и названиях слотов, которая содержится в атрибутах символов грамматики.

Полученная фреймовая структура содержит значимую информацию о системе: сведения о входах и выходах системы, функциях и ограничениях. Для каждой функции также выделяются входы и выходы. Это позволяет на основе фреймовой структуры получить диаграммы потоков данных системы.

Для построения потоков данных сначала производится поиск входов функций, совпадающих с входами системы. Затем функции, на все входы которых поступают данные, помещаются на диаграмму на одном уровне. Их входы соединяются с входами системы. Далее производится поиск функций, входы которых совпадают с выходами полученных на предыдущем шаге функций. Они помещаются на следующем уровне, их входы соединяются с выходами функций предыдущих уровней и с входами системы. Работа алгоритма продолжается до тех пор, пока все функции не будут помещены на диаграмму. После этого производится соединение выходов функций с необходимыми выходами системы.

В работе проведен анализ моделей и методов повышения эффективности проектирования программного обеспечения.

Разработана концепция построения автоматизированной системы семантического анализа текста технического задания, в результате разработки и внедрения которой повысится качество проектирования за счет автоматизации рутинного труда человека по извлечению полезной информации из стандартного документа и отображению ее в виде моделей программного обеспечения.

Автоматизированная система позволяет обрабатывать текст технического задания на программу или программное изделие. В результате работы системы, требования к разрабатываемому программному обеспечению представляются в виде стандартизированных потоков данных, моделей, и т. д.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Заболеева-Зотова, А. В.* Автоматизация начальных этапов проектирования программного обеспечения / А. В. Заболеева-Зотова, Ю. А. Орлова // Изв. ВолГТУ. Серия «Актуальные проблемы управления, вычислительной техники и информатики в технических системах»: межвуз. сб. науч. ст. / ВолГТУ. – Волгоград, 2010. – Вып. 8, № 6(66). – С. 121–124.
2. *Орлова, Ю. А.* Алгоритмическое обеспечение анализа текста технического задания и построения моделей программного обеспечения / Ю. А. Орлова // Изв. ВолГТУ. Серия «Актуальные проблемы управления, вычислительной техники и информатики в технических системах»: межвуз. сб. науч. ст. / ВолГТУ. – Волгоград, 2010. – Вып. 8, № 6(66). – С. 68–72.
3. *Заболеева-Зотова, А. В.* Автоматизация семантического анализа документации технического задания / А. В. Заболеева-Зотова, Ю. А. Орлова // Вестник компьютерных и информационных технологий. – 2008. – № 9. – С. 26–34.
4. *Заболеева-Зотова, А. В.* Атрибутная грамматика формального документа «Техническое задание» / А. В. Заболеева-Зотова, Ю. А. Орлова // Изв. ВолГТУ. Серия «Актуальные проблемы управления, вычислительной техники и информатики в технических системах»: межвуз. сб. науч. ст. / ВолГТУ. – Волгоград, 2008. – Вып. 4, № 2. – С. 39–43.
5. *Заболеева-Зотова, А. В.* Автоматизация процедур семантического анализа текста технического задания / А. В. Заболеева-Зотова, Ю. А. Орлова // Известия ВолГТУ. Серия «Актуальные проблемы управления, вычислительной техники и информатики в технических системах»: межвуз. сб. науч. ст. / ВолГТУ. – Волгоград, 2007. – Вып. 3, № 9. – С. 49–52.
6. *Заболеева-Зотова, А. В.* Моделирование лексического анализа текста технического задания / А. В. Заболеева-Зотова, Ю. А. Орлова // Изв. ВолГТУ. Серия «Актуальные проблемы управления, вычислительной техники и информатики в технических системах»: межвуз. сб. науч. ст. / ВолГТУ. – Волгоград, 2007. – Вып. 2, № 2. – С. 39–42.
7. *Розалиев В. Л.* Предпосылки, возможности, перспективы создания автоматизированной системы распознавания эмоциональности речи / В. Л. Розалиев // Известия ВолГТУ. Серия «Актуальные проблемы управления, вычислительной техники и информатики в технических системах»: межвуз. сб. науч. ст. / ВолГТУ. – Волгоград, 2008. – Вып. 4, № 2. – С. 58–61.
8. *Розалиев В. Л.* Построение модели эмоций по речи человека / В. Л. Розалиев // Известия ВолГТУ. Серия «Актуальные проблемы управления, вычислительной техники и информатики в технических системах»: межвуз. сб. науч. ст. / ВолГТУ. – Волгоград, 2007. – Вып. 3, № 9. – С. 62–65.

ПОСТРОЕНИЕ СЕТЕЙ И ЗАЩИТА ИНФОРМАЦИИ

УДК 004.89

Е. Г. Ишкина, О. В. Щербинина

АРХИТЕКТУРА АДАПТИВНОГО СЕРВИСНО-ОРИЕНТИРОВАННОГО ПРОМЕЖУТОЧНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Астраханский государственный университет
ishkina@aspu.ru, oksana@aspu.ru

Представлена архитектура адаптивного сервисно-ориентированного промежуточного программного обеспечения, в основу которого положено многомерное представление сервиса (веб-сервиса SOAP/REST, мобильного сервиса NFC и др.). Выявлены следующие измерения сервисов: 1) профиль пользователя (неявные знания, полученные на основании анализа истории его взаимодействия с информационной системой, а также явные знания в форме предпочтений); 2) «внешний» физический контекст взаимодействия (место, время, используемое устройство и т. д.); 3) «внутренний» контекст взаимодействия (цель сеанса взаимодействия пользователя с информационной системой и возможные ограничения, накладываемые на результат, в совокупности составляющие задачу). Описан подход для адаптивной генерации новых сервисов (составных) «на лету», основанный на коллективном разуме сервисов, представленном в едином хранилище разнородных сервисов.

Ключевые слова: сервисно-ориентированный подход, адаптивные сервисы, автоматическая композиция сервисов, контекстно-зависимые системы, интеллектуальный анализ сервисов.

E. G. Ishkina, O. V. Scherbinina

SELF-ADAPTIVE MIDDLEWARE ARCHITECTURE

Астраханский государственный университет

This paper introduces a self-adaptive middleware architecture. The underlying model is based upon a multidimensional service representation (either web service (SOAP/REST) or mobile/NFC service); the service dimensions refer to the “4W”: 1) Who? i.e. user profile (implicit knowledge deduced from user interaction history and explicit knowledge in form of preferences); 2) Where and When? “external” (physical) interaction context (location, time, device, etc.); 3) What? “internal” interaction context (user ad-hoc task: goal(s) and optional requirements). An approach for dynamic adaptive service generation (on-the-fly composition) based on collective service intelligence stored in what we call “collective services warehouse” is outlined.

Key words: ubiquitous services, service-oriented middleware, adaptive services, automatic service composition, service mining, context-awareness.

В истории информатики постоянно появлялись новые функциональные слои, упрощающие разработку приложений (в порядке появления): операционные системы, системы управления базами данных, сервера приложений, мобильные сервера, EDGE-сервера. В условиях современных распределенных информационных систем, интегрирующих данные из многочисленных разнородных источников в сети Интернет, существует потребность в дополнительном функциональном слое поверх EDGE-сервера, в котором бы анализировалось взаимодействие пользователей с систе-

мой, система бы постоянно обучалась и могла в дальнейшем предоставлять пользователям более комплексные сервисы, ориентированные на их цели.

Это возможно реализовать на уровне промежуточного программного обеспечения (middleware, ППО), которое могло бы, во-первых, предоставить унифицированный доступ к разнородным сервисам (веб-сервисам: SOAP и REST, семантическим веб-сервисам: OWL-S, NFC-сервисам и т. д.), а во-вторых, реализовать поддержку единого хранилища сервисов и опыта их использования в целях обеспечения адап-

тивного предоставления сервисов конечным пользователям.

В его архитектуре требуется выделить как можно больше повторно используемых компонентов, которые могли бы быть в дальнейшем применены для различных целевых предметных областей, таких как цифровой город, электронное и мобильное правительство, обучающие лабораторные комплексы и т. д. Общим для этих областей является поддержка высокоуровневых действий пользователя [1], возможность реализации всей базовой функциональности в виде сервисов, автоматическая или полуавтоматическая композиция сервисов и предоставление конечному пользователю такого сервиса, который в большей степени соответствует ситуации использования системы.

Система не может хранить в готовом виде те сервисы, которые она будет предоставлять конечным потребителям, так как у каждого из них будет своя уникальная ситуация. Предлагается, по аналогии с ситуационной инженерией методов [2], воплотить и в рамках данного ППО ситуационный подход: система хранит фрагменты сервисов, а также знания об их использовании, а затем по ситуации генерирует композицию сервисов, наиболее уместную в конкретном случае использования.

Адаптивность в данном случае представляет собой реализацию свойства контекстной или ситуационной зависимости систем (context/situation awareness). Контекст – любая информация, которая может быть использована для того, чтобы охарактеризовать ситуацию или сущность. Сущность – человек, место или объект, который рассматривается как релевантный к взаимодействию пользователя и приложения [3]. Система является контекстно-зависимой, если она может извлекать, интерпретировать и использовать контекстную информацию и адаптировать свою функциональность к текущему контексту использования [4]. Ситуационная зависимость – наиболее целостный вариант контекстной зависимости, при котором ситуации рассматриваются как логически агрегированные элементы контекста [5].

В рамках адаптивного сервисно-ориентированного ППО ситуация рассматривается в пространстве трех измерений: 1) профиль пользователя (неявные знания, выведенные на основании анализа истории взаимодействия пользователя с системой, и явные знания в форме

предпочтений); 2) «внешний» контекст (местоположение, время и т. д.); 3) «внутренний» контекст (решаемая задача). Для первых двух измерений набор и важность характеристик зависят от целевой предметной области. Проблема представления соответствующих знаний может быть решена с применением онтологического подхода.

Составные сервисы могут быть добавлены в общедоступный репозиторий либо самим пользователем, имеющим соответствующие права, либо автоматически путем применения алгоритмов интеллектуального анализа данных (data mining) для поиска новых сервисов (service mining) на основании анализа истории работы пользователей с системой.

Интеллектуальный поиск полезных новых композиций веб-сервисов представляет собой актуальную область исследований. Полученные результаты представлены, например, в [6]. Интеллектуальный поиск сервисов может быть основан на использовании эффективных алгоритмов анализа процессов, которые позволяют получить модели процессов на основе анализа совокупности логов событий системы [7].

На рис. 1 представлена архитектура адаптивного сервисно-ориентированного ППО. Его основными задачами, реализованными в виде отдельных слоев являются: 1) интеграция гетерогенных сервисов; 2) их хранение в едином хранилище и постоянный поиск контекстных зависимостей и новых, составных сервисов, представляющих собой полезные сообществу пользователей стратегии решения задач в определенных ситуациях; 3) генерация уникального сервиса, наилучшим образом соответствующего ситуации, на основании сервисов, представленных в хранилище и знаний об их использовании. Отдельный слой служит для генерации представления для сервиса, предоставляемого конечному потребителю (в соответствии с используемым им устройством, пользовательскими предпочтениями и т. д.).

Слой базовых сервисов обеспечивает интеграцию гетерогенных сервисов различных поставщиков благодаря единой метамодели сервисов. Только на этом уровне известны детали вызовов сервисов. Для выполнения заданной функциональности в настоящее время разрабатывается онтология, позволяющая аннотировать сервисы (параметры вызовов сервисов, параметры входной и выходной информации и т. д.).

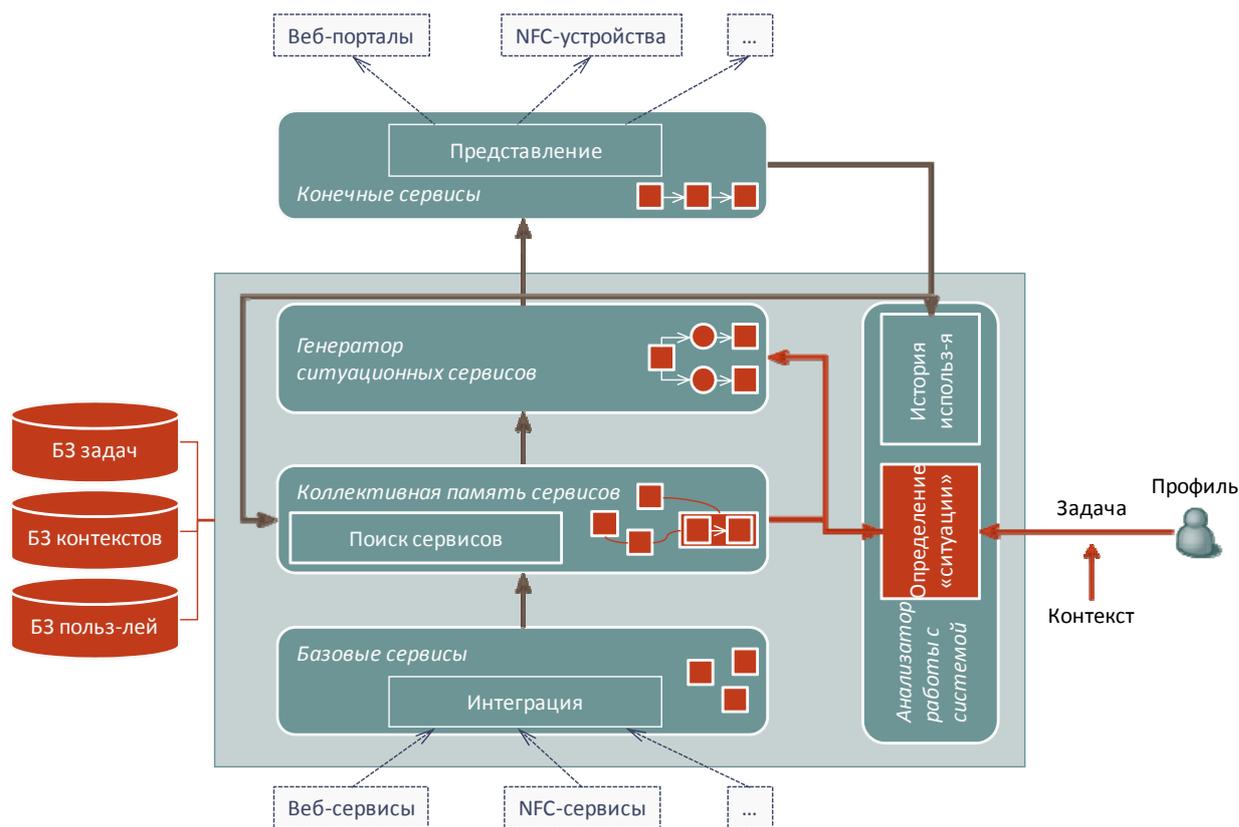


Рис. 1. Архитектура адаптивного сервисно-ориентированного ППО

Коллективная память сервисов необходима для поиска и хранения составных сервисов – паттернов, применимых в определенных ситуациях, выявленных автоматически или добавленных пользователем. На этом уровне определяется второй уровень метаданных сервисов – выявленные в результате анализа истории их использования контекстные факторы, влияющие на их применимость в той или иной ситуации. Аннотируются как интегрированные базовые сервисы, так и составные.

Базовый компонент коллективной памяти – сервис, представляющий собой *фрагмент* использования системы. В общем случае, решением задачи пользователя является составной сервис, который формируется генератором ситуационных сервисов динамически из фрагментов на основании накопленных знаний.

Все базовые сервисы вместе с их формальными описаниями, полученными на этапе интеграции в предыдущем слое сохраняются в коллективной памяти. Затем коллективная память расширяется благодаря найденным автоматически или добавленным пользователями составным сервисам. При этом производится оценка того, насколько полезным является добавление

составного сервиса в коллективную память, то есть насколько велика вероятность его повторного использования. В соответствии с метаданными сервисов по результатам логического вывода определяются прогнозируемые функциональные и нефункциональные характеристики композиций сервисов.

В ходе функционирования распределенной информационной системы, включающей адаптивное сервисно-ориентированное ППО, с помощью алгоритмов интеллектуального анализа данных осуществляется поиск ключевых ситуационных характеристик, влияющих на совместное использование сервисов в композиции. Выявленные таким образом характеристики представляют собой предусловия, используемые при адаптивной генерации сервисов для конечных пользователей.

Каждое из трех измерений ситуации (задача, профиль пользователя, физический контекст) представляется с помощью онтологии, которая позволяет изменять набор и важность ситуационных характеристик, соответствующих целевой предметной области. Сервис является контекстно-независимым компонентом, а его многоуровневые аннотации элементами онтологий, соот-

ветствующих трем измерениям ситуации, позволяют оценивать релевантность сервиса в конкретной ситуации.

На основании данных, полученных в результате определения ситуации, и метамоделей сервисов, хранимых в коллективной памяти, генерируется уникальный сервис, соответствующий заданной ситуации, представляющий собой композицию (фрагментов) сервисов из коллективной памяти. Основным критерий для выбора сервисов – пользовательский сервисно-ориентированный запрос для решения конкретной задачи. Дополнительные критерии – выявленные значимые параметры профиля пользователя и физического контекста. В совокупности эти критерии составляют контекст первого уровня. Затем, на основании знаний о взаимосвязи элементов контекста первого уровня, формируется контекст второго уровня – ситуация.

Далее производится поиск подходящего набора сервисов в соответствии с семантическими характеристиками ситуаций. Если ни одного подходящего сервиса не найдено, производится декомпозиция задачи на подзадачи и определение набора сервисов для каждой из них. Отобранные кандидаты проверяются на синтаксическую (возможность последовательного вызова сервисов, использующих различные протоколы) и семантическую (в пределах ситуации) совместимость.

Далее генерируется композиция сервисов. Элементы композиции в дальнейшем вызываются автоматически через уровень базовых сервисов, обмен данными между элементарными сервисами осуществляется прозрачно для пользователя. Результат композиции представляет собой уникальный сервис, наиболее соответствующий заданной ситуации при условии наличия заданного набора базовых сервисов.

На уровне конечных сервисов решается проблема корректного вызова базовых сервисов как элементов сгенерированной на предыдущем этапе композиции сервисов. Кроме того, решается также задача адаптации интерфейса работы с сервисом в соответствии с используемым устройством и предпочтениями пользователя.

Представлена архитектура адаптивного сервисно-ориентированного промежуточного программного обеспечения и описано назначение и требования к его основным функциональным блокам. Основная идея, реализованная в данной архитектуре, – в современных разнородных распределенных системах необходимо накопление знаний об их использовании. Эти знания не могут более быть представлены только фактами, они должны представлять собой стратегии решения задач (сервисы) в определенных ситуациях (контексте). Современные системы носят задачеориентированный характер, их работа направлена на прозрачное решение комплексных задач пользователей. В дальнейшем сервисы могут быть рекомбинированы с получением новых, составных сервисов. Важно является предугадывание результата композиции сервисов, его применимости в определенной ситуации, схожей с той, в которой он был использован изначально. Это возможно осуществить с применением онтологического подхода.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Sousa, J. P., Poladian, V., Garlan, D., Schmerl, B., Shaw, M. Task-based Adaptation for Ubiquitous Computing. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, Special Issue on Engineering Autonomic Systems, 36(3), 2006, p.328–340.
2. Ralyté, J., Deneckère, R., Rolland, C. Towards a generic model or situational method engineering // Proceedings of the 15th Conference on Advanced Information System Engineering, CAISE'03, Velden, Austria, 2003, p.16–20.
3. Dey, A. K., Abowd, G. D. Towards a better understanding of context and context-awareness // Proceedings of the Workshop on the What, Who, Where, When and How of Context-Awareness, ACM Press, New York, 2000
4. Byun, H. E., Cheverst, K. Utilizing context history to provide dynamic adaptations // Journal of Applied Artificial Intelligence, 2004, 18(6), p.533–548.
5. Anagnostopoulos, C. B., Ntarladimas, Y., Hadjiefthymiades, S. Situational computing: An innovative architecture with imprecise reasoning // Journal of Systems and Software, 2007m 80(12):1993–2014.
6. Zheng, G. Web Service Mining // Thèse de doctorat, Virginia Tech, 2009.
7. Rouached, M., Gaaloul, W., van der Aalst, W. M. P., Bhiri, S., Godart, C. Web Service Mining and Verification of Properties: An approach based on Event Calculus // OTM Confederated International Conferences, 2006.

УДК 004.05

*С. И. Кирносенко, В. С. Лукьянов***ИДЕНТИФИКАЦИЯ ИСПРАВЛЯЮЩИХ РЕВИЗИЙ В СИСТЕМАХ КОНТРОЛЯ ВЕРСИЙ****Волгоградский государственный технический университет**
semyon.kirnosenko@yandex.ru

Рассмотрены три алгоритма идентификации исправляющих ревизий в системах контроля версий, находящие применение в практике. На основе данных, полученных для трех проектов с открытым исходным кодом, проведен анализ эффективности этих алгоритмов. Даны рекомендации по их применению при решении различных задач.

Ключевые слова: качество, системы контроля версий, метрики, ошибки, исходный код.

*S. I. Kirnosenko, V. S. Lukyanov***BUG FIX REVISION IDENTIFICATION IN VERSION CONTROL SYSTEMS****Volgograd State Technical University**

We consider three algorithms of bug fix revision identification in version control systems. Experimental data from three open-source projects are analyzed to evaluate efficiency of these algorithms. We make recommendations how to use them in different situations.

Key words: quality, version control systems, metrics, bugs, source code.

При разработке ПО для сохранения промежуточных состояний исходного кода, организации коллективной работы и прочих задач используются системы контроля версий. Для кода проекта обычно создается отдельное хранилище (репозиторий). Разработчики последовательно добавляют в него код. Каждое вносимое в репозиторий изменение (ревизия), как правило, связано с одной из следующих причин: добавление новой функции, исправление ошибки или рефакторинг. Системы контроля версий хранят большие объемы информации о том, когда и какие изменения в коде были сделаны. Анализ этой информации может дать понимание тех процессов, которые происходят при разработке. В частности, анализ метрик можно использовать для оценки качества ПО. Наиболее распространенной метрикой качества ПО является плотность ошибок. Она измеряется как количество ошибок на тысячу строк кода. Под ошибкой понимается наличие в программе логически-неправильного кода, который приводит к некорректной работе программы при определенных условиях и требует исправления.

При подсчете плотности ошибок, необходимым этапом обработки информации является обнаружение ошибочного кода. Ошибочным кодом, как правило, считается код удаленный в исправляющей ревизии. Исправляющая ревизия – это ревизия системы контроля версий, в которой внесены только те изменения, которые необходимы для исправления отдельно взятой ошибки. Внесение исправлений в виде отдельных ревизий – распространенная техника. Такой подход удобен для осуществления основных

задач управления процессом разработки и широко применяется как в крупных проектах с открытым исходным кодом, так и в коммерческих проектах. Одна из задач автоматизированных средств подсчета метрик состоит в идентификации исправляющих ревизий, т. е. в классификации ревизий на исправляющие и все остальные. От правильности такой идентификации будет зависеть точность моделей предсказания плотности ошибок. Множество таких моделей нуждается в информации об исправляющих ревизиях. Эти модели, в свою очередь, используются для различных задач управления проектами: планирование выпуска новых версий, распределение ресурсов тестирования, оценка рисков крупных изменений и т. п.

Наиболее распространенным способом идентификации исправляющих ревизий является алгоритм на основе анализа лог-сообщения ревизии [1]. Каждый раз, когда разработчик вносит изменения в репозиторий системы контроля версий, он может добавить комментарий в виде простого текста. Это полезная возможность, т. к. в дальнейшем любой другой разработчик, работающий с репозиторием, может по таким комментариям определить что за изменения были сделаны, без необходимости просматривать код. Если вносятся изменения связанные с исправлением ошибки, то можно ожидать, что в комментариях к ним будут обнаружены ключевые слова: «исправление», «ошибка» и т. п. Обнаружив в тексте комментария такие слова, можно с высокой долей вероятности утверждать, что ревизия является исправляющей. В дальнейшем будем условно обозначать этот алгоритм как F1.

Другой метод основан на поиске ошибок, информация о которых содержится в системе учета ошибок. Такие системы часто используются в крупных проектах. Основная их задача – учет обнаружения и исправления ошибок. Каждый раз при обнаружении новой ошибки, в систему учета ошибок добавляется новая запись, соответствующая обнаруженной ошибке. Этой записи присваивается уникальный идентификатор, что позволяет на нее в дальнейшем ссылаться. Первоначально ошибке присваивается неопределенный статус. Далее производится попытка воспроизвести ее и если это удастся, то ошибка передается на исправление. После исправления выполняется контроль того, что ошибка исправлена и если это так, то ошибка помечается как исправленная. В дальнейшем можно получить данные обо всех исправленных ошибках и выполнив поиск по лог-сообщениям системы контроля версий, найти идентификаторы исправленных ошибок. Этот метод можно считать обратным методу, часто используемому вместе с предыдущим, когда выполняется поиск идентификаторов системы учета ошибок по регулярному выражению [1]. В данной работе он рассматривается отдельно по ряду причин, в частности, потому что применение этого метода возможно только при выполнении дополнительных требований (использование системы учета ошибок). В дальнейшем будем условно обозначать этот алгоритм как F2.

Еще один способ наиболее спорен, но также находит применение. Он основан на предположении, что при исправлении ошибки изменения затрагивают минимальное число файлов проекта 1–2, тогда как прочие изменения затрагивают большее число файлов. На основе этого предположения и работает данный алгоритм идентификации, который классифицирует ревизию как исправляющую, если она модифицирует не более двух файлов [2]. В дальнейшем будем условно обозначать этот алгоритм как F3.

Все три алгоритма описанные выше находят свое применение в практике. Однако выбор конкретного алгоритма, в том или ином исследовании, обычно происходит исходя из удобства применения в конкретной ситуации. При этом совершенно не затрагивается вопрос о точности этих алгоритмов. Часто имеют место предположения о том, что точность эта удовлетворительна для задач исследования. Задача данной работы – установить наиболее достоверный способ идентификации исправляющих

ревизий. Кроме того, рассматриваются две подзадачи: идентификация исправляющих ревизий для задач ручной и автоматической обработки. В первом случае, идентификация выполняется с целью получения списка предположительно исправляющих ревизий для их дальнейшего анализа экспертом. При этом ложные срабатывания гораздо менее критичны, чем пропуск действительных исправлений. Во втором случае, поиск выполняется с целью получения списка предположительно исправляющих ревизий для их последующей автоматической обработки в задачах подсчета метрик. При этом, в зависимости от обстоятельств, эксперт принимает решение о необходимости идентификации максимального числа исправляющих ревизий и допустимой доле ложных срабатываний.

Для решения поставленной задачи был проведен эксперимент. Выше описанные алгоритмы были применены для идентификации ревизий из репозитория трех проектов с открытым исходным кодом: `gnometerminal`, `gnome-vfs` и `dia`. Репозитории этих проектов открыты для свободного доступа в интернете (<http://svn.gnome.org/>). Для ускорения процесса анализа данных, все три репозитория были скопированы на локальный компьютер с помощью утилиты `rsvndump` (<http://rsvndump.sourceforge.net/>). Также для всех трех проектов был выполнен экспорт данных из системы учета ошибок `bugzilla` (<https://bugzilla.gnome.org/>). Экспортировались только ошибки отмеченные статусом «FIXED», т. е. исправленные [3]. Далее была произведена выборка ошибок исправленных в период, охватываемый историей, содержащейся с репозиториях кода. С помощью специально разработанного ПО из полученных данных была сформирована база данных, на основе ранее описанной схемы [4]. Далее из базы данных были произведены выборки для получения результатов.

Основные полученные результаты представлены в таблице. Для каждого из трех исследованных репозиториях приведено общее число ревизий. Для каждого алгоритма идентификации приводятся: размерность множества полученных ревизий, размер исследуемой выборки для ручной проверки (5 % от размера множества), количество ложных срабатываний на выборке (FP_B) и статистическая оценка общего числа ложных срабатываний на всем множестве (FP^*). Далее приводятся размерности трех различных множеств образованных из множеств полученных с помощью алгоритмов F1 и F2.

Метрики исследованных проектов

репозиторий		gnome-terminal	gmome-vfs	dia
ревизий		3436	5550	4375
F1	размер	598	1231	993
	выборка	29	61	49
	FP_B	1	8	4
	FP^*	20 (3%)	161 (13%)	81 (8%)
F2	размер	397	408	528
	выборка	19	20	26
	FP_B	0	0	0
	FP^*	0 (0%)	0 (0%)	0 (0%)
F3	размер	2658	3523	2457
	выборка	132	176	122
	FP_B	116	133	101
	FP^*	2335 (87%)	2662 (75%)	2034 (82%)
F1 - F2		278	930	609
F2 - F1		44	79	59
F1 U F2		642	1310	1052

Первое, на что следует обратить внимание – крайне высокое число ложных срабатываний для алгоритма F3. Для трех исследуемых проектов от 75 до 87 % ревизий, идентифицированных этим алгоритмом как исправляющие, на самом деле таковыми не окажутся. Такой результат можно объяснить тем, что в исследуемых проектах изменения, как правило, вносятся очень небольшими порциями, что вообще характерно для большинства крупных проектов с открытым исходным кодом. Для таких проектов этот алгоритм оказывается непригодным. Возможно, изначально этот алгоритм был предложен для использования в специфичном проекте, в котором модификации были крупными.

Алгоритм F2 показал отличные результаты. На выборках для всех трех проектов не было обнаружено ни одного ложного срабатывания. Этот алгоритм можно считать надежным способом идентификации исправляющих ревизий с низкой вероятностью ложных срабатываний. Его можно использовать для поиска исправляющих ревизий как для задач ручной обработки данных, так и для задач автоматического анализа. Главное преимущество алгоритма – низкая вероятность ложных срабатываний, возможность получения дополнительной информации об ошибке, такой как ее серьезность и прочее. Недостатком можно считать то, что алгоритм применим только если используется система учета ошибок.

Алгоритм F1 позволяет обнаружить значительно большее число исправляющих ревизий чем F2, но появляется вероятность ложных срабатываний, доля которых для исследованных проектов составила от 3 до 13 %. Следует отметить, что большая часть ложных срабатываний была следствием упоминания ключевых слов в ревизиях, вносящих исправления в документацию и прочие дополнительные файлы проекта, не относящиеся к коду. Посредством осуществления дополнительной фильтрации, анализирующей модифицированные файлы, например, по расширению и выбирающей только файлы кода, можно снизить число ложных срабатываний примерно вдвое. Явным плюсом алгоритма является его простота. Основной минус заключается в вероятности ложных срабатываний, ввиду несоответствия текста комментария или его отсутствия.

На основе полученных экспериментально результатов, можно сделать следующие выводы о применимости рассмотренных алгоритмов идентификации исправляющих ревизий. Алгоритм F2 можно применять для любых целей. Кроме того, следует помнить, что данный алгоритм не позволяет обнаружить значительное число исправляющих ревизий, внесенных без записей в систему учета ошибок. Алгоритм F1 можно применять для задач ручной обработки исправляющих ревизий. Также этот алгоритм применим для задач автоматической обработки,

но с оговоркой о том, что с увеличением числа идентифицированных исправляющих ревизий также увеличивается и число ложных срабатываний. В каждом конкретном случае эксперт должен решать, что важнее в конкретной задаче: идентифицировать максимальное число исправляющих ревизий или избежать ложных срабатываний. Алгоритм F3 неприменим из-за слишком высокого процента ложных срабатываний.

Для задач ручной обработки оптимальной стратегией идентификации исправляющих ревизий можно назвать метод, основанный на выборе посредством комбинации алгоритмов F1 и F2. Для комбинации алгоритмов результирующее множество ревизий будет получено как объединение множеств, полученных каждым из алгоритмов, входящих в комбинацию. Такой метод позволяет идентифицировать максимальное число исправляющих ревизий. Для задач автоматического анализа оптимальным является использование алгоритма F2. Использование комбинации алгоритмов F2 и F1, как видно из таблицы, позволяет увеличить размер множества идентифицированных исправляющих ревизий в среднем в полтора раза. Но вместе с тем, доля ложных срабатываний в результирующем множестве составит в среднем 8 %. Целесообраз-

ность такого подхода должна определяться экспертом для каждой отдельной задачи.

На основании результатов проведенного эксперимента можно утверждать, что наиболее достоверным из описанных выше алгоритмов идентификации исправляющих ревизий, является F2. Алгоритм F1 вносит элемент недостоверности в виде ложных срабатываний. Алгоритм F3 можно считать недостоверным.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Slivewski, J.* When do changes induce fixes? / J. Slivewski, T. Zimmermann, A. Zeller // MSR '05: Proceedings of the 2005 international workshop on Mining software repositories. – New York, NY, USA: ACM, 2005. – Pp. 1–5.
2. *Ostrand, T. J.* Where the bugs are / T. J. Ostrand, E. J. Weyuker, R. M. Bell // Proc. ACM SIGSOFT Int'l Symp. Software Testing and Analysis (ISSTA). – New York, NY, USA: ACM Press, 2004. – Pp. 86–96.
3. Fault detection and prediction in an open-source software project / M. English, C. Exton, I. Rigon, B. Cleary // PROMISE '09: Proceedings of the 5th International Conference on Predictor Models in Software Engineering. – New York, NY, USA: ACM, 2009. – Pp. 1–11.
4. *Кириносенко, С.* Извлечение данных для подсчета плотности ошибок кода / С. Кириносенко // Приоритетные направления современной российской науки глазами молодых ученых: Всероссийская научно-практическая конференция молодых ученых и специалистов. – Ряз. гос. ун-т им. С. А. Есенина, 2009. – С. 159–161.

УДК 621.31

В. С. Лукьянов, О. В. Степанченко, Е. С. Кузнецова

ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ КАНАЛОВ ПЕРЕДАЧИ ДАННЫХ В ПРОЦЕССЕ ИЗУЧЕНИЯ СТУДЕНТАМИ ВОПРОСОВ ПО НАДЕЖНОСТИ ИНФОРМАЦИОННЫХ СИСТЕМ

Камышинский технологический институт (филиал)
Волгоградского государственного технического университета
ecmsys@vstu.ru, ovste@mail.ru, kuznkat@yandex.ru

Описаны возможности применения методов моделирования при исследовании надежности трактов передачи данных в автоматизированных системах управления (АСУ) с точки зрения надежности технического оборудования. А также рассмотрены возможности использования полученных моделей в процессе подготовки специалистов в области проектирования информационных систем.

Ключевые слова: алгоритм функционирования, интенсивность отказа, интенсивность восстановления, имитационное моделирование, канал передачи данных, коэффициент готовности, коэффициент простоя, надежность системы.

V. S. Lukyanov, O. V. Stepanchenko, E. S. Kuznetsova

DATA TRANSMISSION CHANNELS SIMULATION IN THE PROCESS OF THE INFORMATION SYSTEMS RELIABILITY STUDY

Камышинский технологический институт (филиал)
Волгоградского государственного технического университета

The article presents the possibilities of the simulation technique application in the study of the transmission channels reliability in the computer aided control systems. The description is carried out from the technical equipment reliability point of view. The possibilities of the obtained models application in the process of the information systems design study were analyzed as well.

Key words: operation algorithm, failure intensity, intensity recovery, simulation modeling, data channel, coefficient Preparedness or Availability, coefficient of loss, system reliability.

Рост значения проблемы надежности связан с некоторыми особенностями развития совре-

менной техники. Во-первых, существует стремление к подробному планированию хода произ-

водственных процессов, которые становятся все более сложными. Во-вторых, все больше распространяется автоматизация различных процессов. В-третьих, автоматизированные системы выполняют все более ответственные задачи.

В настоящее время подготовка специалистов в этой области сводится к обучению основным методам проектирования АСУТП. Вопросы надежности рассматриваются, основываясь на опыте и интуиции преподавателя. Студенты, изучившие материал данного направления, прослушавшие лекционную и практическую части курсов, имеют очень смутное представление о практическом влиянии системной надежности на работу проектируемых ими систем.

В дисциплинах общепрофессионального блока дисциплин, изучаемых студентами направления «Информатика и вычислительная техника», в том числе: «Организация ЭВМ и систем», «Операционные системы», «Сети ЭВМ и телекоммуникации» и др., раздел надежности полностью отсутствует. Этому вопросу также мало уделяется внимания в публикациях, рассматривающей вопросы построения АСОИУ [1–3].

В процессе овладения знаниями по учебным дисциплинам, содержащим вопросы оценки качества функционирования технических систем и устройств, а также автоматизированных систем обработки информации и управления (АСОИУ), студенты сталкиваются с необходимостью решения практических задач надежности и эффективности.

В связи с этим изучение параметров оценки АСУ с учетом надежности в учебном процессе приведет к более качественной подготовке специалистов в данной области [1–3].

Современное проектирование сложных автоматизированных технических систем в своей основе широко использует базовый принцип конструирования, в основе которого заложено использование в новых разработках типовых и стандартных материалов, деталей, коммуникационных изделий, модулей, узлов и блоков. Такой подход в значительной степени позволяет проектировать изделия с максимальной надежностью, достижимой при заданных ограничениях на стоимость.

Отсюда актуализируется задача выбора альтернативных проектных вариантов технической системы и выбора вариантов компонентов, наилучшим образом отвечающих техническому заданию (ТЗ). Сегодня имеется большой выбор

комплектующих, поставляемых различными фирмами. При этом перед разработчиком возникает задача выбора лучших вариантов среди огромного количества близких по параметрам изделий, с целью проектирования новых конкурентно-способных разработок.

Перечисленные проблемы могут быть решены благодаря созданию и внедрению в учебный процесс имитационных моделей, реализованных с помощью компьютерных технологий, позволяющих студентам наблюдать за влиянием некоторых характеристик надежности на эффективность функционирования системы в целом.

В данной статье рассматривается функционирование имитационных моделей трактов передачи данных с происходящими в них кратковременными сбоями, не превышающими допустимое время простоя [4]. Такие случаи встречаются в различных технологических процессах, при передаче информации на большие расстояния (теле- и радиоканалы, системы централизованного управления производственными цехами, диспетчерского управления работой городского автотранспорта и т. п.).

Для реализации имитационных моделей были разработаны алгоритмы функционирования каналов передачи данных с кратным и дробным резервированием.

Для создания моделей на основании разработанных алгоритмов использовался пакет программ Trace Mode.

В результате реализации алгоритмов были получены несколько моделей. Структура тракта передачи данных с однократным резервированием представлена на рис. 1.

При выполнении программы задаются параметры канала, на мониторе можно наблюдать процессы перехода системы из одного состояния в другое. Таким образом, пользователь может наглядно наблюдать как начальные значения (интенсивность отказа, время восстановления) влияют на работоспособность системы в целом (на значение коэффициента готовности). Результаты моделирования заносятся в базу данных и экспортируются в офисный пакет MS Access. Предусмотрена возможность вывести результаты в форме диаграмм периодов работы системы в каждом из промежуточных состояний. Для каналов передачи данных были также построены математические модели с использованием пакета программ для инженерных и математических расчетов Mathcad 14.

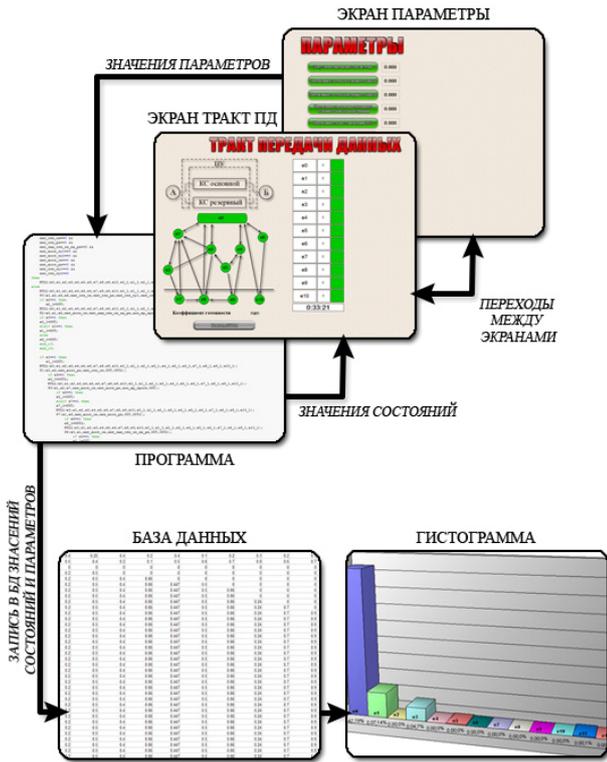


Рис. 1. Структура модели канала передачи данных

Результатом моделирования являются графики, показывающие зависимость выходных параметров от входных.

График изображенный на рис. 2 показывает зависимость коэффициентов готовности $K_{\text{гот}}$ и простоя $K_{\text{прост}}$ от интенсивности восстановления μ_0 , т. е. зависимость вероятности того, что восстанавливаемый объект окажется в работоспособном состоянии или находится в простое в произвольно выбранный момент времени в установленном процессе эксплуатации от условной плотности вероятности восстановления работоспособности объекта, определенной для рассматриваемого момента времени [5]. При увеличении интенсивности восстановления коэффициент готовности стремится к единице, т. е. система работает без сбоев. Коэффициент простоя стремится к нулю, т. е. модель не простаивает.

При увеличении интенсивности отказа рабочих каналов, время наработки на отказ модели также увеличивается (рис. 3).

Если увеличивать интенсивность отказа переключающего устройства, время наработки на отказ тоже увеличивается.

При увеличении интенсивности восстановления рабочих каналов, время восстановления системы уменьшается (рис. 4).

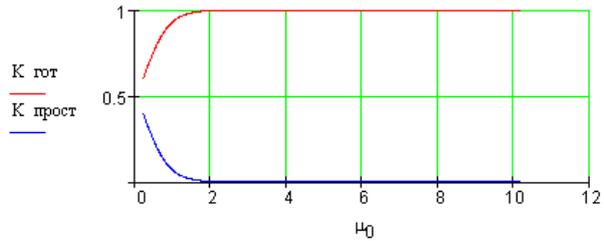


Рис. 2. Зависимость коэффициентов готовности системы $K_{\text{гот}}$ и восстановления системы $K_{\text{прост}}$ от интенсивности восстановления системы μ_0

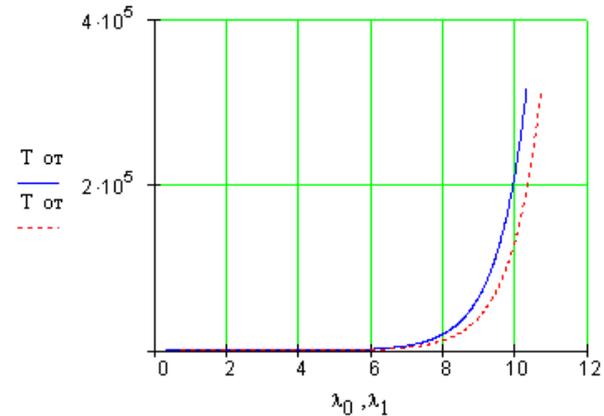


Рис. 3. Зависимость времени нахождения системы в отказе $T_{\text{от}}$ от интенсивности отказа основного λ_0 и резервного λ_1 каналов

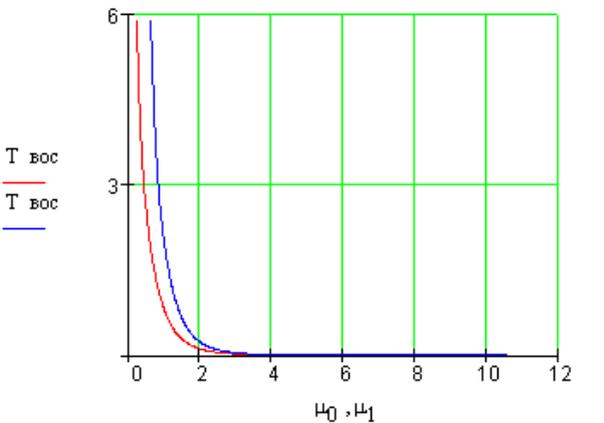


Рис. 4. Зависимость времени восстановления системы $T_{\text{вос}}$ от интенсивности восстановления основного μ_0 и резервного μ_1 каналов

При увеличении интенсивности восстановления переключающего устройства, время восстановления модели также увеличивается.

В результате анализа построенных моделей студенты могут видеть отличие между имитационным и математическим моделированием. Сравнить результаты функционирования трактов передачи данных с кратным и дробным ре-

зервированием. Видеть, как реализован алгоритм функционирования канала в модели. Представить реально работающую систему в виде имитационной модели. Практика самостоятельной доработки модели, ее изменения, разработка своих алгоритмов функционирования также возможна в представленной модели.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Камаев, В. А. Когнитивный анализ качества подготовки специалистов в вузах / В. А. Камаев, М. А. Заболотский, И. А. Полякова, А. В. Тихонин // Современные наукоемкие технологии. – 2005. – № 6. – С. 25–26.
2. Крушель, Е. Г. Компьютерная поддержка обучения студентов-системотехников основам проектирования

АСОИУ / Е. Г. Крушель, А. Э. Панфилов, В. Г. Семенов // Известия Волгоградского государственного технического университета: межвуз. сб. науч. ст. Сер: Актуальные проблемы управления, вычислительной техники и информатики в технических системах. Вып. 2. – Волгоград: РПК «Политехник», 2007. – № 2(28). – С. 72–75.

3. Лукьянов, В. С. Надежность в АСОИУ: учеб. пособие / В. С. Лукьянов, Е. С. Кузнецова: Курс лекций. Часть 1; Волг. гос. техн. ун-т. – Волгоград: РПК «Политехник», 2007. – 99 с.

4. Дружинин, Г. В. Надежность автоматизированных систем / Г. В. Дружинин. – 3-е изд., перераб. и доп. – М.: «Энергия», 1977. – 536 с.

5. ГОСТ 27.002-89. Надежность в технике. Основные понятия. Термины и определения; введ. 01.07.90. – М.: Стандартинформ, 1989. – 27 с. (Межгосударственный стандарт).

УДК 003.26

С. А. Македонский, В. С. Лукьянов

АНАЛИЗ СИСТЕМ ПРОВЕДЕНИЯ ЭЛЕКТРОННОГО ГОЛОСОВАНИЯ

Волгоградский государственный технический университет
ecmsys@vstu.ru

Статья посвящена проведению анализа реализованных систем и разработанных протоколов проведения защищенного электронного голосования.

Ключевые слова: Электронное голосование, электронные выборы, тайное электронное голосование.

S. A. Makedonskiy, V. S. Lukjanov

ANALYSIS SYSTEM FOR ELECTRONIC VOTING

Volgograd State Technical University

This article is dedicated to analyze existing systems and protocols of the secret electronic voting.

Key words: Electronic voting, electronic election, secret electronic voting.

Все большее развитие государственных информационных систем создает условия для разработки и внедрения современных информационных средств, позволяющих автоматизировать и тем самым более эффективно реализовывать программы федеральных, региональных, отраслевых и межведомственных уровней. Одним из примеров таких программ, которые можно автоматизировать, являются проводимые периодически государственные выборы, например, в государственную думу или президентские выборы. Создание системы электронного голосования позволит автоматизировать процесс выборов, существенно сократить расходы на выборы, повысить скорость и уменьшить вероятность ошибок при подсчете голосов.

Системы проведения защищенного электронного голосования должны отвечать ряду требований, среди которых [1–3]:

- 1) обеспечение тайны (анонимности) волеизъявления участника голосования;
- 2) обеспечение одноразового учета голоса участника голосования;
- 3) предотвращение возможности дублирования голоса какого-то другого участника выборов;
- 4) обеспечение строгой однозначной идентификации участников голосования;
- 5) обеспечение достоверности переданных сообщений (например, использование электронно-цифровых подписей);
- 6) обеспечение корректности подсчета конечного результата;
- 7) обеспечение возможности проверки любым из участников, что результат подсчитан правильно;
- 8) обеспечение работоспособности протокола в случае, когда некоторые из его участников нечестны;

9) обеспечение бесперебойной работы программно-технических средств.

К настоящему времени имеется ряд работ в данной области, а также реализованные и опробованные системы электронного голосования [4–6]. Например, в Эстонии с осени 2005 года муниципальные выборы проводятся через Интернет, в Казахстане система электронного голосования «Сайлау», закупленная в Белоруссии, впервые испытывалась на 10% избирательных участков на выборах депутатов в 2004-м году и с тех пор активно внедряется по всей республике. В России 12-го октября 2008 года по инициативе Тульской областной избирательной комиссии, при поддержке центральной избирательной комиссией (ЦИК) России, в порядке эксперимента было проведено электронное голосование, для которого использовались возможности сети Интернет. Позже этот эксперимент был повторен 1 марта 2009 года в Вологодской, Волгоградской и Томской областях, а также 11 октября 2009 года в городе Кингисепп Ленинградской области.

Проведение анализа систем электронного голосования в Эстонии, Казахстане и России на основании перечисленных выше требований, предъявляемых к таким системам, выявило следующие сильные и слабые стороны этих систем.

Система электронного голосования Эстонии позволяет голосовать через Интернет прямо из дома, достаточно иметь компьютер с выходом в интернет и устройство для работы с электронным удостоверением личности. Систему можно считать удобной для пользователя.

Однако большому риску в этой системе подвергается тайна голосования, что связано с используемым алгоритмом обеспечения этого требования. Тайна голосования обеспечивается за счет того, что функции по проверке подписи бюллетеней и их расшифровке разделены между двумя государственными организациями. В случае сговора сотрудников, имеющих доступ к бюллетеням и секретному ключу, система окажется скомпрометированной. Таким образом, у избирателя нет гарантии, что его голос не станет известен сторонним лицам.

Тайна голосования в системе электронного голосования Казахстана «Сайлау» обеспечивается за счет выдачи многоцветных «электронных бюллетеней» на избирательных участках, а также за счет помещения информации о поданном голосе и о регистрации избирателя в различные логически не связанные области энер-

гонезависимой памяти в переносных устройствах для голосования.

Большое внимание в АИС «Сайлау» уделяется безопасности. Реализованные в системе меры обеспечивают надежное и бесперебойное функционирование системы, защиту от хакерских атак. Соблюдение тайны голосования обеспечивается на уровне, не уступающем, а может даже и превосходящем традиционное голосование с использованием бумажных бюллетеней. Такой уровень соблюдения тайны голосования обеспечивается за счет того, что система электронного голосования максимально уподоблена традиционному способу подачи голосов на бумажных бюллетенях.

Отметим, что единственным слабым местом в плане обеспечения тайны голосования является переносное устройство для голосования. Связано это с тем, что у избирателя не может быть полной уверенности в том, что это устройство работает так, как это заявлено производителем. А именно, информация о регистрационных данных избирателя и его голосе могут заноситься не в независимые области памяти, а специально структурироваться так, чтобы впоследствии было ясно, кто как голосовал, и избиратель повлиять на это никак не сможет.

Однако АИС «Сайлау» по сути представляет собой лишь средство автоматизации подсчета голосов и призвано обеспечить существенное повышение эффективности работы избирательных комиссий всех уровней, степени доверия населения республики, международных организаций и субъектов политической деятельности к результатам выборов. Для функционирования системы требуется наличие большого количества обслуживающего персонала и соответственных затрат на их содержание.

Другими словами от системы электронного голосования можно получить гораздо больше преимуществ в случае еще большей автоматизации процессов проведения выборов, как, например, в Эстонии.

В России при проведении экспериментов использовались технологии, позволяющие для подачи голоса использовать диск электронного опроса, мобильный телефон или электронную социальную карту. Однако такие технологии не подходят, например, для выборов всероссийского масштаба, в первую очередь потому, что задачей экспериментов являлось, прежде всего, изучение отношения избирателей к новым формам голосования, и поэтому выполнение

предъявляемых к системе требований по безопасности и соблюдении тайны голосования, гласности и достоверности результатов было далеко не самым важным.

Таким образом, имеющиеся в распоряжении центральной избирательной комиссии России технологии проведения голосования в электронном виде не способны обеспечить выполнение предъявляемых к таким системам требований.

По результатам анализа существующих систем проведения электронного голосования можно заключить, что из всех требований, предъявляемых к таким системам, наиболее сложно реализуемым оказывается требование соблюдения тайны голосования.

Для того, чтобы системы удовлетворяли данному требованию, разработан ряд протоколов, использующих различные подходы для решения этой проблемы. Эти протоколы могут лежать в основе реальных систем электронного голосования, однако не один из них не используется в рассмотренных системах.

Существующие протоколы голосования можно разделить на три типа [1–3]:

- 1) протоколы голосования с перемешиванием;
- 2) протоколы голосования с применением технологии слепой подписи;
- 3) протоколы голосования с разделением, в которых личные бюллетени делятся между различными счетными комиссиями так, что ни одна из них не сможет фальсифицировать результаты голосования.

Проведенный анализ этих протоколов выявил их следующие сильные и слабые стороны.

В протоколах с перемешиванием тайна голосования обеспечивается за счет того, что поданные избирателями голоса перемешиваются организаторами выборов таким образом, что в дальнейшем невозможно восстановить, какой из голосов какому из избирателей принадлежал. А участие в выборах большого числа организаторов, участвующих в перемешивании голосов, является гарантией того, что поданные голоса не станут никому известны, потому что только в случае сговора всех организаторов возможно будет определить, кто как проголосовал. Анализ протоколов с перемешиванием выявил следующие существенные недостатки:

1. Невозможно предотвратить мошенничество со стороны избирателя, который утверждает, что его голос не подсчитан. Если организаторам разрешить отслеживать путь рас-

шифровки поданных голосов таких избирателей на доске бюллетеней, то под угрозой оказывается анонимность избирателей. Следовательно, не выполняется требование 6 – обеспечение корректности подсчета конечного результата, а также требование 8 – обеспечение работоспособности протокола в случае, когда некоторые из его участников нечестны.

2. Если обнаружены ошибки (возможно, кто-то из организаторов таким способом намерено сорвал выборы), то выборы придется проводить заново, а вся информация о голосах уже раскрыта. То есть не выполняется требование 8 – обеспечение работоспособности протокола в случае, когда некоторые из его участников нечестны.

Особенность протоколов голосования с применением слепой подписи заключается в том, что каждому избирателю, имеющему право голоса, при проведении выборов организаторами выдается бюллетень в электронном виде. Выданный бюллетень подписан электронно-цифровой подписью организатора, тем самым давая его владельцу право голоса. Причем организаторы не могут узнать, какой из бюллетеней кому из избирателей был выдан, что достигается применением протокола слепой подписи. Данный протокол позволяет подписать сообщение, содержание которого останется неизвестным подписывающему.

Анализ этого протокола выявил следующие существенные недостатки:

1. В случае, если получение бюллетеня и подача голоса избирателем происходит в течение одного установленного сеанса клиентской программы избирателя с программой избирательной комиссии, то избирательная комиссия может запомнить, какому избирателю были выданы замаскированные бюллетени и, соответственно, от какого избирателя был получен конкретный голос. Следовательно, не выполняется требование 1 – обеспечение тайны (анонимности) волеизъявления участника голосования;

2. После проведения голосования имеется возможность добавления со стороны избирательной комиссии произвольного количества голосов, так как избиратель может проверить корректность подсчета только своего голоса. Следовательно, не выполняется требование 3 – предотвращение возможности дублирования голоса какого-либо другого участника выборов.

Протокол голосования с разделением обеспечивает выполнение требования тайны голо-

сования за счет того, что полученные избирателем бюллетени разделяются между организаторами выборов, то есть между всеми участвующими в выборах избирательными комиссиями.

Анализ этого протокола выявил следующие существенные недостатки:

1. Отсутствуют средства контроля со стороны избирателя за функционированием счетных комиссий. Однако участие в выборах большого числа независимых избирательных комиссий гарантирует соблюдение тайны голосования, так как ни у одной из них не будет возможности самостоятельно определить, кто как проголосовал. Определить итоги голосования могут только все вместе. Для этого каждая избирательная комиссия предоставляет итоги проведенных ими вычислений с поданными частями голосов. Никто из участников голосования не сможет подать голос за другого, так как для этого необходимо участие всех избирательных комиссий.

2. Протокол позволяет избирателю осуществлять выбор только из двух возможных значений. В протоколе невозможно осуществлять выбор из произвольного количества вариантов, а также выбор нескольких кандидатов из списка.

Таким образом, в протоколе с разделением соблюдаются все требования, перечисленные в начале статьи при условии участия в выборах достаточного числа независимых избирательных комиссий.

Обобщая полученные результаты анализа, можно сказать, что ни одна из реализованных и используемых систем электронного голосования либо не отвечает в полной мере предъявляемым к таким системам требованиям, либо не достаточно эффективна, с точки зрения уровня автоматизации. Из предложенных на сегодняшний день протоколов проведения тайного голосования протокол голосования с разделением лучше других отвечает предъявляемым к нему требованиям. Однако требуются дальнейшие исследования, так как для корректной работы протокола необходимо участие большо-

го количества независимых избирательных комиссий, что накладывает на этот протокол определенные ограничения, связанные с объемом вычислений при его реализации.

При этом возникает две задачи:

1. При каких условиях возможна реализация протокола электронного голосования с разделением так, чтобы он обеспечивал соблюдение тайны голосования на должном уровне и бесперебойную работу всех частей системы. Для решения этой задачи целесообразным является создание имитационной модели системы электронного голосования, что является целью новых исследований.

2. Поиск решения, в котором протокол голосования с разделением позволит осуществлять выбор не только одного варианта из двух, но и из произвольного количества вариантов, а также выбор нескольких кандидатов из списка.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Смарт Н.* Мир программирования. Криптография. – Москва: Техносфера, 2005.
2. *Шнайер Б.* Прикладная криптография. 2-е издание. – Москва: Триумф, 2002.
3. *Лифшиц Ю.* «Электронные выборы» [Электронный ресурс] – Режим доступа: <http://yury.name/cryptography/>, 2005.
4. Обзор Системы Электронного Голосования, НИК Эстонии. – Национальная Избирательная Комиссия Эстонии. Таллин 2005.
5. Электронные выборы: опасение фальсификации результатов. – Kazakhstan today 2004 г.
6. *Липень В. Ю., Воронецкий М. А., Липень Д. В.* Результаты апробирования технологий и систем электронного голосования. – United Institute of Informatics Problems NASB 2002.
7. *Македонский, С. А.* Система электронного голосования / С. А. Македонский, В. С. Лукьянов // Открытое образование: [по матер. XXXVI междунар. конф. и дискус. науч. клуба IT+SE'09, майская сессия, Ялта-Гурзуф]. – 2009. – Приложение к журн. – С. 126–128.
8. *Македонский, С. А.* Система электронного голосования / С. А. Македонский, В. С. Лукьянов // Открытое образование: [по матер. XXXVII междунар. конф. и дискус. науч. клуба IT+SE'10, майская сессия, Ялта-Гурзуф]. – 2010. – Приложение к журн. – С. 121–123.

УДК 003.26

*С. А. Македонский, В. С. Лукьянов***УНИВЕРСАЛЬНЫЙ ПРОТОКОЛ ЗАЩИЩЕННОГО ЭЛЕКТРОННОГО ГОЛОСОВАНИЯ****Волгоградский государственный технический университет**
ecmsys@vstu.ru

Статья посвящена разработке протокола электронного голосования, который позволит производить защищенное голосование произвольной формы.

Ключевые слова: Электронное голосование, электронные выборы, тайное электронное голосование.

*S. A. Makedonskiy, V. S. Lukjanov***UNIVERSAL PROTOCOL PROTECTED ELECTRONIC VOTING****Volgograd State Technical University**

The article is devoted to the development of a protocol of electronic voting, which will produce a secure vote of arbitrary shape.

Key words: Electronic voting, electronic election, secret electronic voting.

К настоящему времени имеется ряд работ, посвященных созданию протоколов тайного электронного голосования. Все существующие протоколы подразделяются на три типа:

1) протоколы голосования с перемешиванием;
2) протоколы голосования с применением технологии слепой подписи;

3) протоколы голосования с разделением, в которых личные бюллетени делятся между различными счетными комиссиями так, что ни одна из них не сможет фальсифицировать итоги подсчета голосов избирателей.

Среди этих протоколов лучше других отвечает требованиям, предъявляемым к протоколам тайного голосования, протокол голосования с разделением. Однако в исходном виде, в котором протокол голосования с разделением приводится во всех рассмотренных литературных источниках [1–3], он не пригоден для проведения реального голосования, так как позволяет избирателю осуществлять выбор только из двух возможных значений. В протоколе невозможно осуществлять выбор из произвольного количества вариантов, а также выбор нескольких кандидатов из списка. Для того, чтобы протокол голосования с разделением можно было применять в реальных выборах, предлагается следующая его модификация.

Предполагается, что в голосовании участвуют m лиц с правом голоса и n счетных комиссий. Использование большого числа счетных комиссий обеспечивает анонимность голосующего и предотвращает возможность фальсификации результатов голосования.

Система электронного голосования может иметь следующий вид:

1. Установки системы.

Каждая из счетных комиссий обладает шифрующей функцией с открытым ключом E_i . Будем предполагать, что фиксирована конечная абелева группа A простого порядка Q , в которой выбрана пара элементов B и G , причем никто (включая счетные комиссии) не знает решения уравнения

$$B = G^x \pmod{Q} \quad (1)$$

Каждый из голосующих имеет алгоритм подписи с открытым ключом.

2. Заполнение бюллетеня.

Для подачи голоса каждый j -й избиратель отдельно выбирает голос по каждому k -му пункту голосования:

$$v_{j,k} \in \{-1, 1\} \quad (2)$$

и случайные затемняющие числа

$$a_{j,k} \in Z/QZ \quad (3)$$

(где Z/QZ – множество всех возможных остатков от деления чисел на Q) и публикует свои решения:

$$D_{j,k} = D_{a_{j,k}}(v_{j,k}) = B^{v_{j,k}} G^{a_{j,k}} \pmod{Q}, \quad (4)$$

используя схему обязательств [1]. Эти решения становятся известными всем сторонам голосования: как счетным комиссиям, так и остальным избирателям. Вместе с $D_{j,k}$ избиратель публикует автономные версии протокола доказательства с нулевым разглашением [1] в подтверждение того, что его голоса действительно выбраны из множества $\{-1, 1\}$. Голоса вместе с доказательствами подписываются с помощью схемы подписи, принадлежащей избирателю.

Также для доказательства того, что голоса сформированы корректно, каждый j -й избира-

тель публикует сумму затемняющих чисел $a_j = \sum_k a_{j,k}$. Так как для данного значения обязательства и каждого передаваемого числа существует лишь одно значение затемняющего числа, невозможно подобрать иное значение, позволяющее избирателю сфальсифицировать его голоса, и только избиратель сможет вычислить эту сумму, так как только ему известны все значения затемняющих чисел.

3. Распределение бюллетеней.

Для подведения итогов необходимо передать отданные голоса счетным комиссиям. Для передачи значений $a_{j,k}$ и $v_{j,k}$ каждый избиратель применяет схему Шамира разделения секрета [1]. С этой целью для каждого k -го голоса он выбирает два случайных многочлена по модулю Q степени $T < n$

$$\begin{aligned} R_{j,k}(X) &= v_{j,k} + r_{1,j,k}X + \dots + r_{T,j,k}X^T \\ S_{j,k}(X) &= a_{j,k} + s_{1,j,k}X + \dots + s_{T,j,k}X^T \end{aligned} \quad (5)$$

и вычисляет

$$(u_{i,j,k}, w_{i,j,k}) = (R_{j,k}(i), S_{j,k}(i)) \text{ при } 1 \leq i \leq n \quad (6)$$

Голосующий шифрует пары $(u_{i,j,k}, w_{i,j,k})$, используя алгоритм E_i i -й счетной комиссии, и отправляет ей полученный результат. После этого избиратель передает многочлен $R_{j,k}(X)$, открыто регистрируя

$$\begin{aligned} D_{l,j,k} &= D_{s_{l,j,k}}(r_{l,j,k}) = \\ &= B^{r_{l,j,k}} G^{s_{l,j,k}} \pmod{Q} \text{ при } 1 \leq l \leq T \end{aligned} \quad (7)$$

с помощью все той же схемы обязательств.

4. Проверка достоверности информации.

Каждая из счетных комиссий должна проверить, что пара $(u_{i,j,k}, w_{i,j,k})$ действительно получена от избирателя с номером j и согласуется с переданным обязательством. Это достигается в результате проверки следующих равенств:

$$\begin{aligned} D_{j,k} \prod_{l=1}^T D_{l,j,k}^{i^l} \pmod{Q} &= \\ &= D_{a_{j,k}}(v_{j,k}) \prod_{l=1}^T D_{s_{l,j,k}}(r_{l,j,k})^{i^l} \pmod{Q} = \\ &= B^{v_{j,k}} G^{a_{j,k}} \prod_{l=1}^T (B^{r_{l,j,k}} G^{s_{l,j,k}})^{i^l} \pmod{Q} = \\ &= B^{(v_{j,k} + \sum_{l=1}^T r_{l,j,k} i^l)} G^{(a_{j,k} + \sum_{l=1}^T s_{l,j,k} i^l)} \pmod{Q} = \\ &= B^{u_{i,j,k}} G^{w_{i,j,k}} \pmod{Q} \end{aligned} \quad (8)$$

Кроме того, каждая счетная комиссия проверяет, что все голоса сформированы корректно, и избиратель правильно подал голоса, проголосовав только за наперед заданное количе-

ство кандидатов. Для этого избирательной комиссией вычисляется сумма v единиц количества кандидатов, за которых можно проголосовать, и отрицательных единиц количества остальных кандидатов и проверяется равенство:

$$\begin{aligned} \frac{\prod_k D_{j,k}}{B^v} \pmod{Q} &= \frac{\prod B^{v_{j,k}} G^{a_{j,k}}}{B^v} \pmod{Q} = \\ &= \frac{B^{\sum_k v_{j,k}} G^{\sum_k a_{j,k}}}{B^v} \pmod{Q} = G^{a_j} \pmod{Q} \end{aligned} \quad (9)$$

Если избиратель подал голоса корректно, то $\sum_k v_{j,k} = v$, и тогда равенство должно выполняться.

5. Подсчет голосов.

Каждая из n счетных комиссий подсчитывает бюллетени по каждому k -му кандидату и публикует результаты:

$$U_{i,k} = \sum_{j=1}^m u_{i,j,k} \quad (10)$$

Кроме того, она публикует сумму частей затемняющих чисел:

$$W_{i,k} = \sum_{j=1}^m w_{i,j,k} \quad (11)$$

Любой другой участник процедуры голосования (счетная комиссия или избиратель) может убедиться в корректности опубликованных сумм, проверяя по каждому кандидату, что

$$\begin{aligned} \prod_{j=1}^m (D_{j,k} \prod_{l=1}^T D_{l,j,k}^{i^l}) \pmod{Q} &= \\ &= \prod_{j=1}^m B^{u_{i,j,k}} G^{w_{i,j,k}} \pmod{Q} = B^{U_{i,k}} G^{W_{i,k}} \pmod{Q} \end{aligned} \quad (12)$$

Каждая из сторон процесса может определить итог, беря T значений $U_{i,k}$ и восстанавливая по ним окончательный результат. Дело в том, что $U_{i,k}$ – значение многочлена, представляющего сумму голосов, в точке i для k -го кандидата. Чтобы убедиться в этом, рассмотрим

$$\begin{aligned} U_{i,k} &= \sum_{j=1}^m u_{i,j,k} = \sum_{j=1}^m R_{j,k}(i) = \\ &= (\sum_{j=1}^m v_{j,k}) + (\sum_{j=1}^m r_{1,j,k})i + \dots + (\sum_{j=1}^m r_{T,j,k})i^T \end{aligned} \quad (13)$$

В итоге получается сумма всех выбранных избирателями значений из множества $\{-1;1\}$ для k -го кандидата. А соотнеся эту сумму с общим количеством проголосовавших и количеством кандидатов, за которых можно проголосовать, получается точный результат по количеству поданных голосов. Например, если h – ко-

личество подавших голоса избирателей, f – количество кандидатов, которые можно выбрать, то в процентном соотношении такое значение можно получить следующим образом:

$$\frac{h + \sum_{j=1}^m v_{j,k}}{2 * h * f} * 100\% \quad (14)$$

Предложенный протокол позволяет осуществлять выборы произвольной формы. Протокол предлагается использовать для создания программы электронного голосования, которая предназначена для проведения выборов в больших организациях, имеющих распределенную структуру или большое количество отделов и сотрудников. Примерами использования программы являются проведение выборов в уни-

верситете или на голосовании акционеров в крупной компании. При этом присутствие голосующих на собрании будет не обязательным – проголосовать можно будет удаленно через интернет или корпоративную распределенную сеть.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Смарт Н.* Мир программирования. Криптография. – Москва: Техносфера, 2005.
2. *Шнайер Б.* Прикладная криптография. 2-е издание. – Москва: Триумф, 2002.
3. *Лифшиц Ю.* Электронные выборы – 2005.
4. *Македонский, С. А.* Формальный анализ протокола электронного голосования / С. А. Македонский, В. С. Лукьянов // Открытое образование: [по матер. XXXVII междунар. конф. и дискус. науч. клуба IT+SE'10, осенняя сессия, Ялта-Гурзуф]. – 2010. – Приложение к журн. – С. 39–48.

Научное издание

ИЗВЕСТИЯ

**Волгоградского государственного технического университета
№ 11(71), 2010 г.**

Серия "АКТУАЛЬНЫЕ ПРОБЛЕМЫ УПРАВЛЕНИЯ, ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ
И ИНФОРМАТИКИ В ТЕХНИЧЕСКИХ СИСТЕМАХ"

Выпуск 9

Межвузовский сборник научных статей

Темплан заказных изданий 2010 г. Поз. № 79з.

Подписано в печать 30.11.2010. Формат 60×84 1/8. Бумага офсетная.
Гарнитура Times. Печать офсетная. Усл. печ. л. 18,60. Уч.-изд. л. 16,53.
Тираж 120 экз. Заказ № 768.

Волгоградский государственный технический университет.
400131, Волгоград, просп. им. В. И. Ленина, 28, корп. 1.

Отпечатано в типографии ИУНЛ ВолГТУ.
400131, Волгоград, просп. им. В. И. Ленина, 28, корп. 7.

В сборнике научных статей "Известия ВолгГТУ", серии "Актуальные проблемы управления, вычислительной техники и информатики в технических системах" публикуются статьи, которые содержат результаты теоретических и экспериментальных исследований и представляют научный и практический интерес для широкого круга специалистов в области вычислительной техники и информатики.

Вопрос об опубликовании статьи или ее отклонении решает редакционная коллегия сборника, которая утверждается ректором университета, ее решение является окончательным. Редакция направляет представленный для издания материал на рецензирование.

Рукопись должна быть набрана и сверстана в текстовом редакторе Word и распечатана на лазерном принтере в режиме полной загрузки тонера. Формат бумаги А4 (210×297 мм).

Для ускорения подготовки издания необходимо представлять файлы статей на дискетах или компакт-дисках (CD) в полном соответствии с распечатанным оригиналом. Дискета должна быть вложена в отдельный конверт, на этикетке дискеты указываются фамилии авторов статьи.

При наборе текста следует соблюдать следующие требования: поля – верхнее – 2,0 см, нижнее – 3,0 см, левое – 2,5 см, правое – 2,5 см; шрифт Times, кегль 14, интервал полуторный.

Текст набирается с применением автоматического переноса слов, перед знаками препинания (в том числе внутри скобок) пробелы не допускаются, после них ставится один пробел. Разрядка слов не допускается, следует избегать перегрузки статей большим количеством формул, рисунков, таблиц. Для набора символов в формульном редакторе MS Equation (MS Word) использовать установки (Стиль/Размеры) только по умолчанию; рисунки должны быть выполнены в редакторах векторной графики, таких как CorelDRAW или в любом приложении к Word. Допускается сканирование рисунков в программе Microsoft Photo Editor.

Инициалы и фамилия автора (авторов) пишутся над заглавием статьи. Ниже заглавия, перед основным текстом, указывается организация или предприятие, в котором работает автор статьи. В конце статьи ставится полное название учреждения, которое рекомендует статью для публикации, дата и подпись автора (всех авторов).

Литературные ссылки должны быть оформлены в соответствии с ГОСТом 7.1–2003 "Библиографическая запись. Библиографическое описание. Общие требования и правила составления". Библиографический список использованной литературы, составленный в порядке упоминания в тексте, дается в конце статьи; связь с основным текстом осуществляется цифровым порядковым номером в квадратных скобках в строке. Подстрочные ссылки не допускаются.

Иностранные фамилии и термины в тексте следует приводить в русском переводе. В библиографическом списке фамилии авторов, полное название книг и журналов приводится на языке оригинала.

Ссылки на неопубликованные работы не допускаются.

При обозначении единиц физических величин должна применяться Международная система единиц (СИ).

Объем статьи не должен превышать 8 страниц бумаги формата А4, включая таблицы и библиографический список; число рисунков – не более четырех, включая рисунки, помеченные буквами *a*, *b*, и т. д. Рекомендуется включать в сборник статьи с авторским коллективом не более четырех человек с участием каждого автора в одной–двух статьях.

Статьи должны представлять сжатое четкое изложение результатов, полученных автором без повторов приводимых данных в тексте статьи, таблицах и рисунках. К статье должны быть приложены: сведения об авторах (полные имя, отчество, фамилия, ученая степень, звание, домашний адрес, номера телефонов – служебного и домашнего, E-mail), документация, подтверждающая возможность ее открытого опубликования.