

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Том 22
2016
№ 9

ТЕОРЕТИЧЕСКИЙ И ПРИКЛАДНОЙ НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ

Издается с ноября 1995 г.

УЧРЕДИТЕЛЬ
Издательство "Новые технологии"

СОДЕРЖАНИЕ

ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

- Барский А. Б. Моделирование индуктивного мышления с помощью языка логического вывода ПРОЛОГ 643
- Маренко В. А. Применение когнитивного моделирования для исследования межэтнических отношений 649

МОДЕЛИРОВАНИЕ И ОПТИМИЗАЦИЯ

- Светушков Н. Н. Упрощенная математическая модель расчета температурных полей при детонационном горении 655
- Гейдаров П. Ш. Алгоритм кратчайшего маршрута на основе выделенного набора маршрутов 660
- Мельников Г. А., Мельников Т. А., Губарев В. В. Алгоритмы упрощения деревьев регрессии: обзор и эмпирическое сравнение 669

ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ И СЕТИ

- Печенко И. С. Способы представления спецификаций вычислительных систем: проблемы и возможности машинной обработки. 676

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ОБРАЗОВАНИИ

- Вишнеков А. В., Иванова Е. М. Автоматизация выбора траектории обучения в среде Smart-образования 684

ЦИФРОВАЯ ОБРАБОТКА СИГНАЛОВ И ИЗОБРАЖЕНИЙ

- Мальцев А. В., Михайлюк М. В., Тимохин П. Ю., Торгашев М. А. Технология разделения видеопотоков в дистанционном обучении с помощью нескольких устройств Kinect 692

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В БИМЕДИЦИНСКИХ СИСТЕМАХ

- Абдуллаев Н. Т., Мамедов Н. Я., Агаева Г. С. Применение метода спектрального анализа для дифференциальной диагностики заболевания органов желудочно-кишечного тракта 699

Журнал в журнале НЕЙРОСЕТЕВЫЕ ТЕХНОЛОГИИ

- Кулагин В. П., Иванов А. И., Серикова Ю. И. Корректировка методических и случайных составляющих погрешностей вычисления коэффициентов корреляции, возникающих на малых выборках биометрических данных. 705
- Аведьян Э. Д., Ле Т. Ч. Л. Двухуровневая система обнаружения DoS-атак и их компонентов на основе нейронных сетей СМАС 711

Главный редактор:
СТЕМПКОВСКИЙ А. Л.,
акад. РАН, д. т. н., проф.

Зам. главного редактора:
ИВАННИКОВ А. Д., д. т. н., проф.
ФИЛИМОНОВ Н. Б., д. т. н., с.н.с.

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

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

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

Информация о журнале доступна по сети Internet по адресу <http://novtex.ru/IT>.
Журнал включен в систему Российского индекса научного цитирования.
Журнал входит в Перечень научных журналов, в которых по рекомендации ВАК РФ должны быть опубликованы научные результаты диссертаций на соискание ученой степени доктора и кандидата наук.

CONTENTS

INTELLIGENT SYSTEMS AND TECHNOLOGIES

- Barsky A. B.** Simulation of Inductive Reasoning Using Inference PROLOG 643
Marenko V. A. Cognitive Modelling for the Interethnic Relations Study 649

MODELING AND OPTIMIZATION

- Svetushkov N. N.** Simplified Mathematical Model for Temperature Fields Calculation in Detonation Combustion. 655
Geidarov P. Sh. Algorithm for the Shortest Route Based on the Selected Set of Routes 660
Melnikov G. A., Melnikov T. A., Gubarev V. V. Regression Tree Pruning Algorithms: an Overview and Empirical Comparison 669

COMPUTING SYSTEMS AND NETWORKS

- Pechenko I. S.** Specification Representation Forms: Issues and Ways for Automatic Processing 676

INFORMATION TECHNOLOGIES IN EDUCATION

- Vishnekov A. V., Ivanova E. M.** Automation of the Smart-Education Trajectory Choice 684

DIGITAL PROCESSING OF SIGNALS AND IMAGES

- Maltsev A. V., Mikhaylyuk M. V., Timokhin P. Yu., Torgashev M. A.** The Technology of Video Stream Separation by Means of Multiple Kinect Devices in Distance Education 692

INFORMATION TECHNOLOGIES IN BIOMEDICAL SYSTEMS

- Abdullayev N. T., Mamedov N. Ya., Agayeva G. E.** Application of the Method the Spectral Analysis for Differential Diagnosis the Disease Bodies of the Digestive Tract 699

Journal-in-journal NEUROTECHNOLOGIES

- Kulagin V. P., Ivanov A. I., Serikova Yu. I.** Correction of Methodical and Casual Components of Errors of Calculation of the Coefficients of Correlation Arising on Small Selections of Biometric Data 705
Aved'yan E. D., Le T. T. L. A Two-Level System for DoS attacks and their Components Detection based on the Neural Networks CMAC 711

Editor-in-Chief:

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

Deputy Editor-in-Chief:

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

Chairman:

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

Editorial Board Members:

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

Editors:

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

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

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

УДК 004.032

А. Б. Барский, д-р техн. наук, проф., e-mail: arkbarsk@mail.ru
Московский государственный университет путей сообщения (МИИТ)

Моделирование индуктивного мышления с помощью языка логического вывода ПРОЛОГ

В соответствии с двумя парадигмами искусственного интеллекта — эксперта и ученика — исследуются возможности построения моделей дедуктивного и индуктивного мышления человека на базе языка логического вывода ПРОЛОГ. Если моделирование дедуктивного мышления изучено достаточно и является основной целью данного языка, то модель индуктивного мышления, т. е. формирования новых знаний, предлагается впервые. Суть ее в следующем. По базе знаний, состоящей из фактов и правил, строятся все возможные дедуктивные цепочки логического вывода. В них обнаруживаются и выделяются законченные, желательные повторяющиеся, конструкции. Связанные переменные заменяются их абстрактным представлением для формирования обобщенного вида выделенной конструкции. Так создаются гипотезы об описании новых понятий. Этим гипотетическим понятиям присваиваются имена, и они становятся новыми правилами, дополняющими базу знаний. Одновременно описание новых правил дополняет понятийную логическую нейронную сеть для возможности работы с нечеткими данными. Практика успешного, непротиворечивого применения новых правил должна утвердить их высокую достоверность.

Ключевые слова: парадигмы искусственного интеллекта, дедуктивное и индуктивное мышление, логическая нейронная сеть, ПРОЛОГ, логические цепочки

Введение

По-видимому, единственное систематизированное определение парадигм искусственного интеллекта (ИИ), явившееся отправной точкой моделирования, дал Н. М. Амосов [1].

Важные основополагающие результаты в области ИИ принадлежат Д. А. Поспелову [2, 3]. На базе этих публикаций развиваются идеи логического вывода по нейронным сетям, например, в работах [4–7].

Парадигма — это концептуальное представление сути проблемы и путей ее решения. Изучают две парадигмы искусственного интеллекта.

1. *Парадигма эксперта* предполагает следующие действия при создании системы ИИ:

- **формализация знаний**, т.е. преобразование проблемного знания в форму, предписанную выбранной моделью;
- **формирование базы знаний (БЗ)** — вложение формализованных знаний в программную систему;
- **дедукция** — решение задачи логического вывода на основе БЗ. (Таким образом, БЗ отличается от базы данных (БД) возможностью логического вывода.)

Эта парадигма лежит в основе построения экспертных систем, систем логического вывода, в том числе на языке логического программирования

ПРОЛОГ. Считается, что системы на основе этой парадигмы более изучены.

2. *Парадигма ученика* предполагает следующие действия:

- **обработка фактов, наблюдений и опыта** частных примеров — формирование БД системы ИИ;
- **индуктивное обучение** — превращение БД в БЗ на основе обобщения накопленных знаний и обоснование процедуры извлечения знаний из БЗ. Это означает, что формируются *новые знания* путем вскрытия обобщенной зависимости между объектами, которую мы наблюдаем. Основное внимание здесь уделяется изучению аппроксимирующих, вероятностных и логических механизмов получения общих выводов из частных утверждений. Затем можно обосновать, например, высокую эффективность процедуры ассоциативного поиска в БЗ с помощью аппарата логических нейронных сетей;
- **дедукция** — по обоснованной процедуре выбирается информация из БЗ по запросу (например, оптимальная стратегия управления по вектору, характеризующему сложившуюся ситуацию).

Считается, что исследования в рамках этой парадигмы проведены пока недостаточно ввиду ее неисчерпаемости. Особая трудность заключается в построении моделей индуктивного обучения или,

в более широком понимании, — моделей индуктивного мышления человека.

Из чего следует исходить, приступая к моделированию индуктивного мышления?

1. Наука призвана открывать новые (не известные ранее) законы природы — законы взаимодействия объектов и явлений.

2. "Наука начинается тогда, когда начинают считать", — сказал Д. И. Менделеев. Действительно, вглядываясь в достижения мировых ученых, обнаруживаем важный факт статистического обеспечения экспериментов и выводов. Сколько раз, по известной легенде, должно было упасть яблоко на голову И. Ньютона, чтобы он сформулировал гипотезу: $F = ma$? (Ведь для подтверждения гениальной догадки яблоки должны были падать и падать!)

3. Значит, кроме статистической обработки опыта наука питается логической выводимостью новых знаний. В "простейшем" случае — это математические теоремы. На основе каких статистических оценок Д. Буль мог создать "булеву алгебру"? Да и И. Ньютон, если верить легенде, мог сделать свой вывод на основе измерения получаемой (его головой) энергии $W = mgh$. Затем, в результате аналитических выкладок, он смог перейти к столь важному основополагающему заключению. Даже в наши дни появляются факты, подтверждающие математически выведенные гипотезы большой давности.

Кстати, а почему F равно ma , а не, скажем, ma^2 ? Ведь на основе такого предположения, такой аксиоматики, вполне можно построить некоторый мир. Физики-математики это знают, препарировав вдоль и поперек второй закон механики для вывода новых знаний (учеников) о нашем мире. А по законам диалектики Гегеля все действительное логично (логически выводимо), а все логически выводимое — действительно!¹ Это что — вопрос о параллельных мирах?

Пытаясь расширить действие второго закона механики, мы можем высказать смелую гипотезу:

В Мире Божием действует закон $F = ma^R$, где R — действительное, принимающее значение от 0 до ∞ .

Мы сможем это проверить сегодня экспериментально? Но ведь возможно, что более развитые существа научились осуществлять дрейф по данному показателю степени, и нечеткие НЛО, внезапно появляющиеся и исчезающие, это демонстрируют (если это не искусный обман).

¹ Получила распространение неправильная трактовка слова "логично", воспринятое при раннем изучении трудов Гегеля как "разумно, правильно". Это слово следует воспринимать как "охватываемое разумом, логически выводимо". "Тогда фашизм разумен?!" — объясняют коммунистические идеологи исключение первой части данного тезиса из официальной марксистской философии. А. И. Герцен в "Былое и думы" объясняет "дурно понятую фразу Гегеля". Фраза "Все действительное разумно" — "иначе высказанное начало *достаточной причины* и соответственности логики и фактов".

Таким образом, мы наблюдаем неразрывную связь между дедуктивным и индуктивным мышлением, опирающуюся на многообразие законов природы. Однако важно осознать приоритет: индуктивное мышление строится на основе дедуктивного и далее питает его.

Для формального движения по замкнутому кругу способов мышления необходим формализованный язык логического вывода. Возможно, авторы языка ПРОЛОГ (в работе [8] язык хорошо освещен в учебных целях) не ожидали от него столь ценных возможностей, что оказалось на деле.

Формирование базы знаний на языке ПРОЛОГ

Как известно [8], БЗ задается множеством *фактов*, объединенных в *процедуры*, и множеством *правил* (вывода). Несколько отойдя от специфической терминологии, рассмотрим пример БЗ, описывающей родословную жителей далекого села, посещаемого лишь залетными молодцами (рис. 1).

Факты:

Процедура "мужчина"

мужчина (Иван)

мужчина (Василий)

мужчина (Петр)

мужчина (Федор)

мужчина (Юрий)

Процедура "женщина"

женщина (Марья)

женщина (Елена)

женщина (Ольга)

женщина (Ирина)

Процедура "родитель"

родитель (Марья, Иван) (читать: Марья родитель Ивана)

родитель (Марья, Василий)

родитель (Юрий, Федор)

родитель (Елена, Федор)

родитель (Елена, Марья)

родитель (Ольга, Петр)

родитель (Ольга, Ирина)

родитель (Федор, Ирина)

родитель (Федор, Петр)

родитель (Петр, Иван)

родитель (Петр, Василий)

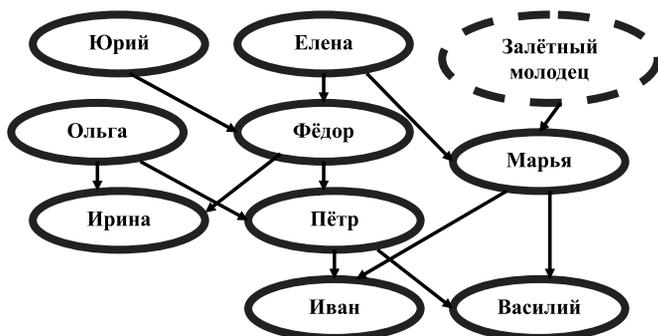


Рис. 1. Генеалогическое дерево

Правила:
мать(X, Y):- *женщина*(X), *родитель*(X, Y);
отец(X, Y):- *мужчина*(X), *родитель*(X, Y);
брат(X, Y):- *мужчина*(X), *родитель*(P, X);
родитель(P, Y), X<>Y;
сестра(X, Y):- *женщина*(X), *родитель*(P, X);
родитель(P, Y), X<>Y;
дядя(X, Y):- *брат*(X, P), *родитель*(P, Y);
тетя(X, Y):- *сестра*(X, P), *родитель*(P, Y).

Правила рекурсивно определяют понятия, которыми можно оперировать на данном уровне знаний о рассматриваемом населенном пункте. Очевидно, что расширение списка понятий, проводимое на основе анализа связей, зафиксированных в БЗ, т.е. введение новых понятий, и является целью индуктивного мышления.

Построим логические нейронные сети [5], отображающие наш объект исследования в динамике его развития. Этим мы совершим переход в область нечетких данных, столь актуальный для следственных органов.

Фактографическая нейронная сеть представлена на рис. 2. В ней выделены и те связи, которые будут доказаны ниже.

Целесообразно использовать следующую функцию активации *i*-го нейрона данной однослойной сети:

$$f_i = \begin{cases} \frac{1}{k_i} \sum_{j=1}^{k_i} v_j, & \text{если эта сумма} \geq h, \\ 0 & \text{в противном случае.} \end{cases}$$

Здесь v_j — величина возбуждения *j*-го рецептора, связанного с *i*-м нейроном, k_i — число входов этого нейрона; h — порог.

Мы видим, что, несмотря на требование (к системам принятия решений) однозначности выводов, неоднозначность в данном случае, например, о том, что Елена одновременно и бабушка, и прабабушка Ивана, также справедлива и может приниматься к действию.

На рис. 3 отображена понятийная нейронная сеть. Для сокращения числа рисунков здесь также отображены новые выводы, определяющие понятия, которые предполагается получить на основе моделирования индуктивного мышления.

Целесообразно использовать ту же функцию активации.

Дедуктивное мышление

Для проверки корректности формулировки решим задачу логического вывода методом дедуктивного мышления. Процедуры этого вывода встроены в язык ПРОЛОГ. Сформулируем *цель*, т.е. определим ту конкретную задачу, для решения которой достаточно (или не достаточно) имеющихся знаний.

Например, поставим целью выявление всех пар (X, Y), для которых справедливо отношение

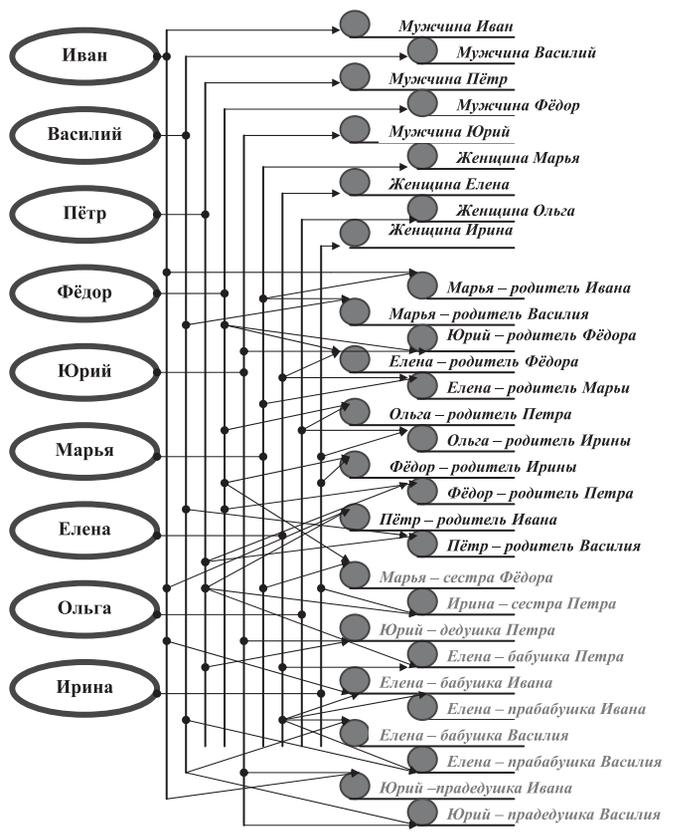


Рис. 2. Фактографическая нейронная сеть

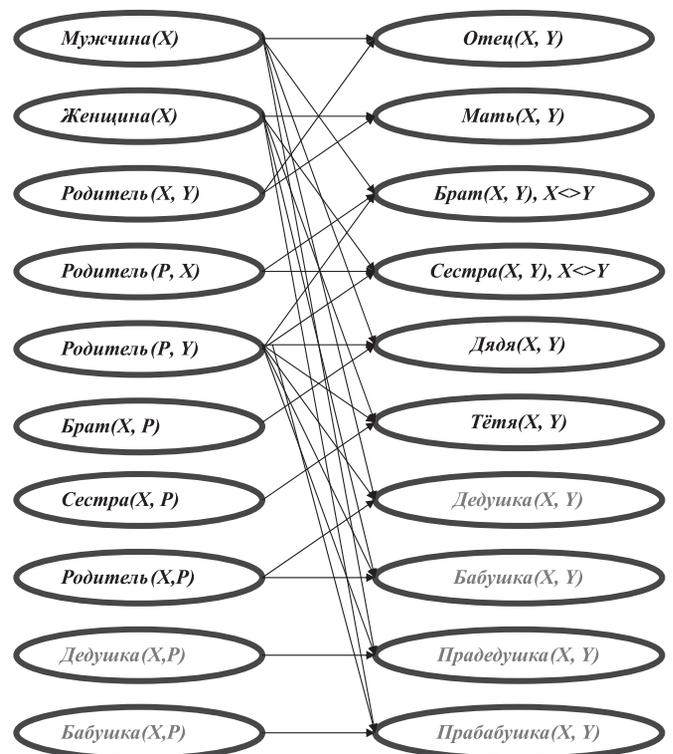


Рис. 3. Понятийная нейронная сеть

сестра(X, Y). Для этого используется метод прямого, пошагового перебора вариантов *связывания переменных*, с возвратом к предыдущему шагу в случае образующегося противоречия. Этот метод называется *backtracking*. Он воспроизводит стратегию "ветвей и границ", где "границы" определяются логически противоречивыми действиями, не допускающими развитие варианта связывания переменных. Про противоречивую попытку связывания переменного говорят: не проходит унификация. Установление факта унификации не всегда возможно "внутри" языка ПРОЛОГ. Часто должны быть привлечены другие знания "физического смысла задачи".

Расшифруем правило:

сестра(X, Y):- *женщина* (X), *родитель*(P, X);
родитель(P, Y), $X <> Y$.

Первый шаг связывания переменных — $X = \text{Марья}$. Для сокращения анализируемых при переборе записей высказывания со всеми связанными переменными исключаются. Преобразуемая далее запись принимает вид

родитель($P, \text{Марья}$), *родитель*(P, Y), $\text{Марья} <> Y$

В процедуре *родитель* находим высказывание *родитель*($\text{Елена}, \text{Марья}$) и полагаем $P = \text{Елена}$. Преобразуемая далее запись принимает вид

родитель($\text{Елена}, Y$), $\text{Марья} <> Y$.

Первое по порядку значение $Y = \text{Марья}$ приводит к противоречию. Находим другое высказывание — *родитель*($\text{Елена}, \text{Федор}$). Преобразуем запись:

$\text{Марья} <> \text{Федор}$. Унификация проходит — одна пара "сестра — брат" найдена.

Продолжим поиск других пар. На предыдущем шаге ищем другое подходящее значение P , но его нет. Возвращаемся еще на шаг назад и полагаем $X = \text{Елена}$. Преобразуемая запись принимает вид

родитель($P, \text{Елена}$), *родитель*(P, Y), $\text{Елена} <> Y$.

Получаем противоречие (не проходит унификация), так как в процедуре *родитель* нет информации о родителе Елены.

Пробуем значение $X = \text{Ольга}$. Запись принимает вид

родитель($P, \text{Ольга}$), *родитель*(P, Y), $\text{Ольга} <> Y$.

Вновь не проходит унификация, так как родители Ольги неизвестны.

Пробуем последнее значение $X = \text{Ирина}$. Запись принимает вид

Родитель($P, \text{Ирина}$), *родитель*(P, Y), $\text{Ирина} <> Y$.

Варианты связывания переменной P на данном шаге *backtracking*'а проходят унификацию лишь при $P = \text{Ольга}$. Запись принимает вид

родитель($\text{Ольга}, Y$), $\text{Ирина} <> Y$.

В процедуре *родитель* находим не исследованное ранее высказывание *родитель*($\text{Ольга}, \text{Петр}$) и полагаем $Y = \text{Петр}$. Запись вырождается: $\text{Ирина} <> \text{Петр}$, что вполне справедливо, т. е. унификация проходит.

Продолжив перебор, убеждаемся, что с данным отношением родства существуют только две пары: (Марья, Федор) и (Ирина, Петр).

На основе сделанных выводов развивается фактографическая нейронная сеть, что частично представлено на рис. 2.

Получение новых знаний методом индуктивного мышления

На основе фактов и процедуры *родитель* построим полную систему логического дедуктивного вывода. На генеалогическом дереве это соответствует всем логическим цепочкам, следующим из каждой вершины, кроме концевых:

- 1) *мужчина*(Юрий), *родитель*($\text{Юрий}, \text{Федор}$), *родитель*($\text{Федор}, \text{Ирина}$);
- 2) *мужчина*(Юрий), *родитель*($\text{Юрий}, \text{Федор}$), *родитель*($\text{Федор}, \text{Петр}$), *родитель*($\text{Петр}, \text{Иван}$);
- 3) *мужчина*(Юрий), *родитель*($\text{Юрий}, \text{Федор}$), *родитель*($\text{Федор}, \text{Петр}$), *родитель*($\text{Петр}, \text{Василий}$);
- 4) *женщина*(Елена), *родитель*($\text{Елена}, \text{Федор}$), *родитель*($\text{Федор}, \text{Ирина}$);
- 5) *женщина*(Елена), *родитель*($\text{Елена}, \text{Федор}$), *родитель*($\text{Федор}, \text{Петр}$), *родитель*($\text{Петр}, \text{Иван}$);
- 6) *женщина*(Елена), *родитель*($\text{Елена}, \text{Федор}$), *родитель*($\text{Федор}, \text{Петр}$), *родитель*($\text{Петр}, \text{Василий}$);
- 7) *женщина*(Елена), *родитель*($\text{Елена}, \text{Марья}$), *родитель*($\text{Марья}, \text{Иван}$);
- 8) *женщина*(Елена), *родитель*($\text{Елена}, \text{Марья}$), *родитель*($\text{Марья}, \text{Василий}$);
- 9) *женщина*(Ольга), *родитель*($\text{Ольга}, \text{Ирина}$);
- 10) *женщина*(Ольга), *родитель*($\text{Ольга}, \text{Петр}$), *родитель*($\text{Петр}, \text{Иван}$);
- 11) *женщина*(Ольга), *родитель*($\text{Ольга}, \text{Петр}$), *родитель*($\text{Петр}, \text{Василий}$);
- 12) *мужчина*(Петр), *родитель*($\text{Петр}, \text{Иван}$);
- 13) *мужчина*(Петр), *родитель*($\text{Петр}, \text{Василий}$);
- 14) *женщина*(Марья), *родитель*($\text{Марья}, \text{Иван}$);
- 15) *женщина*(Марья), *родитель*($\text{Марья}, \text{Василий}$).

Первая стратегия анализа логических цепочек.

Анализируя первую цепочку, видим, что она представляет собой некоторую странную конструкцию, не лишенную закономерности. Чтобы представить ее в абстрактной форме, изменим обозначение конкретных имен переменных по принципу: за крайними будем закреплять абстрактные символы X, Y, \dots , а за внутренними — символы P, Q, \dots . Тогда первая цепочка примет вид:

мужчина(X), *родитель*(X, P), *родитель*(P, Y).

Находим, что эта абстрактная конструкция при разных вариантах связывания переменных повторяется в цепочках 2, 3, 5, 6. В 2 и 3 такую конструкцию можно выделять дважды, с пересечением.

Если процедура "мужчина" при этом явно не фигурирует, она подразумевается благодаря первому вхождению мужского имени.

Тогда целесообразно столь жизненно важное отношение между объектами X и Y выделить в отдельное **правило**, придумав ему наименование:

**дедушка(X, Y):- мужчина(X), родитель(X, P),
родитель(P, Y).**

Конечно, с высоты познаний мы с некоторой иронией относимся к формализации нашего мышления. Мы-то знаем названия тех понятий, которые хотим вывести автоматически.

Новым правилом дополняем понятийную логическую нейронную сеть на рис. 3. Решив "целевую" задачу дедуктивного вывода, дополняем фактографическую логическую нейронную сеть, что частично отображено на рис. 2.

Аналогично замечаем, что абстрактная конструкция вида

женщина(X), родитель(X, P), родитель(P, Y),

впервые встретившаяся при анализе цепочки 4, тоже многократно используется в последующих логических цепочках — 5, 6, 7, 8, 10, 11.

Выбрав название (на основе детского лепета), сформулируем новое правило:

**бабушка(X):- женщина(X), родитель(X, P),
родитель(P, Y).**

Однако слишком длинными представляются нам некоторые цепочки даже после замены новыми правилами. Например, цепочка 2 принимает вид

дедушка(Юрий, Петр), родитель(Петр, Иван).

Абстрагируясь и придумав имя новому понятию, получаем новое правило:

прадедушка(X, Y):- дедушка(X, P), родитель(P, Y).

Аналогично получается правило

прабабушка(X, Y):- бабушка(X, P), родитель(P, Y).

На основе новых правил и следующих из них фактов могут быть дополнены фактографическая и понятийная нейронные сети, что частично отражено на рис. 2 и 3.

Выше говорилось, что новые знания остаются гипотезами до достаточно полного подтверждения опытом и дедуктивными рассуждениями.

Предположим, что явился (не запылится) Залетный Молодец, который сообщил Ивану, что он — его дед. Воспользуемся правилом **дедушка** и попробуем решить задачу дедукции:

**дедушка(Залетный Молодец, Иван):-
мужчина(Залетный Молодец), родитель(Залетный Молодец, P), родитель(P, Иван).**

Первое же высказывание в этой цепочке не проходит унификацию, так как в фактах нет соответ-

ствующей регистрации! Если даже будет установлено, что Залетный Молодец — мужчина, то установление праотцовства — весьма болезненный процесс. Так что второе высказывание вряд ли пройдет унификацию, и рекомендовать Ивану проявление прародительской почтительности мы не станем. Но это доказывает весьма высокую действенность выведенного правила!

При появлении следующего поколения можно будет установить истинность и непротиворечивость выведенных выше правил.

Вторая стратегия анализа логических цепочек основана на совместном рассмотрении логических цепочек, являющихся альтернативным продолжением одна другой.

Предположим, что нам неизвестны правила **брат(X, Y)** и **сестра(X, Y)**.

При совместном анализе цепочек 1 и 2 находим, что их различие начинается с высказываний **родитель(Федор, Ирина)** (первая цепочка) и **родитель(Федор, Петр)**. Запишем в абстрактном, обобщенном виде:

**родитель(X, Y)
родитель(X, Z).**

Возникает вопрос: в каком отношении родства находятся Y и Z при общем X , если а) Y — обязательно мужчина, б) Z — обязательно женщина? Запишем проект конструируемых понятий:

**F1(Y, Z):- мужчина(Y), родитель(X, Y),
родитель(X, Z), $Y < > Z$;**

**F2(Z, Y):- женщина(Z), родитель(X, Z),
родитель(X, Y), $Z < > Y$.**

Воспользуемся где-то услышанными красивыми названиями, сменим обозначения для приведения к "стандартному" виду и запишем два новых правила:

**брат(X, Y):- мужчина(X), родитель(P, X),
родитель(P, Y), $X < > Y$;**

**сестра(X, Y):- женщина(X), родитель(P, X),
родитель(P, Y), $X < > Y$.**

Новые правила отражаются в БЗ и подтверждаются при их применении в задачах достижения цели.

На основе данного множества фактов, осуществляя стратегию совместного анализа цепочек логического вывода, можно высказать гипотезу о новых понятиях **племянник** и **племянница**:

**племянник(X, Y):- мужчина(X), родитель(P, X),
сестра(P, Y);**

**племянница(X, Y):- женщина(X), родитель(P, X),
брат(P, Y).**

Дальнейшее обобщение на базе существующих фактов невозможно.

Действительно, мы знаем, что Ирина — племянница Марьи, а Иван и Василий — племянники Федора по материнской линии. (По отцовской линии они внуки Федора.)

Следует надеяться, что с появлением новых поколений станет возможно дальнейшее обобщение, которое способно привести к неоднозначности и противоречивости выводов. Например, в дополнение к прежнему может быть сформулирован вывод

племянник(X, Y):- *мужчина*(X), *родитель*(P, X),
брат(P, Y).

Для устранения возникшего противоречия в определении *племянник* следует расширить использование в ПРОЛОГе логических операций *И* и *ИЛИ* (в "классическом" ПРОЛОГе ограниченно используется символ ; как знак дизъюнкции, а скобки в предложениях не применяются). Ведь под запятой, разделяющей высказывания в одном предложении, подразумевается конъюнкция *И*, а если использовать в предложениях символ дизъюнкции *ИЛИ*, единственное непротиворечивое правило примет вид:

племянник(X, Y):- *мужчина*(X), *родитель*(P, X),
(*брат*(P, Y) *ИЛИ* *сестра*(P, Y)).

Аналогично

племянница(X, Y):- *женщина*(X), *родитель*(P, X),
(*брат*(P, Y) *ИЛИ* *сестра*(P, Y)).

Заключение

Язык логического программирования ПРОЛОГ в основном направлен на построение дедуктивных цепочек вывода. Для этого используются встроенные программные процедуры. Однако перспективы

использования языка этим не ограничены. Можно воспроизводить принципы индуктивного мышления человека, создавать и встраивать процедуры для формирования новых знаний. Такая программная надстройка языка анализирует записи, выделяет, обобщает и именуется часто встречающиеся конструкции, дополняет ими базу знаний. Задача анализа множества логических цепочек обладает высокой сложностью, так как ее решение основано на переборе. Это иллюстрирует гигантские усилия ученого-исследователя по обнаружению и обоснованию, казалось бы, незначительной зависимости между явлениями природы.

Список литературы

1. Амосов М. Н., Байдык Т. Н., Гольцов А. Д. и др. *Нейрокомпьютеры и интеллектуальные роботы*. Киев: Наукова думка, 1991. 269 с.
2. Пospelov Д. А. *Моделирование рассуждений: Опыт анализа мыслительных актов*. М.: Радио и связь, 1989. 184 с.
3. Пospelov Д. А. Десять "горячих точек" в исследованиях по искусственному интеллекту // *Интеллектуальные системы (МГУ)*. Т. 1. Вып. 1—4. 1996.
4. Барский А. Б. *Нейронные сети: распознавание, управление, принятие решений*. М.: Финансы и статистика, 2004. 175 с.
5. Барский А. Б. *Нейронные сети логического вывода*. Курс лекций. Saarbrücken: LAP LAMBERT Academic Publishing, 2011. 400 с.
6. Барский А. Б. *Математическая логика событий и логические нейронные сети // Информационные технологии*. Приложение. 2007. № 7. 32 с.
7. Барский А. Б. *Нейросетевые технологии обработки информации: учеб. пособие*. СПб: ИЦ "Интермедия". 2016. 322 с.
8. Братко И. *Программирование на языке ПРОЛОГ для искусственного интеллекта*. М.: Мир. 1990. 559 с.

A. V. Barsky, Professor, e-mail: arkbarsk@mail.ru, Moscow State University of Railway Engineering (MIIT)

Simulation of Inductive Reasoning Using Inference PROLOG

In accordance with the two paradigms of artificial intelligence — the expert and student — studied the possibility of constructing models of deductive and inductive reasoning based on human inference PROLOG language. If the simulation of deductive reasoning studied enough and is the main goal of the language, the model of inductive reasoning, that is, the formation of new knowledge, offered for the first time. Its essence is as follows. According to the knowledge base consisting of facts and rules, all possible deductive inference chain built. They are detected and distinguished finished, preferably repetitive structure. Related variables are replaced by their abstract representation for the formation of a generalized type of the selected design. This creates a hypothesis about the description of the new concepts. These hypothetical concepts are named and they are the new rules that complement the knowledge base. At the same time the description of the new regulations complement the conceptual logical neural network to be able to work with fuzzy data. Practice successful, a consistent application of the new rules should confirm their high reliability.

Keywords: artificial intelligence paradigm, deductive and inductive reasoning, logic neural network, PROLOG, the logical chain

References

1. Amosov M. N., Bajdyk T. N., Gol'cov A. D. i dr. *Nejrokomп'yutery i intellektual'nye roboty*, Kiev: Naukova dumka, 1991, 269 p.
2. Pospelov D. A. *Modelirovanie rassuzhdenij: Opyt analiza myslitel'nyh aktov*, Moscow: Radio i svyaz', 1989, 184 p.
3. Pospelov D. A. Desyat "goryachih toчек" v issledovaniyah po iskusstvennomu intellektu, *Intellektual'nye sistemy (MGU)*, 1996, vol. 1, Вып. 1—4, 1996.
4. Barskij A. B. *Nejronnye seti: raspoznavanie, upravlenie, prinyatie reshenij*, Moscow: Finansy i statistika, 2004, 175 p.
5. Barskij A. B. *Nejronnye seti logicheskogo vvoda. Kurs lekcij*. Saarbrücken: LAP LAMBERT Academic Publishing, 2011, 400 p.
6. Barskij A. B. *Matematicheskaya logika sobytij i logicheskie nejronnye seti, Informacionnye tehnologii. Prilozhenie*, 2007, no. 7. 32 p.
7. Barskij A. B. *Nejrosetevye tehnologii obrabotki informacii: ucheb. posobie*, SPb.: IC "Intermediya", 2016, 322 p.
8. Bratko I. *Programmirovaniye na yazyke PROLOG dlya iskusstvennogo intellekta*, Moscow: Mir, 1990, 559 p.

В. А. Маренко, канд. техн. наук, доцент, ст. науч. сотр., marenko@ofim.oscsbras.ru,
Институт математики им. С. Л. Соболева СО РАН, г. Омск

Применение когнитивного моделирования для исследования межэтнических отношений

Цель работы — исследование межэтнических отношений в городской молодежной среде. Реализованы следующие задачи: краткий анализ научной информации по теме исследования, анкетирование в студенческой среде и применение методики когнитивного моделирования.

На начальном этапе исследования формализуется проблемное поле с помощью взвешенного ориентированного графа. Далее описаны результаты вычислительного эксперимента, реализованного с применением численных методов.

Ключевые слова: когнитивная карта, граф, когнитивная модель, вычислительный эксперимент, лингвистическая переменная, межэтническая интеграция

Введение

Россия — многонациональное государство, в котором развитие общественных отношений осуществляется в соответствии со Стратегией государственной национальной политики на период до 2025 года [1]. Основными условиями существования нации, этноса является общность языка, единство компонентов духовной культуры и связанные с ними социально-психологические характеристики сознания и поведения людей.

Президент РФ В. В. Путин на заседании Совета по межнациональным отношениям рассуждал о негативных тенденциях в сфере этносоциальных отношений. По его словам межнациональные конфликты могут привести к разрушению государственности [2]. Президент Казахстана Н. А. Назарбаев, выступая на Ассамблее народов Казахстана, сказал, что, несмотря на этническое, культурное и религиозное многообразие, в стране сохраняется мир и политическая стабильность [3]. Генеральный секретарь ООН Пан Ги Мун на встрече с лидерами государств отмечал, что "слишком много стран в мире разделены по этническому признаку, задерживаются в развитии из-за подозрений и несогласия между нациями".

Оценки ситуации по вопросу мировой межнациональной политики признанными политическими лидерами, а также сложные миграционные процессы в настоящее время являются подтверждением актуальности рассматриваемой темы.

Цель работы — исследование межэтнических отношений в городской молодежной среде.

Для достижения цели сформулированы следующие задачи: краткий обзор научной информации по теме исследования; проведение анкетирования в студенческой среде для получения первичной информации и применение методики когнитивного моделирования для исследования межэтнических отношений.

Большое внимание ученые уделяют изучению межэтнических вопросов. З. Я. Умарова отмечает, что

толерантное сознание как основа духовного мира человека является высшей ценностью общественного развития и межнациональных отношений [4]. А. В. Андреев обосновывает необходимость реализации форм, методов и средств совершенствования воспитания доброты и гуманности в сфере межличностных и межнациональных отношений в поликультурном образовательном пространстве [5]. М. Б. Беджанов рассматривает проблемы поиска новых путей формирования межнациональных отношений, развития их культуры и национального самосознания в новых исторических условиях [6]. Н. В. Круглова пишет, что созданию благоприятного межнационального климата способствует формирование эффективных механизмов для сохранения и развития национальных культур, обычаев и традиций [7]. И. С. Лысенко рассматривает управление этносоциальными отношениями, которое в соответствии с диалектической методологией направлено на устранение социальной энтропии и совершенствование общественных отношений [8]. Д. А. Мункожаргалов считает, что цель формирования этнической толерантности — объединение общества, достижение межнационального согласия политическими средствами [9]. Р. Д. Флорес (США) на основе этнографических данных установил, что этническая вражда между белым коренным населением и латиноамериканцами продолжает расти вследствие неблагоприятной политики государственных органов управления [10]. О. S. McDoom (Англия) разработал теоретическую модель с причинно-следственным механизмом, которая показывает, что во время геноцида насилие быстрее начинается там, где межнациональные связи слабые [11]. Результаты исследований Е. Естевез (Испания) и др. показали, что этнические меньшинства более негативно относятся к правовой системе по сравнению с большинством населения [12]. Исследование С. Десбиенс (Канада) посвящено выяснению результатов межкультурного диалога между некоренными и коренными народами, которым в настоящее время

отводится активная роль в вопросах поликультурного территориального развития на основе кросс-культурного партнерства и инвестиций [13]. Р. Купманс (Германия) показал, что страны со строгой политикой интеграции или с недостаточным всеобщим благосостоянием добиваются лучших результатов интеграции [14]. В исследовании А. Зорли (Нидерланды) использован метод математической статистики, который показал наличие неравного доступа к социальной помощи иммигрантов с межэтническими различиями [15].

Для исследования состояния межэтнических отношений в многонациональной среде авторы статьи предлагают использовать когнитивное моделирование, методика которого адаптирована к современным условиям в Институте проблем управления РАН и апробирована авторами в исследовании социально-психологического феномена "когнитивный диссонанс" личности в молодежной среде Омского региона РФ [16, 17].

Когнитивное моделирование проблемы "межэтническая интеграция"

На начальном этапе исследования формируется проблемное поле из базисных факторов, влияющих на проблему. Формализация проблемного поля осуществляется в виде субъективной схемы — когнитивной карты, представляемой взвешенным ориентированным графом. Вершинами графа являются базисные факторы, а дуги отражают взаимовлияние факторов и имеют направления и "веса", назначаемые экспертами. Согласование весов и дуг осуществляется с применением математической статистики. В результате проведенных процедур проблема фиксируется как динамическая ситуация [18].

Далее проводится вычислительный эксперимент с использованием когнитивной матрицы, необходимой для построения алгоритма вычислений. Вычислительный эксперимент выявляет наиболее существенные факторы, влияющие на проблему, а также может быть использован для осуществления прогноза развития ситуации.

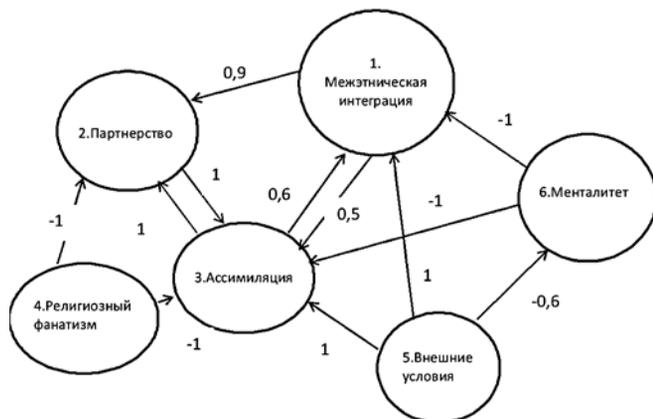


Рис. 1. Когнитивная карта "межэтническая интеграция"

Целевым фактором исследования является "межэтническая интеграция" (1). "Партнерство" (2), "ассимиляция" (3), "религиозный фанатизм" (4), "менталитет" (6) и "внешние условия" (5) определены экспертами как управляющие факторы. Формализация проблемного поля осуществляется взвешенным ориентированным графом $G = \langle V, E \rangle$, где V — множество вершин, $V_i \in V, i = 1, 2, \dots, k$; E — множество дуг, $e_{ij} \in E, i, j = 1, 2, \dots, n$ (рис. 1).

Экспертный опрос выявил следующие рассуждения. "Усиление религиозного фанатизма ослабляет процессы ассимиляции, партнерства и межэтнической интеграции", "улучшение внешних условий способствует процессу межэтнической интеграции" и т.д. Экспертные рассуждения отражаются на когнитивной карте с помощью прямой или обратно пропорциональной зависимости.

Когнитивная модель проблемы формируется в следующем виде: $\Phi = (G, X, F)$, где $G = \langle V, E \rangle$ — ориентированный граф; X — множество параметров вершин $V, X = \{X^{(v_i)}\}, i = 1, 2, \dots, k; X^{(v_i)} = \{x_g^{(i)}\}, g = 1, 2, \dots, n; x_g^{(i)}$ — параметр вершины V_i , если $g = 1$, то $x_g^{(i)} = x_i; X:V \rightarrow R, R$ — множество вещественных чисел; $F = F(X, E) = F(x_i, x_j, e_{ij})$ — функционал преобразования дуг, ставящий в соответствие каждой дуге знак, весовой коэффициент ω_{ij} или функцию $f(x_i, x_j, e_{ij}) = f_{ij}$.

Результаты вычислительного эксперимента

Алгоритм вычисления строится с применением численных методов [19]. Суть вычисления следующая. В одну или несколько вершин графа вносятся возмущения, и наблюдается распространение "волны возмущений" по различным путям в графе. Если в момент времени t в вершину поступало возмущение $p_j \in P$, то переход исследуемой системы из состояния t в $t + 1$ осуществляется по правилу:

$$x_i(t + 1) = x_i(t) + \sum_{j=1}^k f(x_i, x_j, e_{ij})p_j(t), \text{ при известных}$$

начальных значениях. Визуализация численных значений базисных факторов выполнена на нескольких шагах вычислений.

Первые расчеты показали наличие нежелательных явлений: линейного и экспоненциального резонансов (рис. 2, а, б). Эти явления возникли вследствие ошибочно указанных экспертами взаимосвязей между базисными факторами. Для устранения ошибок теории советуют осуществлять структурную перестройку графа [20]. Граф необходимо формировать в виде "розы" с "мостом" в одной вершине. Лепестки "розы" должны повесить устойчивость и уравновесить сформированную структуру.

На рис. 2, а, б ось абсцисс — шаги вычисления, ось ординат — значения базисных факторов в условных единицах. Цифры от единицы до шести соответствуют номерам базисных факторов, указанных на когнитивной карте.

Преобразование взаимосвязей между факторами привело к устойчивой структуре. На рис. 3 представлен результат вычислений при усилении факторов "партнерство" (2), "ассимиляция" (3), "религиозный фанатизм" (4) на одну условную единицу и фактора "внешние условия" (5) на две условные единицы. При таком условии целевой фактор "межэтническая интеграция" (1) достигает значения 3 условных единицы.

При усилении фактора "религиозный фанатизм" (4) до двух условных единиц целевой фактор ослабляется почти в 3 раза (рис. 4).

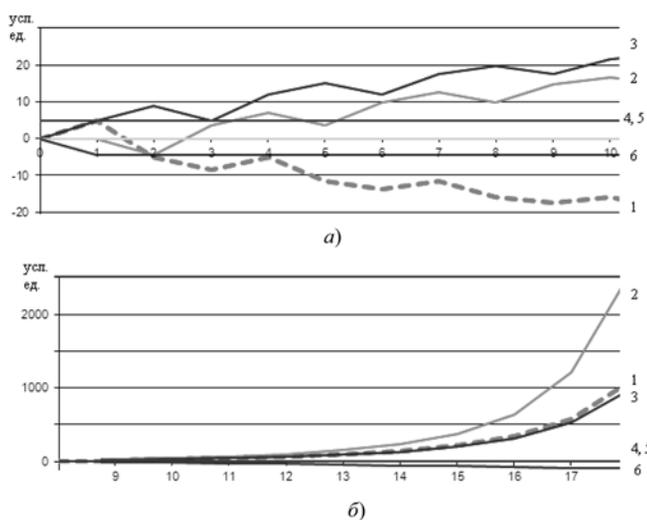


Рис. 2. Фрагменты расчетов: а — линейный резонанс; б — экспоненциальный резонанс

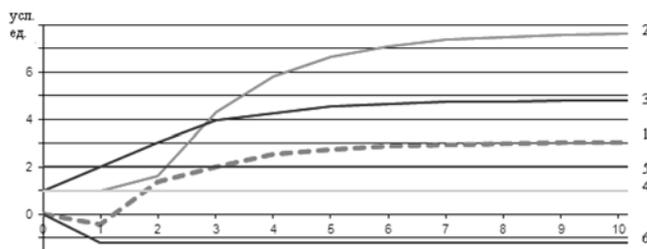


Рис. 3. Результаты расчета при усилении факторов "партнерство", "ассимиляция", "религиозный фанатизм" и "внешние условия"

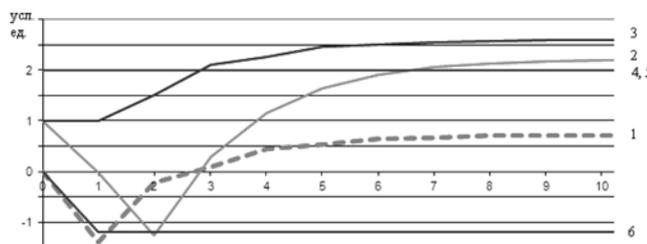


Рис. 4. Результаты расчета при изменении фактора "религиозный фанатизм"

Полученные результаты согласуются со здравым смыслом: чем меньше "религиозный фанатизм", тем выше "межэтническая интеграция" в многонациональном обществе.

На следующем этапе вычислительного эксперимента будут учитываться не только веса и направления дуг, но и значения переменных в вершинах графа. Переменные могут быть различными, в том числе, лингвистическими.

Лингвистическая переменная "ассимиляция"

Понятие лингвистической переменной играет важную роль в принятии решений на основе приближенных рассуждений. Нечетким логическим выводом в нашем исследовании является определение четкого значения выходной переменной "ассимиляция" по входной переменной "когнитивный диссонанс" с применением логических правил, формируемых экспертами.

Лингвистическая переменная "ассимиляция" характеризует процессы, в результате которых члены одной этнической группы усваивают в определенной степени культуру другой этнической группы. Ассимиляция тесно связана с понятием "когнитивный диссонанс", который отражает состояние психического дискомфорта личности [21]. Такое состояние может возникнуть на фоне религиозного, морального и эмоционального несоответствия, например, при наблюдении незнакомого человеку традиций и обычаев других народов. Термами лингвистической переменной "когнитивный диссонанс" могут быть слова "высокий", "средний", "низкий". Ассимиляция может иметь термы "сильная", "средняя", "слабая" и т.д. Между "когнитивным диссонансом" личности и "ассимиляцией" существует взаимосвязь в виде следующих экспертных правил. "Если когнитивный диссонанс высокий, то ассимиляция слабая"; "если когнитивный диссонанс низкий, то ассимиляция сильная" и т.д.

Для получения численных значений лингвистических переменных проведено анкетирование студентов омских вузов. Фрагмент одной из анкет представлен в табл. 1.

В результате анкетирования получены интегральные оценки процесса ассимиляции и состояния личности, фиксируемого термином "когнитивный"

Таблица 1

Фрагмент анкеты "Ассимиляция"

3	Считаете ли Вы, что ваши права ущемляются по национальной принадлежности в повседневной жизни?	Да	Нет	Затрудняюсь ответить
4	Считаете ли Вы, что ваши права ущемляются по национальной принадлежности государственными органами?	Да	Нет	Затрудняюсь ответить

Таблица 2

Значения лингвистических переменных

Входная переменная	Баллы	Выходная переменная	Баллы
Высокий когнитивный диссонанс	[9; 15]	Слабая ассимиляция	[5; 10]
Средний когнитивный диссонанс	[15; 20]	Средняя ассимиляция	[10; 13]
Низкий когнитивный диссонанс	[20; 27]	Сильная ассимиляция	[13; 16]

диссонанс" (табл. 2). Если респонденты ставили низкие баллы за свое материальное положение, жизненные перспективы и т.д., то когнитивный диссонанс у таких респондентов высокий и ассимиляция слабая.

Вычисление степеней принадлежности элементов соответствующему нечеткому множеству осуществлялось с помощью следующих процедур. Эксперту предлагалось сравнить элементы в каждом из нечетких множеств, представленных значениями лингвистических переменных (табл. 2) с помощью схемы "сравнение с образцом" [22]. В роли образца выступал первый элемент рассматриваемого нечеткого множества. Далее с помощью качественных оценок шкалы Саати устанавливали его сходство с другими элементами [23].

Рассмотрим лингвистическую переменную (нечеткое множество) "высокий когнитивный диссонанс", в котором за образец принимали первый элемент. Эксперты установили слабое преимущество между первым и вторым элементами и промежуточное (слабое — существенное) преимущество между первым и третьим элементами. Затем с использованием количественных оценок шкалы Саати формировали первую строку матрицы парных сравнений нечеткого множества "высокий когнитивный диссонанс". Остальные элементы определены

с учетом свойств матрицы по формуле $a_{ij} = \frac{a_{1j}}{a_{1i}}$, $i = 2, \dots, n; j = 1, 2, \dots, n$. Элементы матрицы следующие:

$$M = [a_{ij}]_{n \times n} = \begin{bmatrix} 1 & \frac{1}{3} & \frac{1}{5} \\ 3 & 1 & \frac{3}{5} \\ 5 & \frac{5}{3} & 1 \end{bmatrix}.$$

Так как матрица парных сравнений M диагональная, обратна симметричная, транзитивная, то степени принадлежности элементов соответствующему

нечеткому множеству вычисляют [24] по формуле

$$\mu(s_i) = \frac{1}{a_{1i}^l + a_{2i}^l + \dots + a_{ni}^l},$$

$$i = 1, 2, \dots, n; l = 1, 2, \dots, m.$$

Аналогично проводят расчеты элементов матрицы и степеней принадлежности элементов для значений лингвистических переменных "сильная ассимиляция", "слабая ассимиляция" и др.

Результаты вычислений для значений лингвистических переменных (нечетких множеств) после нормирования приведены в табл. 3.

Визуализация расчетов представлена на рис. 5, а, б.

Используя нечеткий вывод и экспертные правила, определяем четкое значение искомой пере-

Таблица 3
Нормированные значения степеней принадлежности элементов соответствующим лингвистическим переменным

Высокий когнитивный диссонанс		Слабая ассимиляция	
Значение элемента	Степень принадлежности	Значение элемента	Степень принадлежности
9	0,2	5	1
13	0,93	7	0,92
15	1	10	0,19

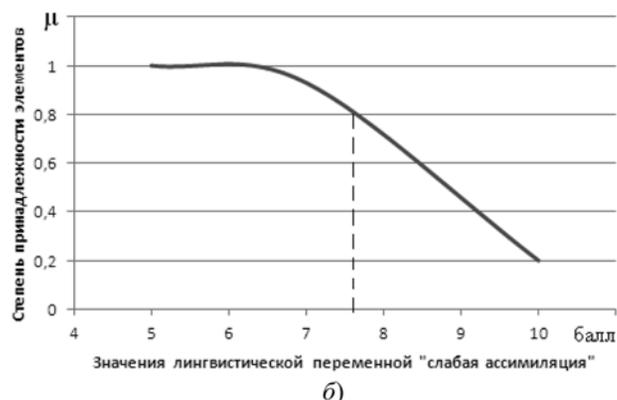
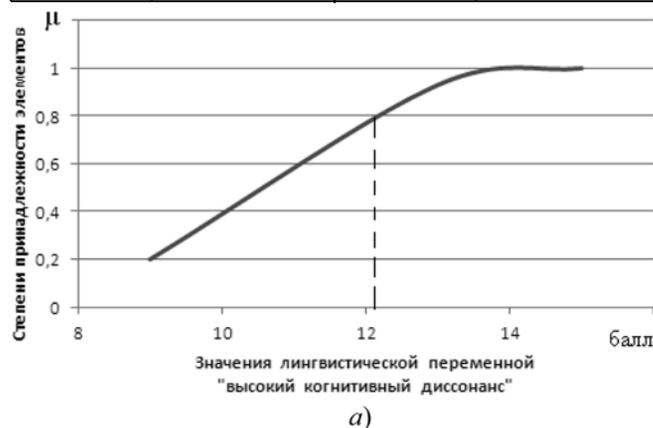


Рис. 5 Лингвистические переменные:

а — высокий когнитивный диссонанс; б — слабая ассимиляция

менной. Если (12,2; 0,8) — элемент лингвистической переменной "высокий когнитивный диссонанс" (рис. 4, а), то четкое значение элемента лингвистической переменной "слабая ассимиляция" — 7,6 баллов (рис. 4, б). Такая процедура будет использоваться для получения четких значений всех управляющих факторов на планируемом этапе вычислительного эксперимента. А также, построив экспериментальные кривые для соответствующей страты общества, можно по значениям когнитивного диссонанса оперативно определять степень ассимиляции, как отдельного индивида, так и межэтнической группы в целом.

Заключение

Исследование показало, что когнитивное моделирование можно применять для изучения различных процессов в социальной сфере. Когнитивная методология может быть использована для выявления факторов, оказывающих на социальные отношения большее или меньшее управляющее воздействие. В результате может быть сформирована система научно обоснованных рекомендаций, направленная на поддержку принятия управленческих решений для государственных структур по предотвращению возможно неблагоприятного развития проблемы в будущем.

Когнитивное моделирование можно применять в образовательном процессе вуза для повышения мотивации к изучению математики, так как для верификации сформированной когнитивной модели и проведения вычислительного эксперимента студентам необходимы знания теории множеств, теории графов, численных методов, имитационного моделирования и других разделов математики.

Список литературы

1. Указ Президента РФ от 19.12.2012 № 1666 "О Стратегии государственной национальной политики Российской Федерации на период до 2025 года" (19 декабря 2012 г.) / КонсультантПлюс. 1992—2014. URL: http://www.consultant.ru/document/cons_doc_LAW_139350/.
2. Путин В. В. Выступление на заседании Совета по межнациональным отношениям // Вестник российской нации. 2014. Т. 4, № 4 (36). С. 8—10.
3. Назарбаев Н. А. Казахстан-2050 // Акorda. 2012.
4. Умарова З. Я. Проблема толерантного сознания в межнациональных отношениях // Теория и практика общественно-го развития. 2010. № 2. С. 57—64.
5. Андреев А. В. Воспитание доброты и гуманности студентов в межличностных и межнациональных отношениях в поликультурном пространстве Поволжья // Вестник Чувашского университета. 2010. № 1. С. 140—143.
6. Беджанов М. Б. Этнические образования и межнациональные отношения // Вестник Адыгейского государственного университета. Серия 1: Регионоведение: философия, история, социология, юриспруденция, политология, культурология. 2012. № 1. С. 126—130.
7. Круглова Н. В. Совершенствование управления межнациональными отношениями в регионах России на примере Северной Осетии // Государственное и муниципальное управление в XXI веке: теория, методология, практика. 2014. № 14. С. 8—13.
8. Лысенко И. С. Управление межэтническими и межнациональными отношениями. Социально-философский анализ // Перспективы науки. 2011. № 6 (21). С. 88—91.
9. Мункожаргалов Д. А. Роль этнической толерантности в межнациональных отношениях // Вестник Бурятского государственного университета. 2014. № 14: 1. С. 95—98.
10. Flores R. D. Living in the Eye of the Storm: How did Hazleton's Restrictive Immigration Ordinance Affect Local Interethnic Relations? // American Behavioral Scientist. 2014. Vol. 58, Is. 13. P. 1743—1763.
11. McDoom O. S. Predicting violence within genocide: A model of elite competition and ethnic segregation from Rwanda. Political Geography. 2014. Vol. 42. P. 34—45.
12. Estévez E., Rachitskiy M., Rodríguez C. Is perception of the mainstream legal system homogeneous across ethnic groups? // European Journal of Psychology Applied to Legal Context. Vol. 5, Is. 2. 2013. P. 155—161.
13. Desbiens C., Rivard É. From passive to active dialogue? Aboriginal lands, development and métissage in Québec, Canada. Cultural Geographies. 2014. Vol. 21. Is. 1. P. 99—114.
14. Koopmans R. Trade-Offs between equality and difference: Immigrant integration, multiculturalism and the welfare state in cross-national perspective // Journal of Ethnic and Migration Studies. Vol. 36. Is. 1. 2010. P. 1—26.
15. Zorlu A. Welfare use of migrants in the Netherlands. International Journal of Manpower. 2013. Vol. 34. Is. 1. P. 83—95.
16. Максимов В. И., Тер-Егизарова Н. В. Когнитивный анализ и управление развитием ситуаций // IV Международная конференция CASC'2004 // Проблемы управления. 2005. № 1. С. 83—87.
17. Маренко В. А., Лучко О. Н. Моделирование когнитивного диссонанса личности как состояния социальной напряженности // Известия Иркутской государственной экономической академии. 2015. Т. 25, № 1. С. 149—159.
18. Кузнецов О. П., Кулинич А. А., Марковский А. В. Анализ влияний при управлении слабоструктурированными ситуациями на основе когнитивных карт // Человеческий фактор в управлении. Сборник. М.: КомКнига, 2006. С. 313—344.
19. Лучко О. Н., Маренко В. А., Лупенцов О. С. Программная система для изучения эффективности образовательного процесса в вузе с использованием когнитивных технологий / Свидетельство о государственной регистрации программ для ЭВМ № 2013661212 от 02.12.2013 г.
20. Кононов Д. А., Косяченко С. А., Кульба В. В. Формирование и анализ сценариев развития социально-экономических систем с использованием аппарата операторных графов // Автоматика и телемеханика. 2007. № 1. С. 121—136.
21. Фестингер Л. Теория когнитивного диссонанса. СПб.: Ювента, 1999. 317 с.
22. Saaty T. L. Decision-making. Method of the analysis of hierarchies. M.: Radio and communication, 1989. 316 p.
23. Ногин В. Д. Упрощенный вариант метода анализа иерархий на основе нелинейной свертки критериев // Вычислительная математика и математическая физика. 2004. Т. 44, № 7. С. 1261—1270.
24. Ротштейн А. П., Штовба С. Д. Нечеткий многокритериальный анализ вариантов с применением парных сравнений // Известия академии наук. Теория и системы управления. 2001. № 3. С. 150—154.

Cognitive Modelling for the Interethnic Relations Study

The work purpose — research of the interethnic relations in youth environment. Research problems: the short analysis of scientific information, questioning in the student's environment and application of a technique of cognitive modeling.

At the initial stage of research the problem field is formed of basic factors: interethnic integration, partnership, religion, assimilation, mentality and external conditions. Formalization of a problem field is carried out in the form of the weighed focused count. Tops of the count are basic factors. Arches have the direction and "weight", appointed by experts. Coordination of scales of arches is carried out with application of mathematical statistics.

The algorithm of computing experiment is constructed with application of numerical methods. Calculation essence the following. Indignations are brought in tops of the count. Distribution of "a wave of indignations" on various ways in the column is observed. The first calculations showed the undesirable phenomena: linear and exponential resonance. They arose because of mistakes of experts. Theorists advise to carry out restructuring of the count in the form of "rose". Petals of "rose" increase stability and counterbalance structure of the count. The recommended transformations led to steady structure. Computing experiment showed that strengthening of religion weakens processes of assimilation, partnership and interethnic integration. Improvement of external conditions promotes process of interethnic integration.

It is planned to be considered not only the weight of arches, but also value of variables in the count's tops. Variables can be linguistic. They play an important role in decision-making on the basis of approximate reasonings. The example of determination of value of an output variable on an entrance variable is given.

Keywords: cognitive map, count, cognitive model, linguistic variable, interethnic integration

References

1. **Ukaz** Prezidenta RF ot 19.12.2012 N 1666 "O Strategii gosudarstvennoj nacional'noj politiki Rossijskoj Federacii na period do 2025 goda" ("On the Strategy of the state national policy of the Russian Federation for the period till 2025") (19 dekabrja 2012 g.) / Konsul'tantPljus. 1992—2014. URL: http://www.consultant.ru/document/cons_doc_LAW_139350/.
2. **Putin V. V.** Vstuplenie na zasedanii sovetu po mezhnacional'nym otnoshenijam. ("Speech at the meeting of the Council for Interethnic Relations"), *Vestnik rossijskoj nacii*. 2014, vol. 4, no. 4 (36), pp. 8—10.
3. **Nazarbaev N. A.** Kazakhstan-2050, *Akorda*, 2012.
4. **Umarova Z. Ja.** Problema tolerantogo soznaniya v mezhnacional'nyh otnoshenijah ("The problem of tolerance in international relations"), *Teorija i praktika obshhestvennogo razvitiya*, 2010, no. 2, pp. 57—64.
5. **Andreev A. V.** Vospitanie dobroty i gumannosti studentov v mezhlichnostnyh i mezhnacional'nyh otnoshenijah v polikul'turnom prostranstve Povolzh'ja ("Education of kindness and humanity students interpersonal and interethnic relations in the multicultural environment of the Volga region"), *Vestnik Chuvashskogo universiteta*, 2010, no. 1, pp. 140—143.
6. **Bedzhanov M. B.** Jetnicheskie obrazovaniya i mezhnacional'nye otnosheniya ("Ethnic Education and Ethnic Relations"), *Vestnik Adygejskogo gosudarstvennogo universiteta. Serija 1: Regionovedenie: filosofija, istorija, sociologija, jurisprudencija, politologija, kurturologija*. 2012, no. 1, pp. 126—130.
7. **Kruglova N. V.** Sovershenstvovanie upravlenija mezhnacional'nymi otnoshenijami v regionah Rossii na primere Severnoj Osetii ("Improvement of interethnic relations in the regions of Russia on an example of North Ossetia"), *Gosudarstvennoe i municipal'noe upravlenie v XXI veke: teorija, metodologija, praktika*, 2014, no. 14, pp. 8—13.
8. **Lysenko I. S.** Upravlenie mezhetniceskimi i mezhnacional'nymi otnoshenijami. Social'no-filosofskij analiz ("Managing interethnic and international relations. Socially-philosophical analysis"), *Perspektivy nauki*, 2011, no. 6 (21), pp. 88—91.
9. **Munkozhargalov D. A.** Rol' jetniceskoj tolerantnosti v mezhnacional'nyh otnoshenijah ("The role of ethnic tolerance in international relations"), *Vestnik Burjatskogo gosudarstvennogo universiteta*, 2014, no. 14: 1, pp. 95—98.
10. **Flores R. D.** Living in the Eye of the Storm: How did Hazleton's Restrictive Immigration Ordinance Affect Local Interethnic Relations? *American Behavioral Scientist*, 2014, vol. 58. Is. 13, pp. 1743—1763.
11. **McDoom O. S.** Predicting violence within genocide: A model of elite competition and ethnic segregation from Rwanda, *Political Geography*, 2014, vol. 42, pp. 34—45.
12. **Estévez E., Rachitskiy M., Rodriguez C.** Is perception of the mainstream legal system homogeneous across ethnic groups? *European Journal of Psychology Applied to Legal Context*, 2013, vol. 5, is. 2, pp. 155—161.
13. **Desbiens C., Rivard É.** From passive to active dialogue? Aboriginal lands, development and mÿtissage in Québec, Canada, *Cultural Geographies*, 2014, vol. 21, is. 1, pp. 99—114.
14. **Koopmans R.** Trade-Offs between equality and difference: Immigrant integration, multiculturalism and the welfare state in cross-national perspective, *Journal of Ethnic and Migration Studies*, 2010, vol. 36. Is. 1, pp. 1—26.
15. **Zorlu A.** Welfare use of migrants in The Netherlands, *International Journal of Manpower*, 2013, vol. 34, is. 1, pp. 83—95.
16. **Maksimov V. I., Ter-Egiazarova N. V.** Kognitivnyj analiz i upravlenie razvitiem situacij ("Cognitive analysis and development management situations"), IV Mezhdunarodnaja konferencija CASC'2004, *Problemy upravlenija*, 2005, no. 1, pp. 83—87.
17. **Marenko V. A., Luchko O. N.** Modelirovanie kognitivnogo dissonansa lichnosti kak sostojanija social'noj naprjazhennosti ("Modeling of cognitive dissonance personality as a condition of social tension"), *Izvestija Irkutskoj gosudarstvennoj jekonomicheskoj akademii*, 2015, vol. 25, no. 1, pp. 149—159.
18. **Kuznecov O. P., Kulinich A. A., Markovskij A. V.** Analiz vlijanij pri upravlenii slabostrukturirovannyimi situacijami na osnove kognitivnyh kart ("Analysis of the effects of the management of semi-structured situations based on cognitive maps"), *Sb. Chelovecheskij faktor v upravlenii*, Moscow: KomKniga, 2006, pp. 313—344.
19. **Luchko O. N., Marenko V. A., Lupencov O. S.** *Programmaja sistema dlja izuchenija jeffektivnosti obrazovatel'nogo processa v vuze s ispol'zovanijem kognitivnyh tehnologij* ("The software system for the study of the effectiveness of the educational process in high school with cognitive technology") / Svidetel'stvo o gosudarstvennoj registracii programm dlja JeVM № 2013661212 ot 02.12.2013 g.
20. **Kononov D. A., Kosjachenko S. A., Kul'ba V. V.** Formirovanie i analiz scenarijev razvitiya social'no-jekonomiceskih sistem s ispol'zovanijem apparata operatornyh grafov ("Formation and analysis of scenarios of socio-economic systems with the use of the machine operator graph"), *Avtomatika i telemekhanika*, 2007, no. 1, pp. 121—136.
21. **Festinger L.** *Teorija kognitivnogo dissonansa*, Saint-Petersburg, Juventa, 1999, 317 p.
22. **Saaty T. L.** *Decision-making. Method of the analysis of hierarchies*, Moscow, Radio and communication, 1989, 316 p.
23. **Nogin V. D.** Uproshhennyj variant metoda analiza ierarhij na osnove nelinejnoj svertki kriteriev ("A simplified version of the analytic hierarchy process based on non-linear convolution of criteria"), *Vychislitel'naja matematika i matematicheskaja fizika*, 2004, vol. 44, no. 7, pp. 1261—1270.
24. **Rotshtejn A. P., Shtovba S. D.** Nechetkij mnogokriterial'nyj analiz variantov s primenenijem parnyh sravnenij ("Fuzzy multi-criteria analysis of options for using pairwise comparisons"), *Izvestija akademii nauk. Teorija i sistemy upravlenija*, 2001, no. 3, pp. 150—154.

МОДЕЛИРОВАНИЕ И ОПТИМИЗАЦИЯ MODELING AND OPTIMIZATION

УДК 519.6

Н. Н. Светушков, канд. техн. наук, доц., e-mail: svt.n.n@mail.ru,
Московский авиационный институт (национальный исследовательский университет)

Упрощенная математическая модель расчета температурных полей при детонационном горении

Исследуются возможности расчета тепловых нагрузок на стенки камеры сгорания при детонационном горении, что актуально при разработке промышленных детонационных двигателей, в том числе и при спиновом детонационном горении. Для расчета была предложена упрощенная математическая модель, основанная на параболических уравнениях в частных производных, отражающая основные особенности распространения детонационной волны в камере сгорания. Подробно представлен вывод этого уравнения и указаны основные допущения, использованные при его получении. Для решения полученного уравнения и проведения численных расчетов использовался новый подход — метод струн, основанный на интегральном представлении уравнения теплопроводности. Показаны преимущества предлагаемого подхода и отмечено, что его использование позволяет избежать "нефизических" осцилляций в численном решении в случае больших градиентов температур. В целях проведения расчетов была модифицирована разработанная ранее программная среда, позволяющая создавать двумерные модели камер сгорания с учетом их конфигурации и задавать начальные и граничные условия. Представленные результаты показали, что путем варьирования свободных параметров модели можно изменять форму шлейфа и характер температурного поля в окрестности детонационной волны в целях более полного соответствия с экспериментальными результатами. Делается вывод об эффективности данного подхода с точки зрения минимизации вычислительных затрат и проведения серии вычислительных экспериментов для оптимизации конструкции камеры сгорания. Разработанный программный код может быть полезен для технических специалистов, рассчитывающих тепловые нагрузки в детонационных камерах сгорания.

Ключевые слова: тепловая нагрузка, детонационная волна, законы сохранения, уравнение теплопроводности, математическое моделирование, программное обеспечение

Введение

В последнее время значительно вырос практический интерес к детонационному горению в газовых смесях. Поскольку давление и температура продуктов сгорания при детонационном сжигании в несколько раз выше по сравнению с обычным (медленным) сжиганием (например, при сжигании стехиометрической водородно-воздушной смеси температура и давление увеличиваются примерно в два раза), возникает естественное предложение по созданию двигателей с повышенной удельной мощностью.

Процесс детонации — это распространение со сверхзвуковой скоростью зоны быстрой экзотермической химической реакции, следующей за фронтом ударной волны. Ударная волна выполняет функцию инициирования реакции окисления, сжимая и нагревая детонирующее вещество (газообразную смесь горючего с окислителем). Фронт ударной волны и зона реакции образуют в комплексе детонационную волну, а выделяющаяся при этом энергия поддерживает ударную волну, обеспечивая самораспространение процесса. Скорость движения детонационной волны может составлять от 2000 до 3000 м/с.

В 1957—1959 годах на основе экспериментальных исследований была открыта мелкомасштабная ячеистая структура детонационного фронта, вызванная неустойчивостью процесса горения. Как выяснилось, размеры ячейки пропорциональны периоду индукции смеси и являются характеристикой ее состава. Знание размеров ячейки очень важно с практической точки зрения, так как позволяет определить критический диаметр, при котором детонационная волна распространяться не будет.

Несмотря на то что детонационному горению посвящено большое число работ (например [1]), остаются открытыми вопросы о температуре во фронте волны и в зоне химических реакций, поведении волны в зависимости от размеров и геометрии каналов камеры сгорания (диафрагм и препятствий), а также влияния на детонацию турбулентности и завихренности потока.

Кратковременность процесса детонационного превращения ($\sim 10^{-7}$ с) и высокие температуры в зоне реакции (порядка нескольких тысяч градусов) делают приемлемыми для определения температуры только оптические методы с высоким временным разрешением, основанные на регистрации излучения детонационного фронта. В литературе имеется

описание немногочисленных опытов, в которых путем спектроскопического анализа вспышки от детонационного заряда удалось непосредственно определить температуры в детонационной волне. Однако с этим методом связаны определенные трудности, сущность которых заключается в том, что источником наиболее интенсивного излучения, наблюдаемого в момент детонации, являются не продукты реакции, а присоединенная ударная волна. Поэтому некоторые авторы считают невозможным прямое определение температуры детонации спектроскопическими методами. В связи с этим проводится большое число экспериментальных исследований (рис. 1, см. третью сторону обложки) и, кроме того, важное место занимают методы математического моделирования [2, 3].

Основной проблемой при проектировании новых двигателей, использующих принцип детонационного горения, является повышенная теплоотдача от продуктов сгорания на стенки камеры из-за более высоких температур в детонационной волне и увеличения коэффициента теплоотдачи в результате срыва пограничного слоя. Очевидно, что тепловые нагрузки на стенки камеры сгорания зависят от многих факторов, таких как температура и размеры фронта детонационной волны, скорость ее перемещения, направление газовых потоков и распределение температурного поля в продуктах горения после прохождения волны.

Учитывая, что при детонационном горении в значительной степени возрастают давление и температура в области фронта волны, которая с большой скоростью перемещается вдоль конструктивно выделенной области, возникает задача адекватного моделирования нестационарного течения и высокоинтенсивных тепловых нагрузок на стенки камеры сгорания. В конечном итоге данная задача сводится к численному решению уравнений в частных производных. Уравнения течения газа основаны на законах сохранения массы, импульса и законе сохранения энергии, которые описываются довольно громоздкими уравнениями в частных производных. Кроме этого, при моделировании надо иметь возможность изменять геометрические размеры и желательную форму камеры сгорания и сопла.

Полное решение общей системы уравнений в частных производных для реагирующих течений хотя и возможно, но с вычислительной точки зрения приводит к большим временным затратам и часто связано с необходимостью привлечения суперкомпьютеров. Обычно в этих случаях применяют известные пакеты, такие как ANSYS, однако даже их использование не гарантирует получения приемлемой инженерной точности при численном исследовании экстремальных течений. Дополнительной проблемой при моделировании такого рода течений является проблема устойчивости численного решения при больших градиентах температур и давлений, а также адаптация алгоритмов для учета

мелкомасштабных вихревых течений [1], что возможно лишь на мелких расчетных сетках.

Параболические уравнения, к которым относятся уравнения теплопроводности, достаточно хорошо изучены и избавлены от этих недостатков. Они хорошо решаются известными пакетами и на обычных персональных компьютерах, позволяют использовать крупные сеточные разбиения (особенно если использовать интегральное представление), а также для них более просто оценить точность получаемых результатов. В силу приведенных причин и в силу того, что для задачи проектирования камеры сгорания определяющими являются лишь тепловые нагрузки, возникла задача упрощения полной системы уравнений и сведения их к уравнениям параболического типа. Такой подход позволяет найти решение с приемлемой инженерной точностью и без больших вычислительных затрат.

Упрощенные параболические уравнения для описания поля температур при детонационном горении

Для вывода уравнений, описывающих передачу теплоты при детонационном горении, будем использовать классические уравнения течения и законы сохранения энергии. Первое предположение, которое можно сделать в целях упрощения вида этих уравнений, заключается в том, что вместо реагирующей смеси можно рассмотреть движущийся в потоке с заданной скоростью и имеющий определенную форму источник теплоты. Это тем более обосновано, так как точный механизм окисления топлива при детонации изучен не в полной мере [1]. В этом случае для анализа тепловой нагрузки на стенки камеры сгорания необходимо лишь задать геометрию и плотность мощности движущегося источника исходя из химической зависимости, определяющей количество энергии, выделяющейся во фронте детонационного горения. Это упрощение позволяет исключить химические реакции и в дальнейшем рассматривать лишь движение газовой смеси и распространение в ней теплоты, приводящей к пространственному изменению температуры вокруг фронта волны. В силу того, что при детонационном горении интерес в основном представляет энергетическая составляющая процесса переноса, рассмотрим закон сохранения энергии [2], записанный в виде уравнений в частных производных (здесь мы пренебрегли работой массовых и поверхностных сил):

$$\begin{aligned} \frac{\partial E}{\partial t} + \frac{\partial}{\partial x} E v_x + \frac{\partial}{\partial y} E v_y + \frac{\partial}{\partial z} E v_z = \\ = q - \frac{\partial}{\partial x} W_x - \frac{\partial}{\partial y} W_y - \frac{\partial}{\partial z} W_z, \end{aligned} \quad (1)$$

где E — полная энергия единицы объема, состоящая из внутренней энергии и энергии движения и потенциальной энергии взаимодействия; v_x, v_y, v_z — скорости течения газа; $q = q(x, y, z, t)$ — плотность мощности тепловых источников (скорость тепло-

выделения внешних источников, отнесенная к единице объема); W_x, W_y, W_z — тепловые потоки за счет теплопроводности. Если рассмотреть несжимаемую жидкость и применить закон Фурье для тепловых потоков, то согласно [2] получаем:

$$\begin{aligned} & \rho \frac{\partial e}{\partial t} + \frac{\partial}{\partial x} e \rho v_x + \frac{\partial}{\partial y} e \rho v_y + \frac{\partial}{\partial z} e \rho v_z = \\ & = \frac{\partial}{\partial x} k_x \frac{\partial T}{\partial x} + \frac{\partial}{\partial y} k_y \frac{\partial T}{\partial y} + \frac{\partial}{\partial z} k_z \frac{\partial T}{\partial z} + q, \end{aligned} \quad (2)$$

где ρ — плотность (постоянная); e — средняя кинетическая энергия, которая в случае идеального газа имеет вид $e = c_v T$; c_v — теплоемкость газа при постоянном объеме (постоянная); T — температура; $k_{x, y, z}$ — коэффициенты теплопроводности по осям.

Уравнение (2) содержит в качестве неизвестных функций скорости движения газа в каждой точке по осям (массовые потоки). Поэтому возникает задача каким-либо образом связать потоки массы с градиентом температур. Для этого проведем следующие рассуждения: заметим, что если на движущуюся точечную массу подействовать постоянной силой в течение продолжительного времени, то при любом направлении начальной скорости с течением времени вектор скорости, а значит, и вектор ее импульса, будет стремиться к направлению действия силы (особенно быстро, если величина этой силы имеет большое абсолютное значение). Поэтому при больших силах или временных отрезках, можно приблизительно записать:

$$MV \approx \alpha F, \quad (3)$$

где функция $\alpha = \alpha(V, F)$ зависит от скорости V и силы F . Исходя из этого соотношения был сделан вывод, что для упрощенного описания течения газов можно предположить, что и массовый поток в каждой точке можно аппроксимировать направлением действия силы — градиентом давления. Таким образом, полагая, что проекции массовых потоков по осям связаны соотношениями

$$\begin{aligned} \rho v_x &= -D_x \frac{\partial p}{\partial x}; \\ \rho v_y &= -D_y \frac{\partial p}{\partial y}; \\ \rho v_z &= -D_z \frac{\partial p}{\partial z}, \end{aligned} \quad (4)$$

где p — давление, а $D_{x, y, z}$ — неизвестные параметры, определяющие связь массовых потоков с градиентом давления.

В соотношениях (4), которые аналогичны закону Фурье для тепловых потоков, варьирование параметров D_x, D_y, D_z позволяет корректировать взаимные направления вектора градиента давления и вектора массового потока. Если эти параметры равны между собой, то направление градиента давления и вектора массового потока совпадают. В противном случае направления этих векторов различны. Как

следствие, появляется возможность более "тонкой настройки" уравнений переноса, особенно если каким-либо образом определить зависимость коэффициентов D от времени или от пространственных координат.

Для простоты будем считать, что мы имеем смесь, близкую к идеальному газу, и тогда для определения давления можно использовать известную зависимость — уравнение состояния идеального газа:

$$p = \rho RT. \quad (5)$$

Дифференцируя (5) по пространственным переменным и учитывая условие несжимаемости, которое в данном случае означает, что плотность не зависит от пространственных переменных, получаем:

$$\begin{aligned} \frac{\partial p}{\partial x} &= \rho R \frac{\partial T}{\partial x}; \\ \frac{\partial p}{\partial y} &= \rho R \frac{\partial T}{\partial y}; \\ \frac{\partial p}{\partial z} &= \rho R \frac{\partial T}{\partial z}. \end{aligned} \quad (6)$$

С учетом (4) получаем:

$$\begin{aligned} \rho v_x &= -\rho R D_x \frac{\partial T}{\partial x}; \\ \rho v_y &= -\rho R D_y \frac{\partial T}{\partial y}; \\ \rho v_z &= -\rho R D_z \frac{\partial T}{\partial z}. \end{aligned} \quad (7)$$

Подставляя полученные выражения в (2), приходим к следующему уравнению в частных производных:

$$\begin{aligned} \rho c_v \frac{\partial T}{\partial t} &= \frac{\partial}{\partial x} c_v T \rho R D_x \frac{\partial T}{\partial x} + \\ &+ \frac{\partial}{\partial y} c_v T \rho R D_y \frac{\partial T}{\partial y} + \frac{\partial}{\partial z} c_v T \rho R D_z \frac{\partial T}{\partial z} + \\ &+ \frac{\partial}{\partial x} k_x \frac{\partial T}{\partial x} + \frac{\partial}{\partial y} k_y \frac{\partial T}{\partial y} + \frac{\partial}{\partial z} k_z \frac{\partial T}{\partial z} + q \end{aligned} \quad (8)$$

или

$$\rho c_v \frac{\partial T}{\partial t} = \frac{\partial}{\partial x} K_x \frac{\partial T}{\partial x} + \frac{\partial}{\partial y} K_y \frac{\partial T}{\partial y} + \frac{\partial}{\partial z} K_z \frac{\partial T}{\partial z} + q, \quad (9)$$

где

$$\begin{aligned} K_x &= c_v \rho R D_x T + k_x; \\ K_y &= c_v \rho R D_y T + k_y; \\ K_z &= c_v \rho R D_z T + k_z. \end{aligned} \quad (10)$$

Таким образом, учитывая сделанные выше допущения, нам удалось избежать решения сложных уравнений движения и получить обычное с формальной точки зрения уравнение теплопроводности для среды, теплофизические параметры которой меняются в зависимости от температуры. Для его решения удобно использовать "метод струн", разработанный ранее для такого рода уравнений [4] и

позволяющий свести (7) к системе интегральных уравнений, которые решаются итерационным способом. Заметим, что итерационный подход, кроме преимуществ по точности расчетов, позволяет также естественным образом учитывать зависимость коэффициентов теплопроводности от температуры.

Уравнения (8) содержат свободные параметры (9), которые могут быть подобраны таким образом, чтобы на тестовых примерах вычисленное распределение температур наиболее точно соответствовало температурам, полученным из экспериментов.

Результаты численного моделирования

Целью проведенных численных расчетов являлись как подтверждение правильности предлагаемого подхода, так и адаптация разработанного ранее программного обеспечения [4, 5] на случай, когда теплофизические характеристики изделия, а также граничные условия зависят от температуры. Обе задачи были успешно выполнены.

В разработанной программной среде (была подана заявка на регистрацию авторских прав) была создана модель линейного безразмерного сопла с соотношением длины канала к его ширине как 10 к 1 (сечение канала постоянно). В силу того, что для решения уравнений (8) используются интегральные уравнения [6], было решено выбрать тестовую сеточную область с небольшим числом точек (10×100), что позволило не только не ухудшить получаемые результаты, но и выполнить расчеты на персональном компьютере. Движение фронта детонационной волны описывалось специальными средствами программной среды как движущийся источник теплоты постоянной мощности, имеющий конфигурацию в виде гребня плоской волны (рис. 2).

Программная среда позволяет задавать различные граничные условия, в том числе и с учетом излучения, но для данных расчетов были выбраны адиабатические условия в целях качественной проверки картины расчетного течения.

В силу того, что предлагаемая модель содержит свободные параметры, были проведены серии рас-



Рис. 2. Модельная область, расчетная сетка и фронт детонационной волны в виде движущегося источника теплоты (показаны его положения в различные моменты времени с постоянным шагом)

четов для определения влияния этих параметров на характер получаемых температурных полей и анализ возможностей по их адаптации для лучшего соответствия с экспериментальной картиной течения.

Результаты выполненных расчетов представлены на рис. 3—5 (см. третью сторону обложки). Течения I, II и III отличаются друг от друга различными обобщенными коэффициентами теплопроводности. В силу того, что вычисления проводились в безразмерном виде, на характере течения не так сказываются абсолютные их значения, как соотношения между собой. В течении I (рис. 3) коэффициенты теплопроводности в продольном и поперечном направлении совпадали ($K_x = K_y$).

В течении II (рис. 4) обобщенная теплопроводность в поперечном направлении была больше, чем обобщенная теплопроводность в продольном направлении в 10 раз. Это привело к тому, что повысилась общая температура за головной частью "шлейфа", а его длина уменьшилась.

В течении III была смоделирована обратная ситуация — теплопроводность по оси X (продольное направление) была в 10 раз больше теплопроводности в поперечном направлении (ось Y).

Характер течения, показанный на рис. 5, говорит о том, что в этом случае более выражена зона предварительного разогрева перед ударной волной, а также значительно увеличиваются размеры высокотемпературного "шлейфа".

Расчеты, показанные на рис. 3—5, выполнялись при одной и той же плотности распределения тепловых источников, и, как отмечалось ранее, менялись лишь обобщенные коэффициенты теплопроводности.

Как видно из полученных скрин-шотов, картина течения качественно весьма хорошо описывает реальные процессы детонационного горения. Варьирование свободных параметров позволяет изменить форму шлейфа за детонационной волной, а также размеры зоны предварительного разогрева перед волной (которая хорошо видна на рис. 5).

Заключение

Численная верификация разработанного подхода, основанного на упрощенном математическом описании процесса распространения детонационной волны в газовой смеси с помощью параболических уравнений теплопроводности, подтвердило правильность исходных предположений. Применяемые расчетные алгоритмы, основанные на интегральном представлении процессов теплопередачи [5], показали быструю сходимость и устойчивое поведение в условиях больших градиентов температур. Использование крупномасштабного сеточного разбиения позволяет значительно сократить вычислительное время, что для проведенных расчетов составило несколько минут, и, таким образом, данный подход позволяет проводить серии расчетов и варьировать геометрию и размеры камеры сгорания в широком диапазоне.

Однако открытым остается вопрос о подборе эффективных коэффициентов теплопроводности для получения адекватных численных результатов. В качестве первого критерия можно попытаться выбирать эти параметры из условия согласования расчетных данных с картиной истечения газовых потоков на срезе экспериментального двигателя (см. рис. 1). Полученные расчетные данные (рис. 3–5), демонстрируют, что геометрия высокотемпературной области существенным образом зависит от величины этих параметров и, таким образом, качественное визуальное (или с помощью тепловизора) совпадение геометрии высокотемпературного ядра на срезе сопла с расчетными данными может быть одним из критериев правильности расчетов.

Для полного расчета тепловых нагрузок на стенки камеры сгорания на основе предложенной модели необходимо задать геометрию, размеры и теплофизические характеристики материала стенок камеры сгорания, условия ее охлаждения (температуру охлаждающей жидкости), частоту импульсов, удельную теплоту сгорания топлива и количество смеси.

Созданное программное обеспечение и разработанный алгоритм численного решения может быть рекомендован конструкторам технических отделов, занимающихся детонационным горением, в том

числе и спиновым, для расчета тепловых нагрузок на стенки камеры сгорания, включая камеры с переменным сечением.

Автор выражает благодарность В. С. Зарубину, профессору кафедры прикладной математики МГТУ им. Н. Э. Баумана, Р. И. Нигматулину, заведующему кафедрой волновой и газовой динамики мехмата МГУ, академику РАН за полезные обсуждения и ценные замечания.

Статья была подготовлена при поддержке гранта РФФИ № 14-38-50538 мол. пр.

Список литературы

1. **Импульсные** детонационные двигатели / Под ред. С. М. Фролова. М.: Торус Пресс, 2006, 592 с.
2. **Зарубин В. С.** Математическое моделирование в технике, 2-е изд. М.: МГТУ им. Н. Э. Баумана, 2003. 497 с.
3. **Patankar S.** Numerical Heat Transfer and Fluid Flow. New York: Hemisphere Publishing Corporation, 1980. 199 p.
4. **Светушков Н. Н.** Метод геометрических интегралов при моделировании процессов теплопередачи в задачах с фазовыми превращениями // Вестник Московского авиационного института. 2012. Т. 19, № 5. С. 182–186.
5. **Светушков Н. Н.** Программный комплекс по расчету нестационарных тепловых полей в сложных двумерных объектах методом геометрических интегралов. Свидетельство о государственной регистрации программ № 2013615984 от 25 июня 2013 года.
6. **Светушков Н. Н.** Метод струн в задачах многомерной нестационарной теплопроводности // Информационные технологии. 2014. № 12. С. 14–19.

N. N. Svetushkov, Associate Professor, e-mail: svt.n.n@mail.ru,
Moscow Aviation Institute (National Research University), Moscow, Russia

Simplified Mathematical Model for Temperature Fields Calculation in Detonation Combustion

This paper investigates the possibility to calculate the heat load on the combustion chamber wall in detonation combustion, which is important in the development of industrial detonation engines, including those with spin detonation combustion. For the calculation was proposed simplified mathematical model based on the parabolic partial differential equations, which reflects the main features of the propagation of detonation in the combustion chamber. The article details the derivation of this equation is presented and shows the main assumptions used in its preparation. To solve the resulting equations and numerical calculations by the authors used a new approach — a method of strings based on the integral representation of the heat equation. The article shows the advantages of the proposed approach and noted that its use avoids the "nonphysical" oscillations in the numerical solution in case of large temperature gradients. For the purpose of calculation was modified previously developed software environment that allows you to create two-dimensional model of the combustion chambers, in accordance with their configuration, and set the initial and boundary conditions. These results have shown that by varying the free parameters of the model, you can change the shape of the plume and the nature of the temperature field in the vicinity of the detonation wave to better compliance with the experimental results. The article concludes that the effectiveness of this approach in terms of minimizing the computational cost and a series of numerical experiments to optimize the combustion chamber design. The developed code can be useful for technicians who expect thermal loads in detonation combustion chambers.

Keywords: thermal load, the detonation wave, the conservation laws, the heat equation, mathematical modeling software

References

1. **Impul'snye** detonacionnye dvigateli. Pod red. S. M. Frolova. Moscow: Torus Press, 2006, 592 s.
2. **Zarubin V. S.** Matematicheskoe modelirovanie v tehnikе, 2-e izdanie. Moscow: MGTU im. N. Je. Baumana, 2003. 497 p.
3. **Patankar S.** Numerical Heat Transfer and Fluid Flow. New York: Hemisphere Publishing Corporation, 1980, 199 pp.
4. **Svetushkov N. N.** Metod geometricheskikh integralov pri modelirovanii processov teploperedachi v zadachah s fazovymi prevrash-

henijami, *Vestnik Moskovskogo aviacionnogo instituta*, 2012, vol. 19, no. 5, pp. 182–186.

5. **Svetushkov N. N.** Programmnij kompleks po raschetu nestacionarnykh teplovykh polej v slozhnykh dvumernykh ob#ektah metodom geometricheskikh integralov. Svidetel'stvo o gosudarstvennoj registracii programm № 2013615984 ot 25 ijunja 2013 goda.

6. **Svetushkov N. N.** Metod strun v zadachah mnogomernoj nestacionarnoj teploprovodnosti, *Informacionnye tehnologii*, 2014, no. 12, pp. 14–19.

П. Ш. Гейдаров, канд. техн. наук, доцент,
Институт Системного Управления НАН Азербайджана, г. Баку

Алгоритм кратчайшего маршрута на основе выделенного набора маршрутов

Рассмотрен алгоритм определения кратчайшего маршрута на основе существующего набора маршрутов. Алгоритм рассматривался применительно к общественному транспорту путем последовательного усложнения и расширения возможностей алгоритма. Приведен базовый алгоритм без географической привязки к местности и с возможностью одной пересадки. Предложено описание структуры базы данных, позволяющей в качестве наименований пунктов остановок гибко использовать названия различного числа объектов, имеющих соответствие с пунктами остановок. На основе базового алгоритма рассмотрен алгоритм с двумя или несколькими пересадками, а также приведено описание возможностей расширения функциональности алгоритма применительно к разным видам транспорта и с привязкой к географическим координатам местности. Также дано описание программных и структурных способов ускорения работы алгоритма.

Ключевые слова: кратчайший маршрут, городской транспорт, общественный транспорт, транспорт в Баку, Центр интеллектуального управления транспортом

Введение

Одной из актуальных задач является задача определения кратчайшего пути от одного пункта до другого, и как частный случай этой задачи — задача определения кратчайшего пути с использованием городского транспорта: автобусов, троллейбусов, метро и т.д. В отличие от классической постановки задачи кратчайшего маршрута, особенность решения данной задачи заключается в наличии изначально известного набора маршрутов, на основе которых и требуется определение кратчайшего маршрута. Решение данной задачи является актуальным на фоне современных темпов и масштабов развития городов, определяемых такими событиями, как строительство новых зданий, высоток, парков, улиц, проспектов; переименование названий улиц, учреждений; замена, объединение или перемещение одних организаций в другие; а также сокращение, переименование или введение новых транспортных линий. Все это зачастую приводит к сложностям ориентирования на местности. Например, в г. Баку за последние 20 лет несколько раз менялась вся сеть автобусных маршрутов: траектория их передвижения, номера и количество маршрутов. Менялись также названия улиц, парков, проспектов. Решение данной задачи также полезно для развития сферы туризма, поскольку задача создания условий быстрого и удобного ориентирования в городе, без необходимости знания языка и города может в целом способствовать развитию этой сферы.

1. Постановка задачи

Нужно отметить, что для задач определения кратчайшего пути существуют различные алгоритмы, которые в целом можно подразделить на два типа: алгоритмы перебора, выдающие оптимальные ре-

зультаты [1—4], и эвристические алгоритмы, определяющие приближенно неточные результаты (например, муравьиные, генетические алгоритмы... [5—12]). Применение последних обосновано в тех случаях, когда задача является нерешаемой иными способами или требует просмотра слишком большого объема информации в ограниченный период времени. В качестве же примера алгоритма перебора для задачи определения кратчайшего пути, можно было бы привести используемый в системах GPS-навигации алгоритм перебора Дейкстры [1, 4], определяющий маршрут на основе перебора вершин графа, предварительно созданных на основе информации о существующих дорогах, маршрутах и т.д. Алгоритм Дейкстры, а также похожие алгоритмы перебора, такие как Беллмана—Форда, Флойда—Уоршелла и др. [1] имеют универсальный характер и применимы к разным задачам, но при этом применительно к отдельным задачам, и в частности в рассматриваемой задаче, имеют ряд слабых мест. В частности, потребуются реализация дополнительного алгоритма для создания графа на основе базы существующих транспортных маршрутов. Во время или после реализации этих алгоритмов понадобится также алгоритм выполнения привязки полученного пути маршрута к существующим транспортным маршрутам. При этом числа пересадок для полученного кратчайшего маршрута может оказаться чрезмерно много, что будет неудобно и неестественно для практического применения этих маршрутов. Кроме того, в ряде случаев универсальные алгоритмы перебора [1, 4] будут более медлительными, чем предлагаемый алгоритм. Связано это с тем, что алгоритм перебора графов [4] последовательно перебирает все смежные вершины графа, двигаясь от одной вершины к другой, таким образом до определения кратчайшего маршрута

будет охвачено значительное число вершин графа (рис. 1, а). Тогда как в предлагаемом алгоритме благодаря наличию базы маршрутов, а также начального и конечного пунктов {НП, КП} число рассмотренных вершин может быть значительно меньше, поскольку поиск ведется на основе тех маршрутов, в которых имеются эти пункты. Последнее особо заметно при наличии малого числа маршрутов и пересадок между начальными и конечными пунктами кратчайшего маршрута (рис. 1, а, б).

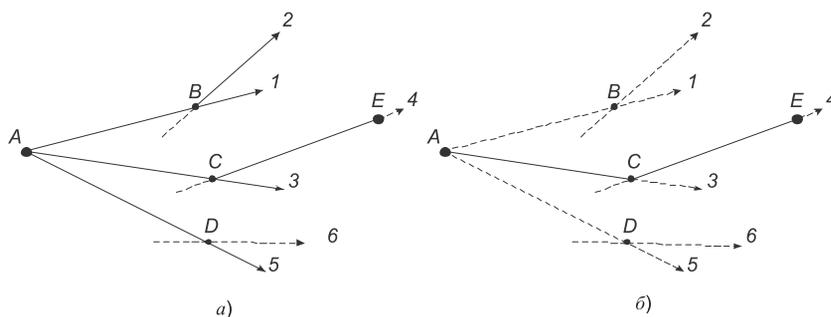


Рис. 1. Пример графа, образованного из шести прямолинейных маршрутов с рассмотренными (сплошные линии) ребрами графа: а — по алгоритму Дейкстры; б — по рассматриваемому алгоритму

На рис. 1 приведен пример графа с числом рассмотренных ребер (сплошные линии графа) по алгоритму Дейкстры (рис. 1, а), и по рассматриваемому алгоритму (рис. 1, б), для задачи определения маршрута от точки А до Е на графе, построенном из шести прямолинейных маршрутов, пересекающихся в комбинации {1, 2} в точке В, {3, 4} в точке С и {5, 6} в точке D. Можно видеть, что для определения маршрута АСЕ алгоритму Дейкстры, для данного примера понадобится пять шагов для достижения точки Е, тогда как в рассматриваемом алгоритме — два шага. Здесь нужно отметить, что при увеличении числа пунктов пересадок в кратчайшем маршруте, а также увеличение числа маршрутов, проходящих через начальный и конечный пункты, число рассматриваемых вершин графа также будет увеличиваться и соответственно скорость работы предлагаемого алгоритма будет уменьшаться.

2. Определение структуры базы данных и алгоритм определения кратчайшего маршрута с одной пересадкой, без привязки маршрутов к системам отчета

Рассмотрим для начала структуру и работу алгоритма нахождения кратчайшего маршрута с одной пересадкой, определяемого по числу остановок и применительно к автобусным маршрутам г. Баку. При этом данный алгоритм реализуется без привязки к географическим координатам пунктов остановок.

Для работы алгоритма предварительно создается структура базы данных, представляющая собой список всех номеров автобусных маршрутов, после каждого из которых следует последовательный перечень всех наименований пунктов остановок маршрута (пример 1). При этом наименование пункта остановки маршрута может включать в себя наименования сразу нескольких объектов, расположенных вблизи пункта остановки транс-

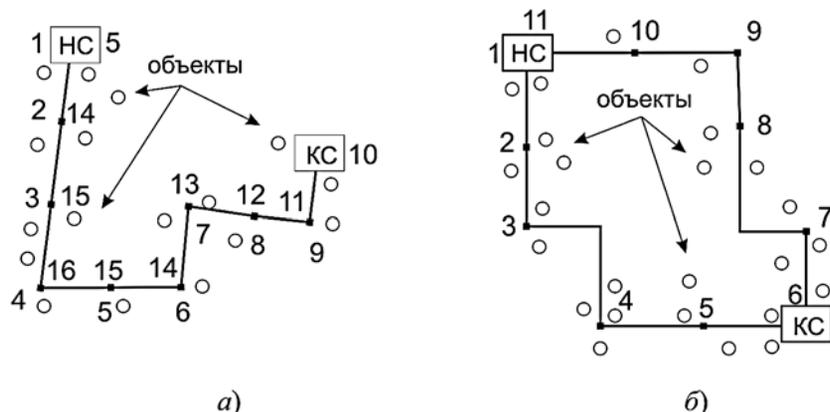


Рис. 2. Незамкнутый маршрут (а), замкнутый маршрут (б): НС, КС — начальная и конечная станция (остановка) маршрута

порта (рис. 2). Структура базы данных организована следующим образом. Все пункты маршрутов транспортных средств расположены построчно друг за другом. Каждый маршрут начинается с номера маршрута транспорта. Название каждого пункта остановки состоит из наименований различных объектов, расположенных вблизи данной остановки маршрута. При этом все названия объектов и улиц, касательные одной остановки транспорта, расположены на одной строке и разделены запятыми последовательно друг за другом в порядке удаления объектов от пункта остановки. Такая организация структуры базы данных позволяет в дальнейшем гибко управлять и менять базу данных, добавляя новые или удаляя старые объекты.

Пример 1. Пример фрагмента базы данных № 104

1. М. Дружба народов
2. Пл. Украины, м. Ази Асланова
3. Радиозавод
- ...
10. Гагаринский мост, больница Нефтяников
11. Консерватория, Центральный банк
12. М. 28 мая, железнодорожный вокзал, Университет нефти и химии
13. Дворец Республики, больница № 4, ул. Басина

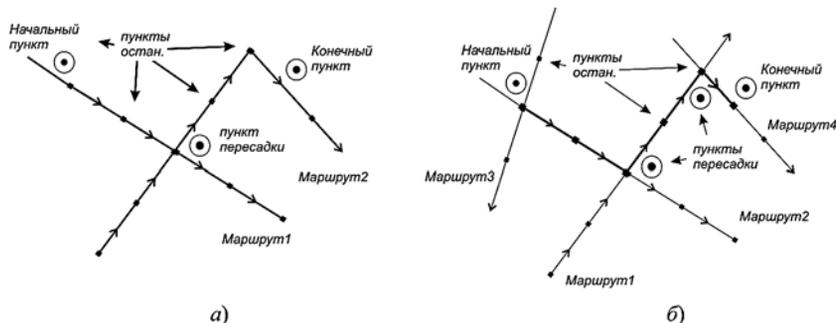


Рис. 3. Пересекаемые (а) и непересекаемые (б) маршруты

14. Пл. Физули, шахматная школа, ул. Гуси Гаджиева, МИД

15. ЦУМ, Пл. фонтанов, торговая

Нужно отметить, что передвижение автобусов может происходить как по замкнутым (рис. 2, б), так и по незамкнутым маршрутам (рис. 2, а). В случае замкнутого маршрута пункты остановок маршрута перечислены в тексте только один раз сверху вниз по направлению движения транспорта (пример 1), в случае незамкнутого маршрута для целостности работы алгоритма пункты остановок маршрута перечислены дважды, сначала в направлении движения от начальной станции (НС) к конечной станции (КС), затем в обратном направлении от КС до НС.

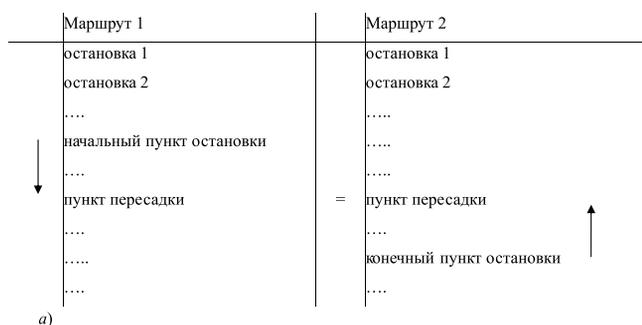
Выполнение алгоритма состоит из четырех циклов (алгоритм 1). При этом 1-й и 2-й циклы выполняют поиск начальных и конечных пунктов назначения в базе данных, а 3-й и 4-й циклы опре-

деляют пункт пересадки, и, соответственно, выполняются только в том случае, когда начальный и конечные пункты назначения в первых двух циклах определяются на разных маршрутах транспорта.

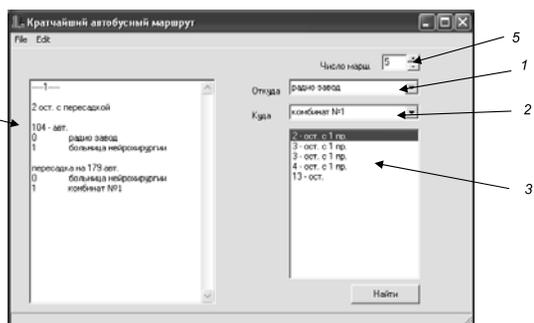
Алгоритм работает следующим образом. После того как пользователь вводит начальный и конечный пункты назначения на рис. 4, б текстовые поля 1 и 2 с наименованиями: "откуда", "куда" (выполняется работа 1-го цикла), последовательно просматривается база данных, начиная с самой верхней строки, на предмет определения первого совпадения начального пункта назначения ("откуда") с записями пунктов остановок маршрутов. При этом просматриваются все записи строки, разделенные запятыми на одной строке. Как только определяется первое совпадение, запоминается номер маршрута, а также номер строки для этой записи. Начиная с этой строки, передается выполнение на 2-й цикл, где последовательно просматриваются строки в целях определения второго пункта назначения ("куда"), начиная с самой верхней строки базы данных. Если второй пункт назначения определяется в пределах этого же маршрута, мы получаем маршрут проезда без пересадки. Если же строка второго пункта назначения определяется на маршруте другого номера маршрута автобуса, то выполняется проверка полученных маршрутов на наличие пункта пересадки. С этой целью последовательно выполняется сравнение записей строк найденных маршрутов (рис. 3, а) на предмет определения идентичного наименования пункта, который и будет являться пунктом пересадки — остановкой пересечения двух маршрутов. При этом просмотр строк идет в направлении движения первого маршрута (3-й цикл, алгоритм 1), начиная с определенной выше строки пункта назначения, и в обратном направлении для второго маршрута (4-й цикл, алгоритм 1), начиная со строки конечного пункта назначения (рис. 4, а). В случае если пункт пересадки определяется, то весь маршрут отдельно запоминается как один из возможных вариантов пути.

При этом при сохранении маршрута помимо начального, конечного и пункта пересадки также дополнительно запоминаются промежуточные пункты проезда объектов остановок, в качестве которых выбираются объекты, указанные первыми в строках, как ближайшие к пунктам остановок.

Если пункт пересечения не определяется, то это означает, что маршруты не пересекаются (рис. 3, б). В этом случае продолжается процесс выполнения 2-го цикла на предмет определения следующего совпадения конечного пункта назначения в оставшихся записях базы данных. Процедура 2-го цикла повторяется аналогичным образом до тех пор, пока процесс просмотра не достигнет конца базы данных,



а)



б)

Рис. 4. Схема работы 3-го и 4-го циклов (а) и программный модуль алгоритма определения кратчайшего маршрута с одной пересадкой без привязки к географическим координатам, в применении к автобусным маршрутам г. Баку (б)

после чего алгоритм возвращается к 1-му циклу и продолжает поиск нового начального пункта со строки последнего определенного начального пункта. Все полученные маршруты, включая номера маршрутов, наименования пунктов остановок, а также место пересадки маршрутов сохраняются и выводятся на экран (см., например, рис. 4, б (3, 4)). При этом полученные маршруты являются вариантами маршрутов без пересадки или с одной пересадкой.

Алгоритм 1. Определения кратчайшего маршрута с одной пересадкой

Блок А — определение строк i, j для НП и КП.

Цикл 1, определение строки НП.

1. Начальные данные: $k = 0$, НП, КП, база всех маршрутов (N — число строк в базе).
2. $k = k + 1$. Переход на $k + 1$ -ую строку в базе маршрутов, $i = k$.
3. Если $k = N$: переход в 14.
4. Если НП = {Объекту строки i }, то $k = 0$, переход в 5 (цикл 2), если нет, переход в 2.

Цикл 2, определение строки КП

5. $k = k + 1$. Переход на $k + 1$ -ую строку в базе маршрутов, $j = k$.
6. Если $k = N$: переход в 2.
7. Если КП = {Объекту строки j } переход в 8 (цикл 3), если нет, переход в 5.

Блок Б — определение пункта пересадки.

Циклы 3, 4.

8. $n = j, k = i$.
9. $k = k + 1$.
10. Если строка пуста, $k = ""$, переход в 12 (цикл 4).
11. Если {Объект строки k } = {Объект строки n }, то k — строка пересадки ($ПР = k$) сохраняем, вычисляем длину маршрута, и выполняем переход в 5, если нет, то переход на 9.
12. $n = n - 1, k = i$.
13. Если в строке n есть "номер маршрута или знак №", то $k = j$ переход в 5 (цикл 2).
14. Переход в 9.

Блок С — обработка и вывод результатов алгоритма.

15. Сортируем полученные маршруты по возрастанию длины маршрута и выводим пользователю.
16. Завершение работы алгоритма.

3. Определение кратчайшего маршрута с двумя или несколькими пересадками

На основе данного алгоритма может быть также получен алгоритм определения кратчайшего маршрута с двумя или несколькими пересадками. Например, для определения кратчайшего маршрута с двумя пересадками (алгоритм 2) выполняется последовательное переназначение каждой k -й остановки i -го маршрута новым начальным пунктом

(НП $_k^{(i)}$) для нового маршрута {НП $_k^{(i)}$, КП}. При этом конечный пункт (КП) остается неизменным — первоначально заданным. После каждого переназначения цикл программы перенаправляется в начало 1-го цикла программы и по аналогичной схеме повторяется поиск пункта пересадки. Полученные маршруты сохраняются. Далее по такой же схеме выполняется обратная задача. Начальный пункт (НП) остается заданным, а конечный пункт (КП $_n^{(j)}$) меняется путем перебора остановок второго выбранного j -го маршрута. Для каждого полученного маршрута между определенными пунктами {НП $_k^{(i)}$, КП} или {НП, КП $_n^{(j)}$ } выполняется описанный выше алгоритм на определение кратчайшего маршрута с одной пересадкой ПР $_{k,n}^{(i,j)}$. В результате получаются итоговые варианты маршрутов с двумя пересадками {НП, НП $_k^{(i)}$, ПР $_{k,0}^{(i,j)}$, КП} и {НП, ПР $_{0,n}^{(i,j)}$, КП $_n^{(j)}$, КП} (рис. 5). Здесь пункты НП $_k^{(i)}$ и КП $_n^{(j)}$ определяют дополнительные пункты пересадки для маршрута с двумя пересадками. На основе полученного списка всех маршрутов вычисляются кратчайшие маршруты. Например, на рис. 5 кратчайший маршрут будет: {НП, ПР $_{0,2}^{(i,j)}$, КП $_2^{(j)}$, КП}.

Если при выполнении основного алгоритма кратчайший маршрут между пунктами {НП $_k^{(i)}$, КП} или {НП, КП $_n^{(j)}$ } определяется без пересадки, то мы получаем маршрут с одной пересадкой по схеме — {НП, НП $_k^{(i)}$, КП} или {НП, КП $_n^{(j)}$, КП}. Если же при назначении пунктов НП $_k^{(i)}$, КП $_n^{(j)}$ выполняется ус-

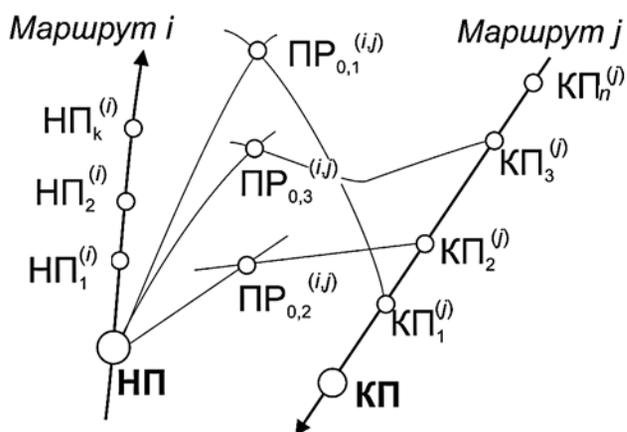


Рис. 5. Пример схемы определения маршрутов с двумя пересадками между пунктами НП и КП, с перебором пунктов КП на транспортном маршруте j

ловие $НП_k^{(i)} = КП$ или $НП = КП_n^{(j)}$, то получим маршрут без пересадки {НП, КП}. То есть расширенный таким образом алгоритм определения кратчайшего маршрута с двумя пересадками включает в себя и определение кратчайшего маршрута с одной пересадкой и без пересадки.

Алгоритм 2. Определение кратчайшего маршрута с двумя пересадками.

1. Блок А (алгоритм 1): определяем номера строк i, j для НП и КП.
Цикл 1.
2. $k = i - 1$.
3. $k = k + 1$, — выбираем новый начальный пункт, $НП_k =$ Объект строки k .
4. Если строка k пуста: $k = ""$, то переходим в 8, цикл 2.
5. Переход в 13, (блок Б) с переменными $НП_k$ и КП.
6. Сохраняем общий маршрут {НП, $НП_k$, ПР, КП}, и вычисляем длину маршрута.
7. Переход в 3.
Цикл 2.
8. $k = j + 1$.
9. $k = k - 1$, — выбираем новый конечный пункт, $КП_k =$ Объект строки k .
10. Если в строке k есть "номер маршрута или знак №", то переход в 1, Блок А, если нет, то выполняем переход в 13, Блок Б с переменными НП и $КП_k$.
11. Сохраняем общий маршрут {НП, ПР, $КП_k$, КП}, вычисляем длину маршрута.
12. Переход в 9.
13. Блок Б (алгоритм 1): определяем пункт пересадки (ПР), возвращаемся в пункт вызова: 5, 9.
14. Блок С (алгоритм 1): сортируем по длине маршрута и выводим пользователю полученные маршруты.
15. Завершение алгоритма.

Аналогичным образом на базе основного алгоритма 1 путем выполнения дополнительных циклов можно получить кратчайшие маршруты с тремя и более числом пересадок, например {НП, $НП_k^{(i)}$, $ПР_{k,n}^{(i,j)}$, $КП_n^{(j)}$, КП}. Нужно сказать, что маршруты с большим числом пересадок создают дополнительные неудобства для самих пассажиров и при этом увеличивают длительность и скорость работы алгоритма. Обычно в условиях большого города достаточны маршруты с числом пересадок 1—3.

В процессе работы алгоритма или после выполнения поиска списка маршрутов, выполняется обработка (алгоритм 1, Блок С) полученных маршрутов. Определяется длина маршрутов, выполняется перераспределение (сортировка) полученных маршрутов в порядке увеличения их длины. Например, на рис. 4, б самый верхний маршрут в списке 3 соответствует самому кратчайшему маршруту. При этом в наиболее простом применении длина маршрута

может оцениваться числом остановок до и после пересадки (рис. 4, б), с учетом также числа пересадок, например, по выражению:

$$D = \sum_{i=0}^P N_i + P, \quad (1)$$

где N_i — число пунктов остановок для i -го маршрута; P — число пересадок. Такой подсчет дает приблизительную оценку и может быть применим, например, в условиях города, где транспортные маршруты, как правило, имеют достаточно много пунктов остановок с короткими и сравнительно равными интервалами пути.

4. Расширение возможностей алгоритма и привязка маршрутов к географическим системам отсчета

В приведенном выше алгоритме длина маршрута оценивалась числом пунктов остановок транспорта. Для более точной оценки пути D в алгоритме необходимо учитывать расстояние между остановками. В этом случае в каждой строке, в конце строки базы данных, вводится дополнительный параметр, определяющий значение расстояния от данного пункта остановки до следующего. При этом для локализации программой данного пункта в строке необходимо инициализировать значение данного параметра, например, как показано ниже, буквой d .

10. Гагаринский мост, больница Нефтяников, $d377$

11. М. Хатаи, $d232$

В процессе работы алгоритм просматривает значение d для каждой строки выделенного маршрута и определяет длину пройденного пути по выражению

$$D = \sum_{i=0}^P \sum_{j=0}^{N_i} d_{i,j,j+1}, \quad (2)$$

где $d_{i,j,j-1}$ — расстояние между j -й и $j+1$ -й пунктами остановок для i -го маршрута; N_i — число пройденных пунктов остановок в i -м маршруте; P — число автобусных маршрутов, используемых в данном маршруте пути.

В выражении (2) значение $d_{i,j,j+1}$, определяющее расстояние между смежными пунктами j и $j+1$ в i -м маршруте, определяется предварительно с учетом координат этих пунктов (x_j, y_j) , (x_{j+1}, y_{j+1}) , а также траектории движения транспорта на данном промежутке. При этом необходимые значения координат могут быть определены путем использования, например, GPS-навигаторов, или же с помощью геоинформационных системных (ГИС) карт [13]. Если траектория движения транспорта между пунктами j и $j+1$ не прямолинейна, то для определения значения длины $d_{i,j,j+1}$ могут быть либо использованы ГИС, для которых встроены

функции вычисления непрямолинейных путей, либо данный путь предварительно разбивается на набор K прямолинейных отрезков $d_{i,j,j+1,k}$:

$$d_{i,j,j+1,k} = \sum_{k=0}^K d_{i,j,j+1,k}. \quad (3)$$

При этом значение $d_{i,j,j+1,k}$ между двумя точками A и B определится как

$$d_{i,j,j+1,k} = d_{|A-B|} = \sqrt{(x_A - x_B)^2 + (y_A - y_B)^2}. \quad (4)$$

Если же значения x и y заданы в географических системах координат (широта и долгота), например в случае использования геоинформационных систем [13], то расстояние $d_{i,j,j+1,k}$ между двумя заданными точками A и B определится по выражению:

$$d_{i,j,j+1,k} = d_{|A-B|} = d_\alpha R, \quad (5)$$

где R — средний радиус земного шара ($R = 6\,372\,795$ м, d_α — расстояние между пунктами A , B , измеряемое в радианах, и определяемое, например, по формуле гаверсинусов [14]:

$$d_\alpha = 2 \arcsin \left(\sqrt{\sin^2 \left(\frac{x_A - x_B}{2} \right) + \cos(x_A) \cos(x_B) \sin^2 \left(\frac{y_A - y_B}{2} \right)} \right), \quad (6)$$

где x_A , x_B , y_A , y_B — значения широты и долготы в точках A и B . Значения $d_{i,j,j+1,k}$, определяемые по выражениям (3–6), можно вычислять предварительно, либо в процессе работы алгоритма, последнее может быть необходимо в случае необходимости вырисовывания траектории пути. При этом значения координат x , y , определяющие каждый отрезок $d_{i,j,j+1,k}$, могут быть добавлены в базу данных, например как множества $X\{x_i\}$ и $Y\{y_i\}$, где x_1 и y_1 — значения координат пункта остановки, соответствующей данной строке:

10. парк Самеда Вургуна, железнодорожное управление, d462, $X\{40.380421, 40.378034, 40.376882\}$, $Y\{49.850292, 49.852502, 49.853092\}$

11. Нефтяная Академия, м. 28 Мая, d79, $X\{40.376588, 40.376301\}$, $Y\{49.852138, 49.85129\}$

12. Отель Empire, посольство Великобритании, d225, $X\{40.375615, 40.37501, 40.375402\}$, $Y\{49.851644, 49.849745, 49.849434\}$

Здесь значения координат заданы в географических координатах широты x и долготы y , полученных путем ГИС-карт и выраженные в градусах. Соответственно для применения выражений (5, 6) необходимо значения x , y перевести в радианы, по выражениям

$$x_{\text{рад}} = \frac{2\pi x_{\text{град}}}{180}; y_{\text{рад}} = \frac{2\pi y_{\text{град}}}{180}, \quad (7)$$

где $x_{\text{рад}}$, $x_{\text{град}}$, $y_{\text{рад}}$, $y_{\text{град}}$ — угловые значения в радианах и градусах, $\pi = 3,141592654$.

Приведенный выше алгоритм может также быть использован применительно к другим видам транспорта: трамвая, метро, троллейбуса и т.д., как отдельно для каждого вида транспорта, так и с использованием разных видов транспорта. При этом в последнем случае необходимо учесть среднюю скорость передвижения каждого вида транспорта, а также среднее время задержки на пункте остановки для данного вида транспорта, значения которых могут быть указаны, например, в первой строке, рядом с номером маршрута, как показано ниже:

№ 104, v40, tz5

1. м. Дружба народов

2. пл. Украины, м. Ази Асланова, d420

В этом случае вместо значения кратчайшего пути D может быть использовано значение потраченного времени пройденного пути T , определяемого по выражению

$$T = \sum_{i=1}^P \left(\frac{D_i}{v_i} + tz_i \cdot N_i \right), \quad (8)$$

где D_i — расстояние пути для i -го маршрута, определяемое по выражениям (2, 3), v_i и tz_i — средние значения скорости и времени задержки на остановке для i -го маршрута.

При этом в зависимости от вида транспорта вместо номера маршрута транспорта могут быть указаны другие наименования, используемые для данного транспорта, например для метро это может быть название конечной станции линии маршрута метро, а каждый маршрут линии метро будет представляться как отдельный маршрут:

Метро Старая крепость, v50, tz2

...

ст. Дружба народов, торговый центр Лачын, рынок 8 км.

ст. Нефтчилар

ст. Кара Караева, магазин Орбита

ст. Улдуз, стадион Шяфа

ст. Нариманова, пос. Монтина, детская поликлиника № 1

...

Если транспорт работает по расписанию, например как в случае железнодорожного транспорта, авиатранспорта, междугородних автобусов, то возможно добавить в каждой строке маршрута транспорта дополнительную запись времени отправки с данного пункта маршрута согласно расписанию. В этом случае необходимо дополнительно выполнять проверку сопоставления времени отправки транспорта по расписанию, со временем T , определенным по выражению (8), для предыдущих рассмотренных регулярно работающих видов транспорта.

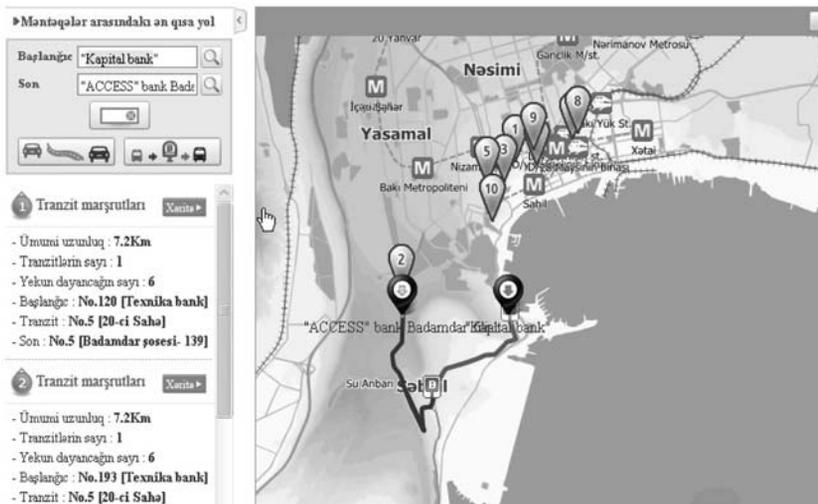


Рис. 6. Применение алгоритма для автобусного транспорта г. Баку

5. Методы ускорения работы алгоритма

Для ускорения работы алгоритма возможны как программные способы, так и структурные: путем оптимизации структуры базы данных.

Программные способы ускорения алгоритма. В качестве программных изменений возможно отметить следующие возможности.

Процесс поиска соответствий начального и конечного пунктов в базе данных, выполняемых в 1-м и 2-м цикле, можно ускорить до выполнения одного цикла. С этой целью в процессе просмотра базы данных уже в 1-м цикле выполняется поиск сразу всех соответствий как начального, так и конечного пункта. Все найденные соответствия начального и конечного пунктов сохраняются в отдельные две строки (первая для начального пункта, вторая для конечного пункта) с указанием номеров строк в базе данных. Например:

НП: Ж/д Вокзал 15, 23, 55, 72, 117, 149

КП: Кинотеатр Дружба 19, 32, 47, 89, 92, 137

Далее, работа алгоритма будет продолжена по аналогичной схеме, но уже с полученным списком номеров строк для найденных начальных и конечных пунктов маршрутов в базе. Ссылка на объект в основной базе будет выполняться по номеру строки. Таким образом процесс просмотра базы данных уменьшится на порядок одного цикла.

Еще одна возможность ускорения процесса поиска кратчайшего маршрута — это ограничение

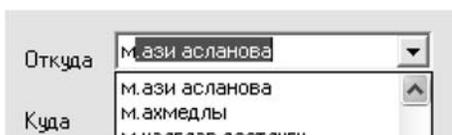


Рис. 7. Пример быстрого ввода наименования объекта пункта отправки в поле списка "Откуда"

числа выводимых маршрутов. Например, на рис. 4, б список выводимых маршрутов в поле 3 ограничивается числом маршрутов, введенных в поле 5. Таким же способом можно получить определение только одного кратчайшего маршрута. Для ускорения процесса можно также дополнительно ограничить поиск маршрутов некоторыми заданными значениями D_{max} или T_{max} . В этом случае в процессе вычислений значений D или T при превышении этих ограничений рассматриваемый маршрут будет далее не рассматриваться и удаляться из списка выбранных маршрутов.

Нужно отметить, что определение нескольких маршрутов имеет смысл как возможность получения нескольких альтернативных путей. Например,

в случае когда данная система используется совместно с другими транспортно-дорожными системами, такими как системы мониторинга дорожно-транспортных происшествий, мониторинга загруженности дорог (пробок) и т. д. В этом случае альтернативой кратчайшего пути может стать другой маршрут — менее короткий, но при этом менее загруженный.

Для ускорения получения результатов работы алгоритма можно также использовать возможности ускоренного ввода наименований объектов пользователем, реализованного также в целях быстрого и корректного ввода наименований объектов путем сравнения вводимого набора букв в списках полей ввода "откуда", "куда" (см. рис. 4, б (1, 2)) и списках "başlangıç", "son" (рис. 6) с содержанием похожих по написанию объектов базы данных. По результатам сравнения выполняется также сортировка схожих записей и выводятся наиболее близкие по написанию записи на просмотр пользователю (рис. 7) с возможностью дальнейшего быстрого отбора нужного наименования объекта ввода.

Оптимизация структуры базы данных. В качестве структурных дополнений возможно привести следующие изменения.

Для случая системы с привязкой к географическим системам координат можно ускорить процесс выполнения алгоритма путем отделения наименований объектов базы данных от численных значений координатных данных, отображающих пункты остановок и траектории движения транспорта от одной остановки к другой. Последняя база и будет являться основной базой данных, с которой будет работать программа. Наименования объектов базы данных будут храниться в отдельной базе данных, например, в алфавитном порядке. Для привязки каждого объекта к пунктам остановки в базе объектов для каждого объекта будут также указаны географические координаты объекта. В этом случае

работа 1-го и 2-го циклов алгоритма изменится следующим образом. В начале будет выполнен поиск начального и конечного объектов (НО, КО) в базе объектов в целях определения соответствующих объектам координат $X_{НО}$, $Y_{НО}$ и $X_{КО}$, $Y_{КО}$. Затем по координатам НО и КО будет выполняться поиск начальных и конечных пунктов остановки в основной базе путем выполнения сравнения координат НО и КО в некотором допустимо заданном диапазоне ближайших к данным объектам координат пунктов остановок. Определение пунктов пересадки в 3-м и 4-м циклах выполняется аналогично по описанной в алгоритме схеме, но путем сравнения уже не наименований объектов, а координат пунктов остановок двух маршрутов на предмет определения некоторой близости в допустимом заданном диапазоне расстояния между пунктами пересадки, например, установленным в 50 м. При таком подходе скорость работы алгоритма увеличится, поскольку исключаются повторы наименования объектов в базе данных.

Еще один способ ускорения работы алгоритма — это преобразование структуры базы данных транспортных маршрутов в структуру графа. Ускорение работы алгоритма в этом случае будет происходить за счет того, что структура графа, в отличие от базы данных с обычным перечислением маршрутов, является более сжатой, это происходит за счет того что фрагменты транспортных маршрутов не повторяются. В этом случае каждому ребру графа может соответствовать сразу несколько маршрутов с разными номерами и типами транспорта. В базе данных в этом случае будут последовательно приводиться не сами маршруты транспорта, а фрагменты маршрутов транспорта (рис. 8), составляющие ребра графа, также разделяемые пустыми строками. Например:

№ 104: стр45, № 95: стр78, № 106: стр201

12. м. 28 мая, железнодорожный вокзал, Университет нефти и химии

13. дворец Республики, больница № 4, ул. Басина

14. пл. Физули, шахматная школа, ул. Гуси Гаджиева, МИД

При этом для того чтобы сохранить целостность работы алгоритма, необходимо в заголовке каждо-

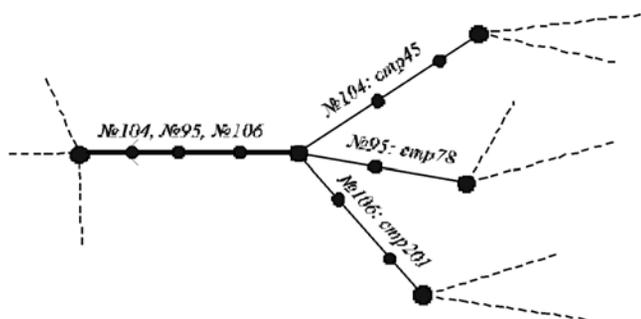


Рис. 8. Пример фрагмента графа

го фрагмента ребра графа указать все номера маршрутов транспорта, имеющих в своем маршруте данный фрагмент пути. Кроме того, для каждого номера транспорта должен быть также указан номер строки базы данных (стр:) для нового, смежного с данным ребром ребра, на котором происходит продолжение маршрута для данного номера и типа транспорта. При этом в процедуре определения пункта пересадки, выполняемого в 3-м и 4-м циклах, добавлен дополнительный цикл, который будет перебирать все номера маршрутов, указанных в оглавлении ребра, и соответственно выполнять данную процедуру аналогичным же образом для всех указанных в заголовке номеров и типов маршрутов. Переход от одного фрагмента маршрута к другому (одного ребра к другому) будет происходить по номеру строки, указанному также для каждого номера маршрута в заголовке ребра графа. Если вершина данного ребра графа не имеет смежных ребер или если маршрут транспорта на данном ребре графа завершается, то номер перехода строки в заголовке графа для данного транспорта не указывается. Соответственно, отсутствие номера строки перехода в заголовке будет приниматься алгоритмом, как завершение маршрута пути для данного номера транспорта.

Заключение

Результаты работы алгоритма могут быть представлены на вывод как в текстовом виде (см. рис. 4, б (4)), так и в графическом виде. На рис. 6 результаты работы алгоритма привязаны к информационной электронной карте г. Баку, на которой вырисовывается полученный маршрут с указанием начальных и конечных пунктов назначения и сменой маршрутов, отличающихся цветами. При выполнении привязки работы алгоритма к геоинформационным картам необходимо, чтобы объекты маршрутов, а также их данные, используемые в базе данных, были соразмерны с наименованиями и данными объектов в информационных картах. Описанный алгоритм определения кратчайшего пути применительно к автобусным маршрутам г. Баку (см. рис. 4, б) был представлен в 2008 г. на рассмотрение в Министерство транспорта Азербайджанской Республики. В результате данный алгоритм был включен в общий пакет реализации систем интеллектуального управления транспортом г. Баку. В настоящее время реализация алгоритма определения кратчайшего маршрута с использованием одной пересадки представлена на сайте Центра интеллектуального управления транспортом Азербайджанской Республики [15] (см. рис. 6), а также на терминалах, установленных на автобусных остановках на центральных улицах города.

Предложенный алгоритм может быть использован не только применительно к задаче определения маршрутов общественного транспорта, но и для лю-

бых других задач, где имеются предварительно выделенные наборы маршрутов. Кроме того, алгоритм может быть использован и в задачах определения кратчайшего пути, где нет выделенных наборов маршрутов. В этом случае будет необходимо предварительно выбрать или создать набор маршрутов (например, случайным образом), к которым затем будет применен алгоритм. При таком подходе близость полученных маршрутов к кратчайшему маршруту, при, будет определяться и зависеть от способа и подходов в выборе набора маршрутов.

Список литературы

1. **Кормен Т. Х., Лейзерсон Ч. И., Ривест Р. Л., Штайн К.** Алгоритмы: построение и анализ = Introduction to Algorithms. 2-е изд. М.: Вильямс, 2006. 1296 с.
2. **Кузнецов О. П., Адельсон-Вельский Г. М.** Дискретная математика для инженера. М.: Энергоатомиздат, 1988. 480 с.
3. **Оре О.** Теория графов. М.: Наука, 1980. 336 с.
4. **Dijkstra E. W.** A note on two problems in connexion with graphs // *Numerische Mathematik*. 1959. Vol. 1. P. 269–271
5. **Moysen F., Manderick B.** The collective behaviour of Ants: an Example of Self-Organization in Massive Parallelism // *Actes de AAAI Spring Symposium on Parallel Models of Intelligence*, Stanford, California, 1988.
6. **Dorigo M., Maniezzo V., Colorni A.** Ant System: Optimization by a Colony of Cooperating Agents // *IEEE Transactions on Systems, Man, and Cybernetics-Part B*. 1996. 26 (1): P. 29–41.

7. **Cauvery N. K., Viswanatha K. V.** Routing in Dynamic Network using Ants and Genetic Algorithm // *International Journal of Computer Science and Network Security*. 2000. Vol. 9, no. 3. P. 36–41.
8. **Казakov П. В.** Использование дифференциальной эволюции при определении множества Парето генетическими алгоритмами многокритериальной оптимизации // *Информационные технологии*. 2015. Т. 21, № 2. С. 109–116.
9. **Аллилуева Н. В.** Применение генетических алгоритмов решения задачи маршрутизации беспилотных летательных аппаратов // *Вопросы радиоэлектроники*. 2016. № 1. С. 47–53.
10. **Лунин Д. В., Скворцов С. В.** Разработка параллельного генетического алгоритма для решения задачи коммивояжера на платформе cuda // *Системы управления и информационные технологии*. 2015. Т. 60, № 2. С. 50–55.
11. **Новиков А. К.** Применение муравьиного алгоритма в задачах маршрутизации транспорта // *Молодежный научно-технический вестник*. 2015. № 11. С. 32.
12. **Кирилина А. С., Тарасова Л. Г., Казакова А. Е., Видная К. А.** Применение роевых алгоритмов в решении транспортно-экспедиционных задач // *Актуальные направления научных исследований XXI века: теория и практика*. 2015. Т. 3. № 7–4 (18–4). С. 296–300.
13. **Longley P. A., Goodchild M. F., Maguire D. J., Rhind D. W.** *Geographic Information Systems and Science*. 2nd edn. John Wiley and Sons, 2005. 454 p.
14. **Gade K.** A non-singular horizontal position representation // *The Journal of Navigation*. 2010. Vol. 63. P. 395–417.
15. **Центр Интеллектуального Управления Транспорта Азербайджанской Республики.** (Nəqliyyat İntellektual İdarəetmə Mərkəzi) <http://www.niim.az/marsrut-secimi>

P. Sh. Geidarov, PhD, Associate Professor,

Institute of Azerbaijan National Academy of Sciences of the System Management

Algorithm for the Shortest Route Based on the Selected Set of Routes

This paper describes an algorithm for determining the shortest transport route based on the selected set of routes. The algorithm is considered in relation to public transport by successive complications and extension algorithm features. Initially described an algorithm for determining the shortest route without the use of geographical coordinates of the area and with one change of transport. On the basis of the underlying algorithm examines the algorithm with two or more transfers. Also describe the possibilities of expanding the functionality of the algorithm, applied to different transport, with the possibilities of geo-referenced location. Also give a description of software and structural means to speed up the algorithm.

Keywords: shortest route, urban transport, public transport, transport in Baku, Intelligent Transport Management Center

References

1. **Cormen T. H., Leiserson C. E., Rivest R. L., Stein C. T.** *Algoritmy: postroenie i analiz = Introduction to Algorithms*, 2nd edn. Moscow: Vilyams, 2006, 1296 p.
2. **Kuznecov O. P., Adelson-Velskiy G. M.** *Diskretnaya matematika dlya inzhenera*, Moscow: Energoatomizdat, 1988, 480 p.
3. **Ore O.** *Teoriya grafov*, Moscow: Nauka, 1980, 336 p.
4. **Dijkstra E. W.** A note on two problems in connexion with graphs, *Numerische Mathematik*, 1959, vol. 1, pp. 269–271.
5. **Moysen F., Manderick B.** The collective behaviour of Ants: an Example of Self-Organization in Massive Parallelism, *Actes de AAAI Spring Symposium on Parallel Models of Intelligence*, Stanford, California, 1988.
6. **Dorigo M., Maniezzo V., Colorni A.** Ant System: Optimization by a Colony of Cooperating Agents, *IEEE Transactions on Systems, Man, and Cybernetics-Part B*, 1996, vol. 26 (1), pp. 29–41.
7. **Cauvery N. K., Viswanatha K. V.** Routing in Dynamic Network using Ants and Genetic Algorithm, *International Journal of Computer Science and Network Security*, 2000, vol. 9, no. 3, pp. 36–41.
8. **Kazakov P. V.** Ispolzovanie differentsialnoy evolyucii pri opredelenii mnojestva Pareto geneticheskimi algoritmami mnogokriterialnoy optimizacii, *Informacionnye tekhnologii*, 2015, vol. 21, no. 2, pp. 109–116.

9. **Allilueva N. V.** Primenenie geneticheskikh algoritmov resheniya zadachi marshrutizacii bespilotnykh letatelnykh apparatov, *Voprosy radioelektroniki*, 2016, no. 1, pp. 47–53.
10. **Lunin D. V., Skvortsov S. V.** Razrabotka parallelnogo geneticheskogo algoritma dlya resheniya zadachi kommivoyajera na platforme cuda, *Sistemy upravleniya i informacionnye tekhnologii*, 2015, vol. 60, no. 2, pp. 50–55.
11. **Novikov A. K.** Primenenie muravinogo algoritma v zadachah marshrutizacii transporta, *Molodejnyy nauchno-tekhnicheskij vestnik*, 2015, no. 11, pp. 32.
12. **Kirilina A. S., Tarasova L. G., Kazakova A. E., Vidnaya K. A.** Primenenie roevykh algoritmov v reshenii transportno-ekspeditsionnykh zadach, *Aktualnye napravleniya nauchnykh issledovaniy XXI veka: teoriya i praktika*, 2015, vol. 3, no. 7–4 (18–4), pp. 296–300.
13. **Longley P. A., Goodchild M. F., Maguire D. J., Rhind D. W.** *Geographic Information, Systems and Science*. John Wiley and Sons, 2nd edn. 2005, 454 p.
14. **Gade K.** A non-singular horizontal position representation, *The Journal of Navigation*, vol. 63, 2010, pp. 395–417.
15. **Центр Интеллектуального Управления Транспорта Азербайджанской Республики.** (Nəqliyyat İntellektual İdarəetmə Mərkəzi), URL: <http://www.niim.az/marsrut-secimi>

Г. А. Мельников, аспирант, Т. А. Мельников, магистрант, **В. В. Губарев**, д-р техн. наук, проф.,
Новосибирский государственный технический университет, г. Новосибирск

Алгоритмы упрощения деревьев регрессии: обзор и эмпирическое сравнение

Дан обзор и выполнена систематизация существующих алгоритмов упрощения деревьев регрессии. Также проведено эмпирическое сравнение пяти ключевых алгоритмов упрощения по трем показателям: время работы, адекватность полученных моделей и их сложность. Результаты экспериментов показывают, что в отличие от деревьев классификации, где себя хорошо зарекомендовали алгоритмы упрощения на основе отсечения ветвей, для деревьев регрессии более предпочтительны алгоритмы ранней остановки. Последние значительно менее трудоемки и строят модели, в большинстве случаев обладающие лучшей адекватностью при сопоставимой сложности.

Ключевые слова: интеллектуальный анализ данных, машинное обучение, нелинейная регрессия, кусочно-заданные модели, деревья моделей, деревья регрессии, упрощение деревьев регрессии

Введение

Деревья регрессии являются одним из важных классов регрессионных моделей, позволяющим представить кусочно-заданную функцию регрессии в интуитивно понятной и наглядной форме в виде дерева принятия решений. В таком дереве внутренние узлы содержат правила разделения пространства объясняющих переменных X , дуги — условия перехода по ним, а листья — локальные регрессионные модели. Впервые этот подход был реализован в алгоритме AID [1] в 1963 г. Однако популярны деревья регрессии стали лишь после публикации в 1984 г. работы [2] и появления алгоритма CART, который и по сей день входит практически в любой пакет статистического анализа данных.

Задача построения деревьев регрессии является NP -сложной [3]. При ее решении необходимо ответить на три основных вопроса:

- Каким образом разделить данные на сегменты?
- Как и какого типа локальные регрессионные модели строить в листьях дерева?
- Каким должен быть размер дерева регрессии?

Исследования в рассматриваемой области главным образом направлены на поиск новых правил разделения данных. Большинство алгоритмов в листьях дерева строят линейные локальные модели (видимо вследствие их простоты и наглядности) и лишь некоторые — константы или полиномы. Третий вопрос хоть и затрагивается во всех работах, но специально ему внимания практически не уделяется.

Разделять пространство объясняющих переменных X можно до тех пор, пока в узле останется лишь один обучающий пример (элемент обучающей выборки). Однако с уменьшением числа примеров в узлах результат становится менее статистически значим, модель начинает "вбирать" в себя шумы, присутствующие в данных. Такая чрезмерно точная настройка модели на обучающую выборку,

как правило, приводит к переобучению (рис. 1). Помимо переобучения, с увеличением размера деревьев регрессии теряется одно из главных их достоинств — простота интерпретации. Вследствие изложенных выше причин задача упрощения дерева регрессии, т.е. задача выбора "правильного" размера дерева, встает особенно остро.

Многие исследователи в данной области используют алгоритмы упрощения деревьев классификации. Некоторые предлагают свои, но их эффективность оценить сложно независимо от введенных ими правил разделения данных. Здесь можно выделить лишь одну работу [4], которая полностью посвящена эмпирическому сравнению алгоритмов упрощения деревьев регрессии. Однако в ней рассматриваются только три алгоритма, два из которых предложены непосредственно авторами статьи, а такой популярный алгоритм, как отсечение ветвей на основе оценки цена—сложность проигнорирован. Это, по нашему мнению, является существенным упущением. Кроме того, в последние годы появились новые алгоритмы ранней остановки [5, 6], которые гипотетически могут существенно снизить трудоемкость построения деревьев регрессии.



Рис. 1. Иллюстрация явления переобучения

Цель данной работы — обзор, систематизация и сравнение современных алгоритмов упрощения деревьев регрессии, а также определение их роли и значимости при построении деревьев регрессии.

Обзор алгоритмов упрощения деревьев регрессии

Можно выделить два основных подхода к упрощению деревьев регрессии:

- отсечение ветвей, которое предполагает построение дерева регрессии максимального размера, затем упрощается снизу вверх путем преобразования узлов в листья;
- ранняя остановка, которая предполагает ограничение роста дерева регрессии на этапе его построения путем прекращения разделения данных при достижении некоторого условия.

Отсечение ветвей. Подход на основе отсечения ветвей хорошо зарекомендовал себя при построении деревьев классификации [7] и был использован в большинстве ранних алгоритмов построения деревьев регрессии [2, 8–10]. Здесь можно выделить три основных алгоритма: упрощение на основе показателя цена—сложность (Cost-Complexity Pruning) [8], алгоритмы упрощения из M5 [9] и RETIS [10]. Как было указано ранее, во всех этих алгоритмах сначала строится дерево регрессии максимального размера, а затем уже осуществляется его упрощение.

Упрощение на основе показателя цена—сложность происходит в два этапа. На первом из полного (максимального размера) дерева регрессии строится последовательность деревьев регрессии уменьшающегося размера, минимизирующая при различных значениях параметра γ значение показателя цена—сложность*:

$$R(T, \gamma) = R(T) + \gamma|T|, \quad (1)$$

где $R(T)$ — средняя квадратическая ошибка дерева регрессии T ; $|T|$ — число листьев дерева или штраф за сложность дерева; γ — параметр регуляризации, контролирующей вклад штрафа в общую оценку. На втором этапе адекватность деревьев регрессии оценивается на дополнительной независимой (валидационной) выборке, и дерево регрессии с наименьшей средней квадратической ошибкой и соответствующее ему значение параметра $\gamma = \gamma^*$ выбираются как итоговые.

Оптимальное значение параметра $\gamma = \gamma^*$ может быть оценено с помощью перекрестной проверки. После чего полное дерево регрессии строится уже на всех обучающих данных и из него выбирается поддереву, минимизирующее (1) при $\gamma = \gamma^*$. В этом случае необходимость в дополнительной независи-

* Детали процесса нахождения последовательности значений параметра γ и соответствующих им деревьев регрессии приведены в [2].

мой выборке данных пропадает, однако возрастает трудоемкость.

В алгоритме M5 для каждого узла на обучающем множестве вычисляется средняя квадратическая ошибка локальной модели. Чтобы учесть сложность модели и число обучающих примеров, по которым она была построена, значение ошибки умножается на поправочный коэффициент:

$$k = \frac{n + mv}{n - v}, \quad (2)$$

где n — число обучающих примеров в узле; v — число параметров модели; m — свободный параметр, контролируемый пользователем. Затем при движении снизу вверх узлы преобразовываются в листья до тех пор, пока скорректированная ошибка уменьшается.

В RETIS алгоритм упрощения деревьев регрессии похож на предыдущий, но скорректированная ошибка для каждого узла вычисляется на основе байесовского подхода, который комбинирует априорные и апостериорные знания. В этом случае показатель адекватности модели ϕ определяется как

$$adjR(\phi) = \frac{m}{n + m} R_{\alpha}(\phi) + \frac{n}{n + m} R_{\phi}, \quad (3)$$

где n — число обучающих примеров в узле; $R_{\alpha}(\phi)$ — средняя квадратическая ошибка модели ϕ , вычисленная на всех обучающих примерах (а не только в текущем узле); R_{ϕ} — средняя квадратическая ошибка модели ϕ , вычисленная на обучающих примерах в текущем узле; m — контролируемый пользователем свободный параметр.

Ранняя остановка. Отсечение ветвей значительно более трудоемко, чем ранняя остановка, так как требует построения полного дерева регрессии. Поэтому работы последних лет сосредоточены в основном на использовании второго подхода — ранней остановки [5, 6, 11, 12].

Самым распространенным [2, 5, 6, 8–12] правилом ранней остановки является ограничение на минимальное число примеров в узле. Практика использования деревьев классификации показала, что данное правило не робастно [7]. И хотя его используют практически все алгоритмы, обычно его применяют совместно с другими правилами упрощения деревьев.

Большинство правил ранней остановки связаны с оценкой адекватности модели на ранее неизвестных данных. Согласно [4–6, 11, 12] здесь можно выделить следующие два основных подхода.

1. Использование *валидационного множества* [4, 12]. При очередном расщеплении узла на независимой выборке сравниваются средние квадратические ошибки моделей до и после разделения данных. Если средняя квадратическая ошибка увеличилась, то разделение отменяется и алгоритм индукции останавливается.

2. Оценка адекватности модели непосредственно из ее характеристик лишь на обучающем множестве [5, 6, 11]. Главным образом, это использование статистических тестов. Потенциально здесь возможно использование информационных критериев выбора моделей (Колмогоровская сложность).

Рассмотрим второй подход более детально. В работе [5] проверяется статистическая гипотеза о том, что все данные в узле порождены некоторым скрытым линейным процессом. Пусть есть три линейных модели: одна построена на всех N обучающих примерах, вторая — на N_L примерах, лежащих слева от точки разделения, и третья — на N_R примерах, лежащих справа. Тогда, если альтернативная гипотеза верна, то сумма квадратов остатков после разделения данных должна быть значительно меньше суммы квадратов остатков до разделения. Это может быть проверено с помощью *теста Чоу*:

$$F = \frac{(RSS - RSS_L - RSS_R)(N - 2d)}{(RSS_L + RSS_R)d}, \quad (4)$$

где F — это F -статистика с d и $N - 2d$ степенями свободы; d — число объясняющих переменных; RSS , RSS_L и RSS_R — сумма квадратов остатков модели до разделения данных и после соответственно. Разделение данных должно быть осуществлено только если p -значение меньше некоторого порогового значения α (т.е. нулевая гипотеза отвергается на уровне значимости α), заданного пользователем.

Эмпирическое сравнение алгоритмов упрощения деревьев регрессии

Рассматриваемые алгоритмы упрощения деревьев регрессии были протестированы на двух синтетических наборах данных из работы [13] и восьми наборах данных из UC Irvine Machine Learning Repository [14] и KEEL-dataset repository [15]. Их краткая характеристика представлена в табл. 1.

Чтобы поставить все тестируемые алгоритмы упрощения в равные условия, в каждом случае использовался один и тот же жадный метод построения деревьев регрессии на основе рекурсивного разделения данных. На текущий момент он явля-

ется самым распространенным методом построения деревьев регрессии. Кратко его можно описать следующим образом.

1. Выбор "лучшего" разделения данных C (объясняющей переменной и точки разделения a или разделяющего множества A), как правило, такого, которое обеспечивает экстремум некоторого критерия R .

2. Разделение данных на подмножества.

3. Рекурсивное применение шагов 1—3 к каждому из подмножеств.

В качестве критерия выбора модельного разделения данных была использована минимизация взвешенной суммы квадратов отклонений локальных моделей:

$$R(T, C) = \frac{N_L}{N} \sum_{i \in I_L} (y_i - g_L(x_i))^2 + \frac{N_R}{N} \sum_{i \in I_R} (y_i - g_R(x_i))^2, \quad (5)$$

где T — исходный набор данных; T_L и T_R — наборы, образованные путем разделения T по C (т. е. по a или A); N , N_L и N_R — число элементов в каждом из наборов; I_L и I_R — индексы принадлежности элементов к T_L и T_R соответственно; g_L и g_R — локальные модели для T_L и T_R соответственно. Данный критерий является довольно трудоемким, поэтому для каждой переменной мы проверяли лишь 20 разделений равномерно распределенных на области ее значений. В качестве локальных моделей была использована множественная линейная регрессия. Ее построение осуществлялось с помощью алгоритма пошаговой регрессии с использованием Байесовского информационного критерия для отбора переменных. Для всех алгоритмов упрощения минимальное число примеров в листе было установлено равным 5 % от числа обучающих данных. Значение свободного параметра m алгоритмов упрощения из M5 и RETIS в обоих случаях было установлено равным 2, согласно рекомендациям [9] и [10], соответственно. Для алгоритмов упрощения на основе оценки цена—сложность и ранней остановки по ошибке на валидационном множестве в качестве валидационного множества использовалось 30 % от обучающей выборки.

Все алгоритмы были реализованы на языке программирования Matlab и протестированы в среде Matlab R2007b на ПК со следующей конфигурацией: Intel Core 2 Duo E6600 и 4 GB RAM.

Результаты эксперимента приведены в табл. 2. Они получены с помощью 10-слойной перекрестной проверки и усреднены по 30 запускам. В качестве показателя адекватности полученных моделей был использован квадратный корень из средней квадратической ошибки (RMSE). Для оценки сложности моделей использовали число вершин в дереве регрессии. В табл. 2 также приведено время построения (в секундах) дерева регрессии.

Для всех значений в табл. 2 приведено среднее квадратическое отклонение. Однако интервальные

Таблица 1

Краткая характеристика рассматриваемых наборов данных

Название набора данных	Объем выборки	Число объясняющих переменных
Armchair	1000	2
Split plane	1000	1
Abalone	4177	8
Ailerons	13 750	40
Auto-mpg	392	7
CPU	209	6
Housing	506	14
Stock	950	10
Breast Cancer Wisconsin	198	34
Triazines	186	61

Эмпирическое сравнение алгоритмов упрощения деревьев регрессии

Данные	Показатели	Алгоритмы				
		M5	RETIS	Цена—сложность	Валидационное множество	Тест Чоу
ArmChar	RMSE	0,47 ± 0,06	0,25 ± 0,059	0,18 ± 0,028	0,19 ± 0,042	0,15 ± 0,016
	Сложность	19,6 ± 1,7	13,1 ± 1,1	16,5 ± 0,9	17,5 ± 0,8	15,9 ± 0,6
	Время, с	0,39 ± 0,01	0,29 ± 0,02	0,44 ± 0,06	0,25 ± 0,01	0,27 ± 0,02
SplitPlane	RMSE	0,002 ± 0,003	0,003 ± 0,003	0,002 ± 0,002	0,002 ± 0,002	0,003 ± 0,003
	Сложность	4,5 ± 0,4	4,29 ± 0,56	4,49 ± 0,51	4,4 ± 0,59	4,27 ± 0,56
	Время, с	0,24 ± 0,02	0,05 ± 0,01	0,06 ± 0,01	0,05 ± 0,01	0,05 ± 0,01
Abalone	RMSE	2,17 ± 0,04	2,16 ± 0,02	2,22 ± 0,02	2,16 ± 0,01	2,15 ± 0,02
	Сложность	28,1 ± 0,7	27,6 ± 0,5	6,5 ± 2,8	7,4 ± 1,3	19,4 ± 0,7
	Время, с	7,6 ± 0,1	9,0 ± 0,2	10,4 ± 0,2	5,3 ± 0,5	9,0 ± 0,3
Ailerons	RMSE	0,000162 ± 0,0	0,000162 ± 0,0	0,000165 ± 0,0	0,000163 ± 0,0	0,000163 ± 0,0
	Сложность	27,4 ± 0,6	28,4 ± 0,6	10,0 ± 2,5	14,7 ± 1,3	18,4 ± 0,9
	Время, с	104,1 ± 4,4	129,6 ± 14,7	129,3 ± 6,9	90,3 ± 4,7	124,5 ± 12,8
Auto-mpg	RMSE	3,26 ± 0,16	3,23 ± 0,16	3,35 ± 0,09	3,11 ± 0,1	3,22 ± 0,15
	Сложность	30,5 ± 0,6	25,6 ± 1,0	4,4 ± 2,8	5,9 ± 1,1	19,5 ± 1,1
	Время, с	4,15 ± 0,1	3,9 ± 0,1	5,1 ± 0,3	1,93 ± 0,21	4,0 ± 0,4
Housing	RMSE	4,82 ± 1,64	4,90 ± 1,53	5,08 ± 1,76	4,78 ± 1,06	4,93 ± 1,66
	Сложность	30,5 ± 0,5	20,9 ± 1,3	5,87 ± 2,2	7,3 ± 1,2	20,3 ± 0,8
	Время, с	27,5 ± 2,0	26,3 ± 0,9	30,3 ± 0,9	13,6 ± 1,6	28,5 ± 0,8
Machine	RMSE	68,84 ± 38,54	55,11 ± 6,56	61,17 ± 9,33	56,68 ± 7,61	56,48 ± 6,67
	Сложность	30,4 ± 0,5	17,4 ± 1,3	2,65 ± 0,8	4,0 ± 0,9	16,7 ± 1,1
	Время, с	2,9 ± 0,1	3,0 ± 0,1	3,6 ± 0,1	1,1 ± 0,1	2,9 ± 0,1
Stock	RMSE	0,89 ± 0,05	0,90 ± 0,04	1,06 ± 0,09	1,05 ± 0,08	0,88 ± 0,037
	Сложность	29,9 ± 0,4	22,8 ± 0,9	16,9 ± 2,3	19,9 ± 1,4	29,8 ± 0,6
	Время, с	13,4 ± 0,3	14,5 ± 0,5	17,1 ± 0,2	12,7 ± 0,4	15,8 ± 0,4
Breast	RMSE	77,21 ± 7,92	65,96 ± 3,69	30,59 ± 0,8	31,1 ± 1,21	30,33 ± 0,44
	Сложность	29,5 ± 0,6	21,1 ± 0,9	1,0 ± 0,0	1,1 ± 0,1	1,0 ± 0,0
	Время, с	320,0 ± 30,2	436,4 ± 90,6	370,4 ± 47,7	23,3 ± 2,1	37,0 ± 3,5
Triazine	RMSE	0,54 ± 0,76	0,17 ± 0,02	0,16 ± 0,03	0,16 ± 0,03	0,14 ± 0,0
	Сложность	28,9 ± 0,7	19,8 ± 1,1	2,0 ± 1,0	3,2 ± 0,7	1,0 ± 0,0
	Время, с	24,5 ± 0,6	23,2 ± 1,0	24,3 ± 0,9	8,9 ± 1,3	4,9 ± 0,2

оценки очень трудны для анализа и не наглядны. Поэтому для сравнения алгоритмов по показателю RMSE, значения которого сравнительно близки у различных алгоритмов, применяли статистические тесты и все выводы осуществляли на их основе. Сложность же получаемых моделей и время исполнения сильно различаются между сравниваемыми группами алгоритмов. При этом среднее квадратическое отклонение для этих показателей обычно составляет менее 10 % от их значения. Поэтому для их сравнения достаточно проанализировать лишь средние значения.

Чтобы выделить лишь статистически значимые различия по показателю RMSE, попарно сравним исследуемые алгоритмы с помощью непараметрического статистического теста Уилкоксона [16] (табл. 3). Согласно табл. 3 алгоритм упрощения на основе теста Чоу строит модели, обладающие наименьшей ошибкой аппроксимации данных (в 22 случаях из 40 наблюдаются значимо меньшие значения показателя RMSE). Далее практически с одинаковыми результатами идут алгоритм ранней остановки по ошибке на валидационном множестве и

Таблица 3

Сравнение RMSE моделей с помощью теста Уилкоксона ($\alpha = 0,05$)

Алгоритм	M5	RETIS	Цена—сложность	Валидационное множество	Тест Чоу	Всего побед
M5	—	0	4	2	1	7
RETIS	4	—	5	2	1	12
Цена—сложность	3	3	—	0	0	6
Валидационное множество	5	4	4	—	1	14
Тест Чоу	5	5	7	5	—	22
Всего поражений	17	12	20	9	3	

Примечание. Под победой здесь понимается статистически значимое меньшее среднее значение показателя RMSE, а под поражением, соответственно статистически значимое большее

алгоритм упрощения на основе Байесовского подхода (14 и 12 побед, соответственно). Аутсайдерами здесь являются алгоритм упрощения на основе оценки цена—сложность и алгоритм M5 (7 и 6 побед, соответственно). Таким образом, когда интерес представляют исключительно прогностические свойства моделей, то безусловным фаворитом здесь является алгоритм ранней остановки на основе теста Чоу.

Вместе с тем с помощью алгоритма упрощения на основе оценки цена—сложность создаются наиболее простые модели (рис. 2). Они в среднем в 3,5 раза проще, чем модели, полученные с помощью алгоритма упрощения M5; в 3 раза проще, чем при упрощении на основе Байесовского подхода; в 2 раза проще, чем при использовании теста Чоу; и в среднем имеют размер на 1,5 узла меньше, чем при использовании ранней остановки по ошибке на валидационном множестве. Поэтому здесь необходимо понять: значимо ли уменьшение RMSE при наблюдаемом увеличении сложности модели? Для этого можно воспользоваться F-тестом [17], применив его ко всему дереву регрессии на тестовых данных (табл. 4). Из табл. 4 видно, что при использовании алгоритма упрощения на основе теста Чоу рост сложности моделей по сравнению с алгоритмами упрощения на основе оценки цена—сложность и валидационного множества оправдан лишь в трех из восьми случаев и на двух наборах данных были получены более компактные модели, обладаю-

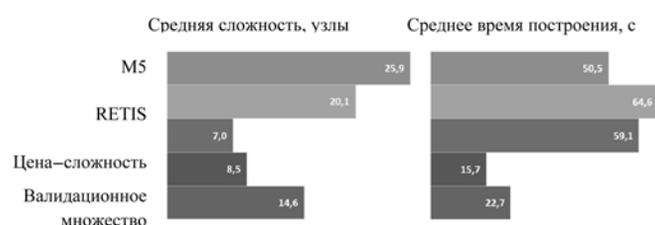


Рис. 2. Усредненные по всем наборам данных значения показателей

Таблица 4
Сравнение моделей с помощью F-теста ($\alpha = 0,05$)

Алгоритм	M5	RETIS	Цена—сложность	Валидационное множество	Тест Чоу	Всего побед
M5	—	2	4	2	1	9
RETIS	8	—	3	2	1	14
Цена—сложность	6	7	—	5	5	23
Валидационное множество	8	8	5	—	5	26
Тест Чоу	9	9	5	5	—	28
Всего поражений	31	26	17	14	12	

Примечание. Под победой здесь понимается, что модель согласно F-тесту статистически значимо лучше; в обратном случае имеет место поражение.

щие меньшей ошибкой. Поэтому последние два алгоритма можно рекомендовать, когда необходимо получить простые и интерпретируемые модели.

Если же сравнивать скорость работы алгоритмов упрощения деревьев регрессии, то самым быстрым вариантом является ранняя остановка по ошибке на валидационном множестве. Данный алгоритм в среднем в 1,4—4 раза быстрее остальных алгоритмов. На втором месте идет алгоритм ранней остановки на основе теста Чоу, в среднем проигрывающий первому лишь 7 с. Остальные три алгоритма упрощения по времени работают примерно одинаково. В среднем они приблизительно в 3 раза медленнее алгоритмов ранней остановки.

Анализируя рис. 2 и табл. 3, заключаем, что алгоритмы упрощения деревьев регрессии из M5 и RETIS строят неоправданно сложные модели. При этом они не дают никаких конкурентных преимуществ по сравнению с другими алгоритмами ни по точностным характеристикам, ни по скорости работы. Но здесь необходимо отметить, что данные алгоритмы имеют настраиваемые параметры, регулирующие их работу, влияние которых мы никак не рассматривали. И при значительных дополнительных временных затратах эти алгоритмы можно настроить под каждый из рассматриваемых наборов данных.

Заключение

В работе, во-первых, выполнены обзор и систематизация существующих алгоритмов упрощения деревьев регрессии. Во-вторых, проведено эмпирическое сравнение пяти ключевых алгоритмов: M5, RETIS, упрощения на основе показателя цена—сложность, ранней остановки по ошибке на валидационном множестве и на основе теста Чоу. По результатам экспериментов сделаны следующие выводы.

1. Упрощение деревьев регрессии является одним из ключевых шагов как для получения простых и интерпретируемых моделей, так и для достижения высоких точностных характеристик.

2. В отличие от деревьев классификации, где себя хорошо зарекомендовали алгоритмы упрощения на основе отсечения ветвей, для деревьев регрессии более предпочтительны алгоритмы ранней остановки. На рассмотренных наборах данных:

- а) последние в среднем в 3 раза быстрее;
- б) ранняя остановка по тесту Чоу чаще остальных алгоритмов приводит к получению моделей с наименьшей ошибкой аппроксимации данных;

в) ранняя остановка по ошибке на валидационном множестве строит деревья регрессии в среднем лишь на 1,5 узла больше самых компактных моделей, а по ошибке аппроксимации данных идет сразу за упрощением по тесту Чоу.

3. Алгоритмы упрощения на основе отсечения ветвей M5 и RETIS строят неоправданно сложные

модели и при этом не дают никаких конкурентных преимуществ.

4. При отсечении ветвей на основе оценки цена—сложность и ранняя остановка по ошибке на валидационном множестве строятся самые компактные деревья регрессии. При этом почти на половине исследуемых наборов данных рост сложности, получаемый при использовании теста Чоу, не оправдан. Поэтому эти два алгоритма можно рекомендовать, когда необходимо получить простые интерпретируемые модели. А так как ранняя остановка по ошибке на валидационном множестве почти в 4 раза быстрее, то ее стоит предпочесть отсечению ветвей на основе оценки цена—сложность.

5. Ранняя остановка по тесту Чоу чаще остальных алгоритмов приводит к получению моделей с наименьшей ошибкой аппроксимации данных, но по сравнению с отсечением ветвей на основе оценки цена—сложность и ранней остановкой по ошибке на валидационном множестве, наблюдается двукратное усложнение моделей, что не всегда оправдано. Поэтому данный алгоритм можно рекомендовать, в первую очередь, когда интересны прогностические свойства получаемых моделей.

Список литературы

1. **Morgan J. N., Sonquist J. A.** Problems in the analysis of survey data, and a proposal // J. Amer. Statist. Assoc., 1963. N. 58. P. 415—434.
2. **Breiman L., Friedman J. H., Olshen R. A., Stone C. J.** Classification and Regression Trees. Belmont: Wadsworth International Group, 1984. 259 p.
3. **Hyafil L., Rivest R. L.** Constructing optimal binary decision trees is NP-complete // Information Processing Letters. 1976. N. 5 (1). P. 15—17.
4. **Torgo L.** A Comparative Study of Reliable Error Estimators for Pruning Regression Trees // In Proceeding of the Iberoamerican Conference on Artificial Intelligence. Springer-Verlag, Coelho, 1998.
5. **Potts D., Sammut C.** Incremental Learning of Linear Model Trees // Machine Learning. 2005. Vol. 61. P. 5—48.
6. **Vogel D., Asparouhov O., Scheffer T.** Scalable look-ahead linear regression trees // Proc. of 13th ACM SIGKDD. New York: ACM Press, 2007. P. 757—764.
7. **Sreerama K. Murthy.** Automatic Construction of Decision Trees from Data: A Multi-Disciplinary Survey // Data Mining and Knowledge Discovery. 2005. Vol. 2, N. 4. P. 345—389.
8. **Loh W.-Y.** Regression trees with unbiased variable selection and interaction detection // Statistica Sinica. 2002. Vol. 12. P. 361—386.
9. **Quinlan J. R.** Learning with continuous classes // Proc. AI'92, 5th Australian Joint Conference on Artificial Intelligence. Singapore: World Scientific, 1992. P. 343—348.
10. **Karalic A.** Employing linear regression in regression tree leaves // Proc. of the 10th European Conference on Artificial Intelligence / Ed. B. Neumann. Vienna: Wiley, 1992. P. 440—441.
11. **Malerba D., Esposito F., Ceci M., Appice A.** Top-down induction of model trees with regression and splitting nodes // IEEE Transactions on Pattern Analysis and Machine Intelligence. 2004. N. 26. P. 612—625.
12. **Melnikov G. A., Gubarev V. V.** Ant Colony Based Semi-Greedy Algorithm for Regression Tree Induction // Proc. of the 8-th International forum on strategic technology 2013, (IFOST 2013), Mongolia, Ulaanbaatar, 28 June — 1 July 2013. Ulaanbaatar, 2013. Vol. II. P. 238—240.
13. **Мельников Г. А.** Применение методов искусственного интеллекта для исследования инфекционных заболеваний: Магистерская дис. ... "Магистр техники и технологии": 230100. Новосибирск, 2012. 141 с.
14. **Frank A., Asuncion A.** UCI Machine Learning Repository. Irvine, CA: University of California, School of Information and Computer Science, 2010. [http://archive.ics.uci.edu/ml]
15. **Alcalá-Fdez J., Fernández A., Luengo J., Derrac J., García S., Sánchez L., Herrera F.** KEEL Data-Mining Software Tool: Data Set Repository, Integration of Algorithms and Experimental Analysis Framework // Journal of Multiple-Valued Logic and Soft Computing. 2011. Vol. 17. P. 255.
16. **Wilcoxon F.** Individual comparisons by ranking methods // Biometrics. International Biometric Society. 1945. Vol. 1, N. 6. P. 80—83.
17. **Green W. H.** Econometric analysis. 5th ed. Bearson education, 2003. 1056 p.

G. A. Melnikov, Postgraduate student, grmel89@gmail.com,

T. A. Melnikov, Master student, temmelnik@gmail.com,

V. V. Gubarev, PhD, Professor of Department of Computer Engineering
Novosibirsk State Technical University

Regression Tree Pruning Algorithms: an Overview and Empirical Comparison

Regression trees belong to a very important class of regression models which allows to split feature space into segments with building specialized local model for each of them and to achieve visualizable, easy interpretable and accurate piecewise models. As for the classification tree, choosing the right size of the tree is one of the key issues of regression tree induction. Unfortunately, this issue is given very little attention. The majority of works focused on the development of data splitting algorithms.

The first part of the paper gives an overview and systematization of existing regression tree pruning algorithms. These algorithms can be divided into two standard groups: pre-pruning and post-pruning. Most authors follow the best practices of classification trees induction algorithms and use cost-complexity pruning or design their own post-pruning algorithms. There are only few works where used pre-pruning. Is worth noting here only two algorithms from this group: the first uses validation dataset to estimate generalization error and the second are based on the Chow test. In the second part of the paper, we conducted an empirical comparison of five key pruning algorithms in three indicators: running time of the algorithms, adequacy and complexity of the obtained models. The results of experiments show that, unlike the case of the classification trees, pre-pruning algorithms are more preferred to post-pruning algorithms in regression tree induction. The first are much less time-consuming, induction time decreased on average by three times. In addition, the pre-pruning algorithms induct models that in most cases has a better adequacy and complexity comparable with that of the best post-pruning models.

Future work should be focused on the development pre-pruning algorithms of regression tree induction. Of particular interest, in our opinion, should be given to adaptation of statistical tests and information criteria of model selection.

Keywords: data mining, machine learning, non-linear regression, piecewise models, model trees, regression trees, regression tree pruning

References

1. **Morgan J. N., Sonquist J. A.** Problems in the analysis of survey data, and a proposal, *J. Amer. Statist. Assoc.*, 1963, no. 58, pp. 415—434.
2. **Breiman L., Friedman J. H., Olshen R. A., Stone C. J.** *Classification and Regression Trees*, Wadsworth International Group, Belmont, 1984, 259 p.
3. **Hyafil L., Rivest R. L.** Constructing optimal binary decision trees is NP-complete, *Information Processing Letters*, 1976, no. 5 (1), pp. 15—17.
4. **Torgo L.** A Comparative Study of Reliable Error Estimators for Pruning Regression Trees, *Proceeding of the Iberoamerican Conference on Artificial Intelligence*, Springer-Verlag, Coelho, 1998.
5. **Potts D., Sammut C.** Incremental Learning of Linear Model Trees, *Machine Learning*, 2005, vol. 61, pp. 5—48.
6. **Vogel D., Asparouhov O., Scheffer T.** Scalable look-ahead linear regression trees, *Proc. of 13th ACM SIGKDD*, New York, ACM Press, 2007, pp. 757—764.
7. **Murthy Sreerama K.** Automatic Construction of Decision Trees from Data: A Multi-Disciplinary Survey, *Data Mining and Knowledge Discovery*, 2005, vol. 2, no. 4, Kluwer Academic Publishers, 2005, pp. 345—389.
8. **Loh W.-Y.** Regression trees with unbiased variable selection and interaction detection, *Statistica Sinica*, vol. 12, 2002, pp. 361—386.
9. **Quinlan J. R.** Learning with continuous classes, *Proc. AI'92, 5th Australian Joint Conference on Artificial Intelligence, Singapore, World Scientific*, 1992, pp. 343—348.
10. **Karalic A.** Employing linear regression in regression tree leaves, *Proc. of the 10th European Conference on Artificial Intelligence*, ed. B. Neumann, Vienna, Wiley, 1992, pp. 440—441.
11. **Malerba D., Esposito F., Ceci M., Appice A.** Top-down induction of model trees with regression and splitting nodes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2004, no. 26, pp. 612—625.
12. **Melnikov G. A., Gubarev V. V.** Ant Colony Based Semi-Greedy Algorithm for Regression Tree Induction, *Proc. of the 8-th International forum on strategic technology 2013, (IFOST 2013), Mongolia, Ulaanbaatar, 28 June — 1 July 2013, — Ulaanbaatar, 2013*, vol. 11, pp. 238—240.
13. **Melnikov G. A.** *Primenenie metodov iskusstvennogo intellekta dlya issledovaniya infektsionnykh zabolevaniy: masterskaya dis. ... "Magistr tekhniki i tekhnologii"*: 230100. Novosibirsk, 2012, 141 p. (Melnikov G. A. Application of artificial intelligence methods for the study of infectious diseases: Master's thesis... "Master of engineering and technology": 230100 / Melnikov Grigoriy Andreevich, Novosibirsk, 2012. 141 p.)
14. **Frank A., Asuncion A.** *UCI Machine Learning Repository* [http://archive.ics.uci.edu/ml] / Irvine, CA: University of California, School of Information and Computer Science, 2010.
15. **Alcalá-Fdez J., Fernández A., Luengo J., Derrac J., García S., Sánchez L., Herrera F.** KEEL Data-Mining Software Tool: Data Set Repository, Integration of Algorithms and Experimental Analysis Framework, *Journal of Multiple-Valued Logic and Soft Computing*, 2011, vol. 17, pp. 255.
16. **Wilcoxon F.** Individual comparisons by ranking methods, *Biometrics*, 1945, vol. 1, no. 6, International Biometric Society, pp. 80—83.
17. **Green W. H.** *Econometric analysis*, 5th ed., Bearson education, 2003, 1056 p.



VIII Всероссийская (с международным участием) научно-практическая конференция

"ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ОБРАЗОВАНИИ" ("ИТО-Саратов—2016")



2—3 ноября, г. Саратов, СГУ им. Н. Г. Чернышевского

НАПРАВЛЕНИЯ РАБОТЫ КОНФЕРЕНЦИИ:

- Цели, содержание и методика преподавания информатики и ИКТ.
- Информационные технологии в образовании: дошкольном, школьном, средне-профессиональном, высшем и дополнительном.
- Информационные технологии в работе с одаренными детьми.
- Информационная образовательная среда учебного заведения, ИКТ в управлении образованием.
- Открытое образование и дистанционное обучение.
- Проектная деятельность в информационной образовательной среде (секция для школьников и студентов).
- Информационные технологии в дополнительном образовании (переподготовка, повышение квалификации и др.)
- Опыт применения ИКТ в профессиональном образовании.

Подробную информацию о конференции см. на сайте:
<http://saratov.ito.edu.ru>

Способы представления спецификаций вычислительных систем: проблемы и возможности машинной обработки

Спецификации вычислительных систем могут состоять из множества документов с использованием различных форм описания, таких как естественные языки, диаграммы, схемы, таблицы и формальные языки. Процесс создания спецификаций может быть охарактеризован как процесс перехода от описанных на естественном языке требований к системе, не структурированных и содержащих обычно множество неточностей, к более формализованным спецификациям, не содержащим неточностей, двусмысленностей и противоречий. Описаны различные формы спецификаций, проблемы, связанные с каждой из них, а также возможности анализа и валидации спецификаций.

Ключевые слова: требования, спецификация, валидация, естественный язык, формальный язык, машинная обработка, диаграмма UML, автоматическая трансляция, таблица, шаблон таблицы

Введение

Сложность вычислительных систем многократно возросла за последние десятилетия. Они могут состоять из сотен взаимосвязанных аппаратных и программных компонентов. Все возрастающая сложность проектирования таких систем ведет к повышению важности создания их полных и точных спецификаций, в которых описано, каким образом компоненты системы взаимодействуют между собой и с окружением вычислительной системы. От качества таких спецификаций во многом зависит успех последующих этапов проектирования системы.

Процесс создания спецификаций включает в себя несколько этапов (рис. 1): выявление техни-

ческих требований к системе, их анализ, документирование, верификация и дальнейшая поддержка изменений в документации [1, 2].

Выявление технических требований к системе — это первый этап в процессе создания спецификаций, на котором необходимо определить потребности различных заинтересованных лиц — заказчиков системы и ее конечных пользователей. Этот этап особенно важен, так как часто именно в процессе выявления требований в спецификацию будущей системы могут закрадываться ошибки, противоречия и неточности. Обычно это связано с проблемами в коммуникации между системными архитекторами, с одной стороны, и заказчиками либо пользователями — с другой, с невозможностью точно описать некоторые идеи заинтересованных сторон. Такие ошибки в случае слишком позднего обнаружения или недостаточно тщательного устранения могут привести к катастрофическим для проекта последствиям [3—7]. Стоимость такой ошибки для проекта может возрасти, по некоторым данным, в десять и более раз за каждый этап разработки системы, на котором эта ошибка осталась невыявленной. В случае если такая ошибка остается невыявленной на момент выпуска продукта, последствия могут быть фатальными. Одним из известных примеров является крушение ракеты-носителя Ариан-5 4 июня 1996 г. вследствие ошибки в спецификации управляющего программного обеспечения.

Анализ требований к системе — второй этап в процессе создания спецификаций, включающий в себя проверку выявленных ранее требований к системе

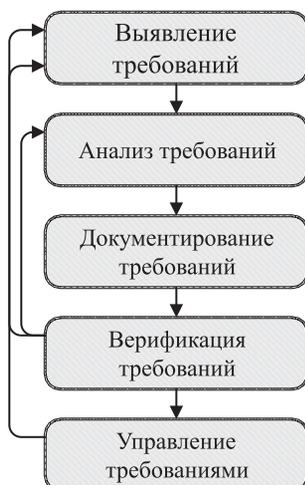


Рис. 1. Процесс создания спецификаций

на непротиворечивость, различные несоответствия, упущения и другие возможные ошибки, а также устранение таких ошибок.

На этапе *документирования требований* выявленные требования систематизируются, и на их базе создается спецификация будущей системы. Следует заметить, что на этом этапе спецификация еще не имеет окончательного вида вследствие итеративности процесса создания спецификаций. Ошибки, найденные на этапе верификации, а также изменения требований к системе требуют правок в спецификациях, после которых снова необходима верификация.

Верификация спецификаций состоит в проверке того, насколько описанная в спецификации функциональность системы соответствует ожидаемым требованиям. Кроме того, в процессе верификации спецификаций проверяется точность, полнота и непротиворечивость спецификации, а также соответствие описанных характеристик системы заявленным требованиям. Верификация спецификаций может выполняться как вручную, так и с использованием средств автоматизации, однако автоматизированная верификация может применяться лишь к спецификациям, представленным в формализованном виде. Ошибки, найденные в процессе верификации, приводят к необходимости изменения спецификации, а в некоторых случаях — и к коррекции требований к системе.

Важно отметить, что требования к вычислительным системам имеют тенденцию меняться и после создания спецификации, уже в процессе проектирования, что приводит к необходимости внесения изменений в спецификации, поддержки этих изменений и проверки их корректности.

Процесс создания спецификаций обычно состоит в переходе от полученных от заказчиков требований к формализованным спецификациям [8, 9]. Первичные требования обычно представляются в виде простого текстового описания, непрозрачны, могут включать двусмысленности и иметь множество различных трактовок. Финальная же версия спецификаций должна быть точной, корректной и недвусмысленной. Такой подход требует работы по трем направлениям:

- важно стремиться к созданию максимально полно описанной спецификации, включающей все детали будущей системы;
- необходимо добиться полного согласия заинтересованных сторон по поводу правильности и полноты спецификаций, для чего необходимо точно и недвусмысленно изложить в ней все требования к системе;
- спецификация должна быть представлена в формализованном виде для достижения однозначности изложенной в ней информации, а также для обеспечения возможности ее верификации.

В данной работе рассматривается главным образом третье направление работы над спецификациями, главная задача которого — формализация представления данных о системе. Инструменты такой формализации, методы создания формализованных спецификаций и языки описания требований могут быть подразделены на три категории [10]: объектно-ориентированные [11, 12], процесс-ориентированные [13] и поведенческие. Каждый из этих подходов имеет свои плюсы и минусы. Любого из этих методов создания и формализации спецификаций обычно недостаточно для того, чтобы полностью избежать проблем с целостностью, однозначностью, несвязностью и неточностью данных о системе.

Далее в работе будут рассмотрены различные способы представления спецификаций, классифицированные по степени формализации. Неформализованные и полностью формализованные способы представления спецификаций описаны относительно кратко, а основное внимание уделено частично формализованным способам представления спецификаций, в частности, диаграммному и табличному как наиболее интересным и перспективным с точки зрения оптимального соотношения затрат на создание и возможностей дальнейшей обработки и анализа.

Языки описания спецификаций

На сегодняшний день спецификация проекта обычно состоит из множества документов, специфицирующих отдельные аспекты работы проектируемой системы. Эти документы создаются с использованием различных языков и форматов описания поведения и характеристик системы, таких как текстовые описания на естественном языке, подразделяющиеся на неструктурированные и структурированные, различные диаграммы последовательности операций, таблицы, рисунки, а также формальные языки. При этом в процессе написания спецификации описания некоторых характеристик системы проходят определенную последовательность трансформаций, чаще всего от менее формализованной формы к более формализованной (рис. 2). К примеру, изначальные требования были оформлены в виде текста на естественном языке, затем на основе этого текста были составлены таблицы, с помощью которых на последних этапах создания спецификации был написан код на одном из языков программирования.

Тексты на естественном языке являются примером наименее формализованного описания системы и, соответственно, имеют все недостатки, связанные с неточностью, двусмысленностью и возможностью различных трактовок. Несмотря на все эти недостатки, тексты на естественных языках на ранних этапах процесса проектирования обычно составляют основную массу данных о системе. Есть несколько причин такого широкого распростране-

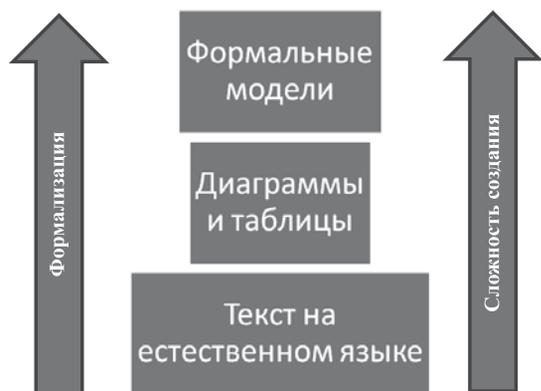


Рис. 2. Формы представления спецификаций

ния текстов на естественном языке в высокоуровневых спецификациях: простота создания, отсутствие семантических ограничений, позволяющих описывать с их помощью только отдельные аспекты системы, а также отсутствие необходимости серьезной подготовки специалистов, создающих документацию, связанной с освоением необходимых инструментов и языков.

Существенным недостатком текстов на естественном языке является сложность, а во многих случаях и невозможность автоматической трансформации их в другие, более формализованные формы спецификации. Однако существуют работы по переводу естественного языка в программный код [14—16]. Для корректной работы большинству таких переводчиков требуется правильное структурирование исходного текста и ограничения в используемой лексике, потому как задача обработки естественного языка не может быть решена с использованием существующих вычислительных систем и алгоритмов.

Формальные языки предоставляют возможность наиболее точного, лишённого проблем двусмысленности, описания системы. Обычно описания на формальном языке представляют собой строго описанные частные модели, отражающие определенные аспекты поведения будущей системы. Такие модели удобны тем, что их верификация может быть проведена автоматически, что является серьезным преимуществом в случае, когда сложность системы достаточно велика. Ряд работ описывает возможности проверки таких моделей [17—19].

Несмотря на преимущества спецификации поведения и характеристик системы с использованием формальных языков, обычно лишь небольшая часть спецификаций описана с их помощью. Это объясняется высокой сложностью освоения этих языков и создания с их помощью моделей системы, их плохой читабельностью, что приводит к невозможности быстрого обмена данными о будущей системе между заинтересованными сторонами, а также семантическими ограничениями, значительно сужающими область применения формальных языков.

Частично формализованные спецификации содержат как данные, описанные на формальных языках, так и дополнительную информацию на естественном языке. Примером такой спецификации является диаграмма последовательности операций: синтаксис и семантика диаграммы формально заданы, но подписи к объектам диаграммы на естественном языке составляют существенную часть представленных в ней данных. Такие формы документации имеют достоинства и недостатки обеих выше описанных категорий спецификаций. Более того, благодаря существующим методам автоматизации проектирования возможно нивелировать недостатки, присущие таким формам документации, вследствие чего частично формализованные формы спецификаций являются крайне интересным объектом для изучения.

Спецификации в форме диаграмм

Одной из самых распространенных и удобных форм создания частично формализованных спецификаций являются диаграммы. С их помощью можно описать и структуру будущей системы, и ее поведение. Наиболее распространенным визуальным языком создания диаграмм в сфере разработки спецификаций является UML (Unified Modelling Language), ставший практически стандартом для написания спецификаций программных систем. В рамках UML существует множество типов диаграмм: структурные диаграммы для описания строения системы, а также различные диаграммы последовательности операций для описания ее поведения. Диаграммы UML чаще всего представляют собой спецификацию высокого уровня и не содержат детально и формализованно изложенной информации, и к ним нельзя применить формальные методы анализа. Вместо этого обычно модифицируют исходные диаграммы UML, добавляя в них дополнительные данные, а затем трансформируют их в более формализованные модели [20].

Ряд работ рассматривает различные области применения диаграмм UML [21—24].

В частности, в работе [21] рассматривается процесс создания представленных в виде UML-диаграмм объектно-ориентированных моделей для спецификации систем управления данными о продукте и выявляются достоинства и недостатки таких моделей.

В статье [22] представлена методология создания спецификаций поведения киберфизических систем с использованием диаграмм UML. Для описания специфических для предметной области компонентов предлагается использовать такие механизмы расширения синтаксиса UML-диаграмм, как UML-профили и стереотипы.

В работах [23, 24] UML-диаграммы используются при создании спецификации логических контроллеров. Для обеспечения соответствия спецификации установленным пользователями требо-

ваниям к поведению контроллеров применяется формальная проверка модели, при которой требования приводятся к виду формул темпоральной логики, а диаграммы UML формализуются и переводятся в программный код.

В [25] также рассматривается интеграция инструментов формальной проверки моделей и методов спецификации, основанных на UML-диаграммах. Здесь целью является создание спецификаций встроенных систем, а в качестве примера рассматривается система управления медицинским оборудованием.

В работах [26, 27] рассматриваются возможности переиспользования UML-диаграмм в процессе разработки программного обеспечения. Будучи де-факто стандартом в этой области, UML-диаграммы используются практически повсеместно для описания программных систем, поэтому возможность их переиспользования чрезвычайно важна, так как позволяет экономить значительную часть времени работы аналитиков и архитекторов программных систем.

В статье [28] описывается исследование влияния различных факторов на читабельность и усваиваемость моделей предметных областей, созданных с использованием диаграмм UML. Исследование было проведено при участии более 100 студентов, от которых требовалось прочитать и понять специфицированные таким образом модели. Было выявлено, что явное руководство по использованию модели, включенное в спецификацию, серьезно улучшает ее читабельность и усваиваемость.

Ряд работ описывает возможности автоматической трансляции UML-диаграмм в более формализованные формы спецификации с помощью их дополнительной детализации. Обеспечение возможности такой трансформации — крайне перспективное направление исследований, так как приведение UML-диаграмм к более формализованному виду позволит использовать методы формального анализа для их проверки.

К примеру, работа [29] представляет систему, которая, пользуясь набором определенных правил, переводит диаграмму классов UML в программный код, специфицирующий поведенческие аспекты системы.

В работе [30] описывается метод генерации вариантов тестов для программной системы из диаграммы UML. Создание тестов обычно происходит на основе уже написанного кода, но их автоматическая генерация на основе спецификаций уменьшит затраты времени в фазе тестирования в разы. Для создания тестов из UML-диаграммы сначала генерируется набор сценариев, из которых далее берется вся необходимая информация для создания тестов.

Работа [31] описывает систему для оценки производительности будущей системы, представленной в виде UML-диаграммы. Для этого к имеющимся

данным добавляются необходимые аннотации с помощью расширения для языка UML, OMG-профиля. Затем диаграмма трансформируется в промежуточную модель, которая затем и оценивается с точки зрения производительности.

Работа [32] также описывает возможность вычисления различных метрик производительности (времени отклика, времени прохождения сигнала, пропускной способности) модели в виде диаграммы UML. При этом диаграммы UML трансформируются в сети Петри — известную математическую абстракцию для описания динамических систем. Сети Петри являются формальным языком, что позволяет проводить их анализ, в том числе анализ производительности, с использованием формальных методов. Ряд работ описывает возможности по дальнейшей работе с сетями Петри [33—35].

В статье [36] предлагается метод перевода спецификаций в виде UML-диаграмм на естественный язык, что позволяет получить спецификацию на естественном языке, свободную от стандартных для естественного языка недостатков: неточности и двусмысленности. Несмотря на то что спецификации на естественном языке являются менее формализованными, а значит, такая трансформация будет "шагом назад" в процессе создания спецификации, итеративность такого процесса и изменчивость требований клиентов требуют вовлеченности будущих пользователей в процесс создания спецификаций на всех его стадиях, что не может быть достигнуто, если спецификации будут описаны на формальном или полужормальном языке. Именно поэтому важна возможность переводить спецификации из формализованных форм на естественный язык и обратно.

В работе [37] описана система, позволяющая создавать спецификации протоколов коммуникационных систем из спецификаций сервисов в виде UML-диаграмм. Полученные в этой работе результаты позволяют автоматически генерировать крайне сложные коммуникационные протоколы, избегая больших временных затрат на последовательность операций по разработке протокола, его анализу, поиску и исправлению ошибок.

Интересна работа [38], в которой представлен основанный на диаграммах UML визуальный язык VLC. Он позволяет визуализировать дополнительные данные, не поддерживаемые в UML, а также управлять применением формальных техник к диаграммам посредством генерации формально описанной модели из диаграммы. Ряд работ [39, 40] описывает возможности применения языка VLC в рамках различных процессов создания спецификаций, а в работе [41] рассматривается возможность генерации Z-нотаций — строго формализованных форм спецификаций, используемых для моделирования программных продуктов, из диаграмм VLC.

Спецификации в форме таблиц

Еще одним удобным полужформализованным средством разработки спецификаций являются таблицы. Они обычно служат для детального описания определенных данных о структуре системы. Табличные данные формализуемы и могут быть подвергнуты анализу и валидации, если заданы структура каждой таблицы и ее связь с другими данными, содержащимися в спецификации.

Работы, касающиеся табличного формата данных, можно условно разделить на две категории: предлагающие возможности валидации табличных данных [42—45] и описывающие механизмы создания таблиц, корректных по построению [46—48]. Стоит отметить, что формальный анализ табличных данных при отсутствии информации о структуре таблицы невозможен в большинстве случаев, поэтому все подходы к валидации табличных данных, не полагающиеся на знания структуры таблиц и предметной области, требуют непосредственного участия пользователя.

В статье [42] рассматривается система полужавтоматической верификации табличных данных, выполненная в виде расширения для Microsoft Excel, самой распространенной системы работы с таблицами в мире. В данной системе обязанность по проверке корректности и указанию ошибок лежит на пользователе, а система подсказывает необходимые исправления. В [43] описывается похожая система полужавтоматической верификации таблиц, но с несколько другим подходом к разделению обязанностей между пользователем и системой. Здесь пользователь задает набор необходимых ограничений и ожидаемых значений для части ячеек таблицы, после чего система предлагает возможные исправления в табличных формулах. В работе [44] предлагается метод тестирования табличных данных, основанный на методологии потока данных. С помощью этого метода особенно удобно проводить тестирование таблиц с числовыми данными, так как формулы и их сложные зависимости являются основным объектом рассмотрения в методологии потока данных.

В статье [45] рассматривается возможность применения мутационного тестирования для валидации таблиц. Метод мутационного тестирования был разработан для оценки систем тестирования программного обеспечения и заключается в последовательном применении мутационных операторов, имитирующих программные ошибки, к разным областям исходного кода и последующем тестировании получившихся программ. Для того чтобы считаться достаточно полным, набор тестов должен выявить всех "мутантов". Ранее мутационное тестирование не получило широкого распространения вследствие своей огромной вычислительной сложности, но с развитием вычислительных технологий этот метод начал находить широкое приме-

нение в процессе создания пакетов тестов для программных систем. Работа [45] показывает, что этот метод можно с успехом применять и для оценки систем валидации табличных данных.

Все вышеописанные подходы к валидации таблиц являются полужавтоматическими, причем в большинстве из них пользователь сам должен указывать системе на ошибки в данных. Полная автоматизация тестирования таблиц может быть создана только при условии соответствия табличного синтаксиса и семантики определенной модели. Для решения этой задачи в последнее время предложено несколько подходов. Все они предлагают создавать таблицы в два этапа: сначала создаются шаблоны таблиц, а затем отдельные экземпляры, соответствующие этим шаблонам. Такие таблицы по построению свободны от многих типовых для табличных данных ошибок, например, некорректных ссылок, типов и пробелов в записях.

Самым известным проектом в этом направлении стала объектно-ориентированная система создания таблиц ClassSheets [49]. В ней созданию таблиц предшествует описание их модели.

Важным аспектом при создании таблиц на основе шаблонов является поддержка изменений этих шаблонов после создания таблиц. В этом случае необходимо применять к уже существующим экземплярам таблиц операторы, дублирующие изменения в их шаблонах. В частности, работа [47] описывает пары трансформаций, одна из которых применяется к модели (шаблону), а другая — к экземпляру таблицы. Статьи [48, 50, 51] описывают расширения для системы ClassSheets, также реализующие такие трансформации.

Заключение

В данной работе были рассмотрены различные формы представления спецификаций, среди которых основной акцент был сделан на диаграммной и табличной формах как на частично формализованных и представляющих наибольший интерес в качестве объекта анализа.

На основе приведенного обзора работ можно сделать вывод, что крайне перспективными направлениями исследований является поиск способов автоматической трансляции спецификаций в формальные модели и поиск возможностей создания корректных спецификаций на частично формализованных языках. Примером первого направления является генерация различных формальных моделей из диаграмм с добавлением некоторой дополнительной информации, а также генерация диаграмм из текстового описания [52]. Примером второго является генерация корректных по построению таблиц с помощью объектно-ориентированного подхода.

Список литературы

1. **Khan M. N. A., Khalid M., ul Haq S.** Review of requirements management issues in software development // *International Journal of Modern Education and Computer Science (IJMECS)*. 2013. N. 5.1. 21 p.
2. **Pandey D., Suman U., Ramani A. K.** An Effective Requirement Engineering Process Model for Software Development and Requirements // *International Conference on Advances in Recent Technologies in Communication and Computing Management*. 2010. P. 287–291.
3. **Neetu K. S., Pillai A. S.** State-Of-The-Practice Survey And Comparison In Requirements Engineering In Healthcare IT Projects // *International Conference on Advances in Computing, Communications and Informatics*. Aug 2012.
4. **Lee Y. K., Kim N. H., Kim D., Lee D. H., In H. P.** Customer Requirements Elicitation based on Social Network Service // *KSTT Transactions On Internet And Information Systems*. Oct 2011. Vol. 5.10.
5. **Keller T.** Contextual Requirements Elicitation // *Seminar in Requirements Engineering*. Spring. Department of Informatics, University of Zurich. 2011.
6. **Dhungana D., Seyff N., Graf F.** Research Preview: Supporting End-user Requirements Elicitation Using Product Line Variability Models // *Requirements Engineering Foundation for Software Quality*. Mar 2011. Vol. 6606. P. 66–71.
7. **Neetu K. S., Pillai A. S.** A survey on global requirements elicitation issues and proposed research framework // *Software Engineering and Service Science (ICSESS)* // 4th IEEE International Conference on. IEEE. 2013. P. 554–557.
8. **Pohl K.** The three dimensions of requirements engineering // *Seminal Contributions to Information Systems Engineering*. Springer Berlin Heidelberg. 2013. P. 63–80.
9. **Pohl K., Ulfat-Bunyadi N.** The Three Dimensions of Requirements Engineering: 20 Years Later // *Seminal Contributions to Information Systems Engineering*. Springer Berlin Heidelberg. 2013. P. 81–87.
10. **Blekhman A., Wachs J. P., Dori D.** Model-Based System Specification With Tesperanto: Readable Text From Formal Graphics // *Systems, Man, and Cybernetics: Systems* // *IEEE Transactions*. 2015. Vol. 45 (11). P. 1448–1458.
11. **Kasser J.** Object-oriented requirements engineering and management // PhD diss., Systems Engineering Society of Australia. 2003.
12. **Liu Z., Jifeng H., Li X., Chen Y.** A relational model for formal object-oriented requirement analysis in UML // *Formal Methods and Software Engineering*. Berlin, Germany: Springer. 2003. P. 641–664.
13. **Bastani B.** Process-oriented abstraction of the complex evolvable systems // *ACM SIGSOFT Softw. Eng. Notes*. 2008. Vol. 33, N. 3. P. 1.
14. **Liu H., Lieberman H.** Metafor: Visualizing stories as code // *Proc. 10th Int. Conf. Intell. User Interf., San Diego, CA, USA*. 2005. P. 305–307.
15. **Harel D.** Can programming be liberated, period? // *Computer*. Jan. 2008. Vol. 41, N. 1. P. 28–37.
16. **Harel D.** From play-in scenarios to code: An achievable dream // *Computer*. 2001. Vol. 34, N. 1. P. 53–60.
17. **Gasparini L., Norman T. J., Kollingbaum M. J., Chen L., Meyer J. J. C.** Verifying Normative System Specification containing Collective Imperatives and Deadlines // *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*. International Foundation for Autonomous Agents and Multiagent Systems. 2015. P. 1821–1822.
18. **Plagge D., Leuschel M.** Seven at one stroke: LTL model checking for high-level specifications in B, Z, CSP, and more // *International journal on software tools for technology transfer*. 2010. Vol. 12.1. P. 9–21.
19. **Legay A., Delahaye B., Bensalem S.** Statistical model checking: An overview // *Runtime Verification*. Springer Berlin Heidelberg. 2010. P. 122–135.
20. **Amálio N.** Generative frameworks for rigorous model-driven development. PhD thesis, Dept. Computer Science, Univ. of York. 2007.
21. **Eynard B., Gallet T., Nowak P., Roucoules L.** UML based specifications of PDM product structure and workflow // *Computers in Industry*. 2004. Vol. 55.3. P. 301–316.
22. **Gavrilescu M., Magureanu G., Pescaru D., Jian I.** Towards UML software models for Cyber Physical System applications // *Telecommunications Forum (TELFOR)*, 2012 20th. IEEE. 2012. P. 1701–1704.
23. **Grobelna I., Grobelny M.** UML Activity Diagrams in Requirements Specification of Logic Controllers // *Conference: Design and Analysis of Control Systems (DACs 2015)*, At Athens, Greece. 2015. Vol. 1702, No. 1. 100005 p.
24. **Grobelna I., Grobelny M., Adamski M.** Model Checking of UML Activity Diagrams in Logic Controllers Design // *Proceedings of the Ninth International Conference on Dependability and Complex Systems DepCoS-RELCOMEX*. June 30–July 4, 2014, Brunywn, Poland. Springer International Publishing. 2014. P. 233–242.
25. **Daw Z., Cleaveland R., Vetter M.** Integrating model checking and UML based model-driven development for embedded systems // *Electronic Communications of the EASST 66*. 2014.
26. **Robles K., Fraga A., Morato J., Llorens J.** Towards an ontology-based retrieval of UML Class Diagrams // *Information and Software Technology*. 2012. Vol. 54.1. P. 72–86.
27. **Salami H. O., Ahmed M. A.** UML artifacts reuse: state of the art // *International Journal of Soft Computing and Software Engineering*. 2014. Vol. 3. P. 115–122.
28. **Reinhartz-Berger I., Sturm A.** Comprehensibility of UML-based software product line specifications // *Empirical Software Engineering*. 2014. Vol. 19.3. P. 678–713.
29. **Albert M., Cabot J., Gómez C., Pelechano V.** Generating operation specifications from UML class diagrams: A model transformation approach // *Data & Knowledge Engineering*. 2011. Vol. 70.4. P. 365–389.
30. **Linzhang W., Jiesong Y., Xiaofeng Y., Jun H., Xuandong L., Guo Z.** Generating test cases from UML activity diagram based on gray-box method // *Software Engineering Conference, 2004. 11th Asia-Pacific*. IEEE. 2004. P. 284–291.
31. **Distefano S., Scarpa M., Puliafito A.** From UML to Petri nets: The PCM-based methodology // *Software Engineering, IEEE Transactions*. 2011. Vol. 37.1. P. 65–79.
32. **Bernardi S., Merseguer J.** Performance evaluation of UML design with Stochastic Well-formed Nets // *Journal of Systems and Software*. 2007. Vol. 80.11. P. 1843–1865.
33. **Grobelna I., Wisniewska M., Wisniewski R., Grobelny M., Mroz P.** Decomposition, validation and documentation of control process specification in form of a Petri net // *Human System Interactions (HSI)*, 2014 7th International Conference on. IEEE. 2014. P. 232–237.
34. **Karatkevich A.** Dynamic analysis of Petri net-based discrete systems // *Lecture Notes in Control and Information Sciences*. Springer Berlin Heidelberg, 2007. Vol. 356.
35. **Wisniewski R., Karatkevich A., Adamski M., Kur D.** Application of comparability graphs in decomposition of Petri nets // *7th International Conference on Human System Interactions (HSI)*, Portugal. 2014. P. 216–220.
36. **Meziane F., Athanasakis N., Ananiadou S.** Generating Natural Language specifications from UML class diagrams // *Requirements Engineering*. 2008. Vol. 13.1. P. 1–18.
37. **Al Dallal J., Saleh K. A.** Synthesizing distributed protocol specifications from a UML state machine modeled service specification // *Journal of Computer Science and Technology*. 2012. Vol. 27.6. P. 1150–1168.
38. **Amálio N., Kelsen P., Ma Q.** The visual contract language: abstract modelling of software systems visually, formally and modularly // *Univ. of Luxembourg, Tech. Rep. TRLASSY-10-03*, 2010.
39. **Amálio N., Kelsen P.** Modular design by contract visually and formally using VCL. // *Visual Languages and Human-Centric Computing (VL/HCC)*, 2010 IEEE Symposium. 2010. P. 227–234.
40. **Amálio N., Kelsen P., Ma Q., Glodt C.** Using VCL as an aspect-oriented approach to requirements modelling. // *Transactions on aspect-oriented software development VII*, Springer Berlin Heidelberg. 2010. P. 151–199.
41. **Amálio N., Glodt C., Kelsen P.** Building VCL models and automatically generating Z specifications from them // *FM 2011: Formal Methods*. Springer Berlin Heidelberg. 2011. P. 149–153.
42. **Abraham R., Erwig M.** Goal-directed debugging of spreadsheets // *Visual Languages and Human-Centric Computing*, 2005 IEEE Symposium on. IEEE. 2005. P. 37–44.
43. **Abraham R., Erwig M.** GoalDebug: A spreadsheet debugger for end users // *Proceedings of the 29th international conference on Software Engineering*. IEEE Computer Society. 2007. P. 251–260.

44. Fisher M., Rothermel G., Creelan T., Burnett M. Scaling a Dataflow Testing Methodology to the Multi-paradigm World of Commercial Spreadsheets // Software Reliability Engineering, 2006. ISSRE'06. 17th International Symposium on. IEEE. 2006. P. 13–22.

45. Abraham R., Erwig M. Mutation operators for spreadsheets // Software Engineering, IEEE Transactions. 2009. Vol. 35.1. P. 94–108.

46. Abraham R., Erwig M., Kollmansberger S., Seifert E. Visual specifications of correct spreadsheets // Visual Languages and Human-Centric Computing, 2005 IEEE Symposium on. IEEE. 2005. P. 189–196.

47. Mendes J. Coupled evolution of model-driven spreadsheets // Software Engineering (ICSE), 2012 34th International Conference on. IEEE. 2012. P. 1616–1618.

48. Cunha J., Fernandes J. P., Mendes J., Saraiva J. MDSheet: A framework for model-driven spreadsheet engineering // Proceedings

of the 34th International Conference on Software Engineering. IEEE Press. 2012. P. 1395–1398.

49. Engels G., Erwig M. ClassSheets: automatic generation of spreadsheet applications from object-oriented specifications // Proceedings of the 20th IEEE/ACM international Conference on Automated software engineering. ACM. 2005. P. 124–133.

50. Cunha J., Fernandes J. P., Mendes J., Pacheco H., Saraiva J. Bidirectional Transformation of Model-Driven Spreadsheets // ICMT. 2012. Vol. 12. P. 105–120.

51. Stevens P. Bidirectional model transformations in QVT: Semantic issues and open questions // Model Driven Engineering Languages and Systems. Springer Berlin Heidelberg. 2007. P. 1–15.

52. Печенко И. С., Венгеп О. В. Способ автоматизированного создания диаграмм последовательности операций для сценариев поведения системы, описанных в виде текста // Информационные технологии. 2015. Том 21, № 1.

I. S. Pechenko, Research Associate, e-mail: ivan.pechenko@intel.com
ZAO "Intel AO"

Specification Representation Forms: Issues and Ways for Automatic Processing

The complexity of computer systems has grown dramatically over past thirty years. The growth of system design complexity leads to increasing importance of full and accurate system specifications creation. The quality of these specifications affect greatly on the product design success.

Requirement engineering process consists of several parts: requirements elicitation; requirements analysis; specification creation; specification validation and verification; requirements management.

Specification can consist of many documents written in different languages, for example: natural languages, structural and flow diagrams, spreadsheets and formal languages. The whole specification creation process can be characterized as a transition from natural language requirements got from stakeholders to more formal specification. Natural language requirements are commonly inaccurate, inconsistent, incomplete and cannot be automatically processed. Formal language specifications are the example of the most accurate and unambiguous system description. They also have another great advantage: they can be analyzed using formal methods. Despite this, usually only a small part of system specification is written in formal languages, because they are hard to learn and to read and have many semantic restrictions. Diagrams and spreadsheets stay somewhere between natural and formal languages: they have formal syntax, but also additional data in natural language.

Diagrams are one of the most common and convenient form of semi-formal specifications. UML is the most prevalent visual language in requirement engineering. There are both structural and behavioral types of UML diagrams. Specifications in UML contain usually high-level information and cannot be directly analyzed using formal methods. Instead of this, usually UML diagrams are modified, supplemented with additional data and then translated into some formal language. Many recent studies use this flow for diagram analysis.

Spreadsheets are another common form of semi-formal specifications. They usually serve as detailed description of system structure. They can be analyzed with formal methods only if the structure and syntax of the spreadsheet are set. There are two common directions in spreadsheets specification studies. One group of works consider spreadsheet validation capabilities. They use semi-automatic validation because, if spreadsheet structure is not set, formal validation methods cannot be applied to the spreadsheet. Another group of studies consider making spreadsheets correct by construction using spreadsheet templates.

Keywords: requirements, specification, validation, natural language, formal language, automatic processing, UML diagram, automatic translation, spreadsheet, spreadsheet template

References

1. Khan M. N. A., Khalid M., ul Haq S. Review of requirements management issues in software development, *International Journal of Modern Education and Computer Science (IJMECS)*. 2013, no. 5.1, 21 p.
2. Pandey D., Suman U., Ramani A. K. An Effective Requirement Engineering Process Model for Software Development and Requirements, *International Conference on Advances in Recent Technologies in Communication and Computing Management*, 2010, pp. 287–291.
3. Neetu K. S., Pillai A. S. State-Of-The-Practice Survey And Comparison In Requirements Engineering In Healthcare IT Projects, *International Conference on Advances in Computing, Communications and Informatics*. Aug 2012.
4. Lee Y. K., Kim N. H., Kim D., Lee D. H., In H. P. Customer Requirements Elicitation based on Social Network Service, *KSTT Transactions On Internet And Information Systems*, Oct 2011, vol. 5.10.
5. Keller T. Contextual Requirements Elicitation, *Seminar in Requirements Engineering*. Spring 2011. Department of Informatics, University of Zurich.
6. Dhungana D., Seyff N., Graf F. Research Preview: Supporting End-user Requirements Elicitation Using Product Line Variability Models, *Requirements Engineering Foundation for Software Quality*, Mar 2011, vol. 6606, pp. 66–71.
7. Neetu K. S., Pillai A. S. A survey on global requirements elicitation issues and proposed research framework, *Software Engineering*

and Service Science (ICSESS). 4th IEEE International Conference on IEEE, 2013, pp. 554–557.

8. **Pohl K.** The three dimensions of requirements engineering, *Seminal Contributions to Information Systems Engineering*. Springer Berlin Heidelberg, 2013. pp. 63–80.

9. **Pohl K., Ulfat-Bunyadi N.** The Three Dimensions of Requirements Engineering: 20 Years Later, *Seminal Contributions to Information Systems Engineering*. Springer Berlin Heidelberg, 2013, pp. 81–87.

10. **Blekhnan A., Wachs J. P., Dori D.** Model-Based System Specification With Tesperanto: Readable Text From Formal Graphics, *Systems, Man, and Cybernetics: Systems*, IEEE Transactions, 2015, vol. 45 (11), pp. 1448–1458.

11. **Kasser J.** *Object-oriented requirements engineering and management*. PhD diss., Systems Engineering Society of Australia. 2003.

12. **Liu Z., Jifeng H., Li X., Chen Y.** A relational model for formal object-oriented requirement analysis in UML, *Formal Methods and Software Engineering*. Berlin, Germany: Springer, 2003, pp. 641–664.

13. **Bastani B.** Process-oriented abstraction of the complex evolvable systems, *ACM SIGSOFT Softw. Eng. Notes*, 2008, vol. 33, no. 3. P. 1.

14. **Liu H., Lieberman H.** Metafor: Visualizing stories as code, *Proc. 10th Int. Conf. Intell. User Interf.*, San Diego, CA, USA, 2005. pp. 305–307.

15. **Harel D.** Can programming be liberated, period? *Computer*, Jan. 2008, vol. 41, no. 1, pp. 28–37.

16. **Harel D.** From play-in scenarios to code: An achievable dream. *Computer*, 2001, vol. 34, no. 1, pp. 53–60.

17. **Gasparini L., Norman T. J., Kollingbaum M. J., Chen L., Meyer J. J. C.** Verifying Normative System Specification containing Collective Imperatives and Deadlines, *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems. International Foundation for Autonomous Agents and Multiagent Systems*, 2015, pp. 1821–1822.

18. **Plagge D., Leuschel M.** Seven at one stroke: LTL model checking for high-level specifications in B, Z, CSP, and more, *International journal on software tools for technology transfer*, 2010, vol. 12.1, pp. 9–21.

19. **Legay A., Delahaye B., Bensalem S.** Statistical model checking: An overview, *Runtime Verification*. Springer Berlin Heidelberg, 2010, pp. 122–135.

20. **Amálio N.** *Generative frameworks for rigorous model-driven development*. PhD Liss, Dept. Computer Science, Univ. of York. 2007.

21. **Eynard B., Gallet T., Nowak P., Roucoules L.** UML based specifications of PDM product structure and workflow, *Computers in Industry*, 2004, vol. 55.3, pp. 301–316.

22. **Gavrilescu M., Magureanu G., Pescaru D., Jian I.** Towards UML software models for Cyber Physical System applications, *Telecommunications Forum (TELFOR), 2012 20th. IEEE*, 2012, pp. 1701–1704.

23. **Grobelna I., Grobelny M.** UML Activity Diagrams in Requirements Specification of Logic Controllers, *Conference: Design and Analysis of Control Systems (DACs 2015), At Athens, Greece*, 2015, vol. 1702, no. 1. P. 100005.

24. **Grobelna I., Grobelny M., Adamski M.** Model Checking of UML Activity Diagrams in Logic Controllers Design, *Proceedings of the Ninth International Conference on Dependability and Complex Systems DepCoS-RELCOMEX*. June 30–July 4, 2014, Brunyw, Poland. Springer International Publishing, 2014, pp. 233–242.

25. **Daw Z., Cleaveland R., Vetter M.** Integrating model checking and UML based model-driven development for embedded systems, *Electronic Communications of the EASST* 66. 2014.

26. **Robles K., Fraga A., Morato J., Llorens J.** Towards an ontology-based retrieval of UML Class Diagrams, *Information and Software Technology*, 2012, vol. 54.1, pp. 72–86.

27. **Salami H. O., Ahmed M. A.** UML artifacts reuse: state of the art, *International Journal of Soft Computing and Software Engineering*, 2014, vol. 3, pp. 115–122.

28. **Reinhartz-Berger I., Sturm A.** Comprehensibility of UML-based software product line specifications, *Empirical Software Engineering*, 2014, vol. 19.3, pp. 678–713.

29. **Albert M., Cabot J., Gómez C., Pelechano V.** Generating operation specifications from UML class diagrams: A model transformation approach, *Data & Knowledge Engineering*, 2011, vol. 70.4, pp. 365–389.

30. **Linzhang W., Jiesong Y., Xiaofeng Y., Jun H., Xuandong L., Guo Z.** Generating test cases from UML activity diagram based on

gray-box method, *Software Engineering Conference, 2004. 11th Asia-Pacific. IEEE*, 2004, pp. 284–291.

31. **Distefano S., Scarpa M., Puliafito A.** From UML to Petri nets: The PCM-based methodology, *Software Engineering, IEEE Transactions*, 2011, vol. 37.1, pp. 65–79.

32. **Bernardi S., Merseguer J.** Performance evaluation of UML design with Stochastic Well-formed Nets, *Journal of Systems and Software*, 2007, vol. 80.11, pp. 1843–1865.

33. **Grobelna I., Wisniewska M., Wisniewski R., Grobelny M., Mroz P.** Decomposition, validation and documentation of control process specification in form of a Petri net, *Human System Interactions (HSI), 2014 7th International Conference on. IEEE*, 2014, pp. 232–237.

34. **Karatkevich A.** Dynamic analysis of Petri net-based discrete systems, *Lecture Notes in Control and Information Sciences*, Springer Berlin Heidelberg, 2007, vol. 356.

35. **Wiśniewski R., Karatkevich A., Adamski M., Kur D.** Application of comparability graphs in decomposition of Petri nets, *7th International Conference on Human System Interactions (HSI), Portugal*, 2014, pp. 216–220.

36. **Meziane F., Athanasakis N., Ananiadou S.** Generating Natural Language specifications from UML class diagrams, *Requirements Engineering*, 2008, vol. 13.1, pp. 1–18.

37. **Al Dallal J., Saleh K. A.** Synthesizing distributed protocol specifications from a UML state machine modeled service specification, *Journal of Computer Science and Technology*, 2012, vol. 27.6, pp. 1150–1168.

38. **Amálio N., Kelsen P., Ma Q.** The visual contract language: abstract modelling of software systems visually, formally and modularly, *Univ. of Luxembourg, Tech. Rep. TRLASSY-10-03*. 2010.

39. **Amálio N., Kelsen P.** Modular design by contract visually and formally using VCL, *Visual Languages and Human-Centric Computing (VL/HCC), 2010 IEEE Symposium*, 2010, pp. 227–234.

40. **Amálio N., Kelsen P., Ma Q., Glodt C.** Using VCL as an aspect-oriented approach to requirements modelling, *Transactions on aspect-oriented software development VII*, Springer Berlin Heidelberg, 2010, pp. 151–199.

41. **Amálio N., Glodt C., Kelsen P.** Building VCL models and automatically generating Z specifications from them, *FM 2011: Formal Methods*. Springer Berlin Heidelberg, 2011, pp. 149–153.

42. **Abraham R., Erwig M.** Goal-directed debugging of spreadsheets, *Visual Languages and Human-Centric Computing, 2005 IEEE Symposium on. IEEE*, 2005, pp. 37–44.

43. **Abraham R., Erwig M.** GoalDebug: A spreadsheet debugger for end users, *Proceedings of the 29th international conference on Software Engineering. IEEE Computer Society*, 2007, pp. 251–260.

44. **Fisher M., Rothermel G., Creelan T., Burnett M.** Scaling a Dataflow Testing Methodology to the Multi-paradigm World of Commercial Spreadsheets, *Software Reliability Engineering, 2006. ISSRE'06. 17th International Symposium on. IEEE*, 2006, pp. 13–22.

45. **Abraham R., Erwig M.** Mutation operators for spreadsheets, *Software Engineering, IEEE Transactions*, 2009, vol. 35.1, pp. 94–108.

46. **Abraham R., Erwig M., Kollmansberger S., Seifert E.** Visual specifications of correct spreadsheets, *Visual Languages and Human-Centric Computing, 2005 IEEE Symposium on. IEEE*, 2005, pp. 189–196.

47. **Mendes J.** Coupled evolution of model-driven spreadsheets, *Software Engineering (ICSE), 2012 34th International Conference on. IEEE*, 2012, pp. 1616–1618.

48. **Cunha J., Fernandes J. P., Mendes J., Saraiva J.** MDSheet: A framework for model-driven spreadsheet engineering, *Proceedings of the 34th International Conference on Software Engineering. IEEE Press*, 2012, pp. 1395–1398.

49. **Engels G., Erwig M.** ClassSheets: automatic generation of spreadsheet applications from object-oriented specifications, *Proceedings of the 20th IEEE/ACM international Conference on Automated software engineering. ACM*, 2005, pp. 124–133.

50. **Cunha J., Fernandes J. P., Mendes J., Pacheco H., Saraiva J.** Bidirectional Transformation of Model-Driven Spreadsheets, *ICMT*, 2012, vol. 12, pp. 105–120.

51. **Stevens P.** Bidirectional model transformations in QVT: Semantic issues and open questions, *Model Driven Engineering Languages and Systems*. Springer Berlin Heidelberg, 2007, pp. 1–15.

52. **Pechenko I., Venger O.** Sposob avtomatizirovannogo sozdaniya diagram posledovatel'nosti operaciy dlya scenariy povedeniya systemy, opisannih v vide texta, *Informacionnye tehnologii*, 2015, vol. 21, no. 1.

УДК 51-7

А. В. Вишнеков, д-р техн. наук, проф., avishnekov@hse.ru,

Е. М. Иванова, канд. техн. наук, доц., emivanova@hse.ru

Национальный исследовательский университет "Высшая школа экономики", Москва

Автоматизация выбора траектории обучения в среде Smart-образования

В работе рассматриваются вопросы выбора траектории обучения в среде Smart-образования. Предлагается методика выбора траектории на основе применения методов поддержки принятия решений, позволяющая использовать как опыт и знания экспертов в различных областях организации и обеспечения образовательного процесса, так и пожелания самого обучаемого. Предложенная методика позволяет автоматизировать процесс выбора наиболее рациональной траектории обучения и сократить время на принятие решения.

Ключевые слова: Smart-образование, траектория обучения, методы поддержки принятия решения, метод минимального расстояния

Введение

Интенсивность развития современных технологий диктуют необходимость пересмотра подхода к образовательным стандартам, средствам, технологиям, методикам, вопросам организации учебного процесса [1]. Новый подход должен обеспечить более глубокое усвоение материала, сокращение сроков и затрат на образовательный процесс без снижения его качества. Одним из способов решения этой задачи может стать применение среды Smart-образования.

Smart-образование предполагает использование в едином образовательном процессе множества образовательных элементов: технологий, ресурсов и средств. Исследователи в области Smart-образования выделяют следующие его характеристики/элементы [2]:

- использование большого количества информационных источников;
- непрерывное обновление учебных материалов;
- максимальное разнообразие мультимедиа (аудио, видео, графика);
- способность образовательного учреждения быстро и просто подстраиваться под уровень и потребности слушателя;
- высокий профессионализм преподавателей;
- сочетание образовательной и научной деятельности;
- персонализация курсов и дисциплин;
- творческий подход к процессу обучения;
- использование Smart-учебников;

- электронная библиотека;
- автоматическая фильтрация по уровню освоения материала (рейтинг знаний);
- групповая работа соавторов и читателей в Интернет-пространстве;
- виртуальный кампус: электронная библиотека, межвузовский обмен материалами, электронное обучение, электронные курсы;
- междисциплинарные задания;
- увеличение доли практических занятий;
- участие в актуальных инновационных проектах: региональных, отраслевых, общегосударственных...

В качестве одного из необходимых элементов такого Smart-образования называется персонафикация подхода к образовательному процессу, в частности, в работе [2] — персонафикация набора курсов и дисциплин, в работе [3] — персонафикация содержания учебных материалов.

При использовании принципа персонафикации авторами предлагается для повышения эффективности процесса обучения разработать и применить процедуры формирования и выбора индивидуальных образовательных траекторий, максимально соответствующих потребностям обучающихся. Под образовательной траекторией будем понимать набор и последовательность применяемых элементов, технологий и средств обучения. На основе этих данных может быть сформирована наиболее рациональная траектория обучения не только для конкретного обучаемого, но и для учебной группы.

Рассмотрим более детально элементы Smart-образования, которые могут быть использованы при

формировании индивидуальных образовательных траекторий.

Элемент 1. Обязательной составляющей любого образовательного процесса является самостоятельное изучение учащимися материалов курса. Это можно сделать с использованием большого количества различных информационных источников [4]. Тогда в качестве первой группы элементов траектории Smart-образования можно назвать:

- стандартные привычные всем учебники и учебные пособия, справочная литература;
- научные периодические издания и материалы конференций;
- Интернет-ресурсы, перечень которых формирует как преподаватель, так и сами учащиеся;
- подготовленные преподавателями и учебными ассистентами для лучшего усвоения материала многочисленные электронные материалы и документы:
 - текстовые файлы,
 - презентации,
 - статистические таблицы,
 - мультимедиа-материалы (аудио, видео, графика).

Элемент 2. Следующей важной составляющей любого образования, в том числе и Smart-образования, является способ организации учебного процесса (виды учебных занятий). Тогда в качестве второй группы элементов траектории Smart-образования можно назвать:

- очное прослушивание лекций;
- очное участие в семинарах;
- организация очных лабораторных или практических занятий;
- очное участие в мастер-классе с приглашенными специалистами;
- участие в очных студенческих научно-практических конференциях;
- заочное обучение путем просмотра кейса учебных материалов;
- заочное выполнение домашних заданий;
- групповая работа преподавателей и учащихся в Интернет-пространстве;
- самостоятельные исследования по предложенной преподавателем тематике;
- участие в online-мероприятиях (доклады, вебинары, конференции...);
- участие в деловых играх;
- учебные, научно-исследовательские и производственные практики;
- очные консультации;
- заочные консультации
 - на форумах,
 - через электронную почту,
 - посредством систем мгновенного обмена сообщениями,

— через информационную образовательную среду учебного заведения,

— через системы видеоконференций.

Элемент 3. К третьей группе элементов траектории Smart-образования можно отнести численность учебной группы:

- возможность обучения в малой группе;
- возможность обучения в большой группе (с потоком студентов);
- возможность индивидуального обучения;
- чередование этих видов обучения.

Элемент 4. К четвертой группе элементов траектории Smart-образования можно отнести технологии преподнесения материала. Каждый человек обладает различной способностью запоминания и предпочтениями к способу усвоения новых знаний. Правильно подобранная индивидуальная технология преподнесения знаний дает наибольший эффект для конкретного человека. Здесь можно назвать следующие технологии преподнесения знаний:

- преобладание визуальной информации (текстовой, числовой, графической);
- преобладание аудиоинформации;
- мультимедийные источники информации.

Элемент 5. Также в процессе обучения следует организовать текущий и итоговый контроль знаний. В качестве пятой группы элементов можно использовать различные способы проверки знаний обучаемого:

- очное выполнение контрольных работ;
- очный экзамен;
- очное/заочное тестирование;
- участие в деловых играх;
- выполнение проектов;
- выполнение домашних заданий;
- защита курсовых работ;
- защита междисциплинарных курсовых работ;
- оценка отчетов по различным видам практик:
 - учебной;
 - производственной;
 - научно-исследовательской;
- итоговый междисциплинарный экзамен;
- защита выпускных квалификационных работ.

Элемент 6. Следует также подобрать темп подачи материала для конкретного обучаемого. Шестой группой элементов может быть следующий вариант:

- медленный с многократным повторением пройденного материала;
- стандартный с фиксацией основных положений;
- ускоренный с упором на самоподготовку;
- регулируемый студентом (по готовности перейти к следующему разделу).

Элемент 7. Один из важнейших элементов — организация учебного процесса с точки зрения очередности и числа одновременно изучаемых предметов. К седьмой группе элементов Smart-обучения относятся следующие:

- семестровая система (множество предметов, параллельно изучаемых в течение полугода);
- модульное обучение (небольшое число дисциплин, изучаемых за относительно короткий промежуток времени — 2—3 месяца);
- последовательная траектория изучения дисциплин друг за другом (одна дисциплина — несколько недель и переход к следующей только после успешной сдачи экзамена по текущей, иначе повторное изучение);
- индивидуальный план обучения.

Следует отметить, что набор элементов и их содержание могут варьироваться в зависимости от учебного заведения, вида образовательной программы и базового образовательного стандарта.

Целью Smart-образования является приведение в соответствие качества предоставляемого и качества получаемого образования [5]. Разрабатываемая методика автоматизации выбора траектории обучения позволяет определить наиболее рациональную для обучаемого и преподавателя индивидуальную траекторию Smart-образования, состоящую из определенного набора элементов. На основе выбранной траектории могут быть созданы сценарии учебных мероприятий всех курсов, которые будут увлекать студента, побуждать его к учебной и научной деятельности [3].

В качестве эксперимента Smart-университет может организовать тестовый период обучения. Для студентов на начальном периоде обучения Smart-университет выявляет эффективность применения различных элементов Smart-образования на практике в целях выбора в дальнейшем наиболее рациональной траектории обучения. Например, в течение нескольких занятий проводятся последовательно очное обучение, затем дистанционное обучение, предлагается участие в мини-проектах, деловых играх, изменяется темп подачи и технология преподавания материала и т. п. После чего проводится изучение отклика участников образовательного процесса.

Постановка задачи

Идея предложенной методики заключается в том, чтобы помочь студенту выстроить индивидуальную траекторию обучения и сформировать индивидуальный кейс учебных материалов, средств и технологий обучения. Такие траектории можно выстраивать как для одного курса, так и для всей образовательной программы. Сравнение траекторий может проводить либо сам учащийся, либо группа экспертов с применением методов поддержки принятия решений.

Задачей университета является внедрение Smart-технологий в учебный процесс по образовательным программам основного и дополнительного образования. Следует сформировать группу экспертов,

включающую методистов, психологов, преподавателей, кураторов, экономистов, административных работников или других специалистов по решению руководителя образовательной программы. Группа экспертов для каждого обучаемого должна сформировать набор элементов и траекторию Smart-обучения, на основании оценки возможностей университета и результатов некоторого входного тестирования/опроса обучаемого и ознакомления с его портфолио.

Поиск решения поставленной задачи является трудоемким. Данная задача может быть отнесена к многокритериальной задаче принятия решения в условиях определенности исходной информации, поэтому имеет смысл автоматизировать выбор траектории на основе применения методов поддержки принятия решений. Требуется сформировать множество альтернативных решений (образовательных траекторий) и критериев выбора наиболее рациональной траектории в конкретных условиях.

Альтернативными решениями будут являться траектории обучения, сформированные экспертами. Например, одна из траекторий может выглядеть так:

Траектория 1

Элемент 1. Виды информационных источников:

- бумажные учебники и учебные пособия;
- научные периодические издания;
- авторские учебные пособия, предоставляемые преподавателями;
- Интернет-ресурсы.

Элемент 2. Способ организации учебного процесса: очные занятия (лекции, семинары, лабораторные...).

Элемент 3. Численность учебной группы: обучение в большой группе (с потоком студентов).

Элемент 4. Технологии преподавания учебного материала: мультимедийные источники информации.

Элемент 5. Способы проверки знаний обучаемого:

- очное выполнение контрольных работ;
- очный экзамен;
- выполнение проектов;
- итоговый междисциплинарный экзамен;
- защита выпускных квалификационных работ.

Элемент 6. Темп подачи материала: стандартный с фиксацией основных положений.

Элемент 7. Очередность изучения предметов: семестровая система.

Критерии отбора формируют эксперты на основе нормативных документов, профессионального опыта, сведений о возможностях и предпочтениях студентов, опыта предыдущих лет обучения. В качестве таких критериев можно назвать, например, следующие [2, 3]:

К1. Качество образования (*комплексный критерий, состоящий из нескольких вложенных подкритериев*):

К1.1. Выполнение требований образовательных стандартов.

К1.2. Степень освоения профессиональных компетенций.

К1.3. Степень освоения универсальных компетенций.

К1.4. Степень достижения заявленных целей и результатов обучения.

К2. Доступность элементов образовательной траектории.

К3. Стоимость обучения.

К4. Время, затраченное на освоение образовательной программы (дисциплины).

К5. Степень удовлетворения ожиданий участников образовательного процесса от предоставляемых образовательных услуг.

К5.1. Достоверность получаемой информации.

К5.2. Полнота получаемой информации.

К5.3. Целостность получаемой информации.

К5.4. Ясность изложения материала.

К5.5. Актуальность учебного материала.

При выборе метода поддержки принятия решений необходимо учитывать, что критерии оценки образовательных траекторий носят как численный, так и лингвистический характер, и число альтернативных образовательных траекторий, как правило, невелико (не превышает 10), в соответствии с числом экспертов. Так как решение по выбору образовательной траектории может быть принято при согласованном мнении всех экспертов, то целесообразно применение групповых методов поддержки принятия решений [6].

Рассмотрим некоторые методы, применяемые для согласования групповых (экспертных) решений.

Метод предпочтений основан на ранжировании всех альтернативных траекторий, выполняемом группой экспертов [7]. Каждый эксперт независимо от других выполняет ранжирование траекторий. Если мнения экспертов сильно различаются, то необходимо провести уточнение экспертных оценок до получения допустимого значения коэффициента согласованности оценок. Основная сложность этого метода заключается в достижении заданной согласованности экспертных суждений.

Аналогичная сложность возникает при использовании метода ранга (каждый эксперт выстраивает сам порядок альтернатив), который отличается от метода предпочтений лишь способом задания экспертных ранжировок [8].

Метод кластеризации экспертных оценок позволяет получить согласованные экспертные оценки путем достаточно сложного итерационного процесса коррекции и согласования позиций экспертов. Этот метод требует задания степени предпочтительности альтернативы в абсолютных единицах

на заданных шкалах критериев без процедуры попарного сравнения альтернатив, что может привести к ошибкам при оценке альтернатив, так как данная процедура является сложной для ЛПР (лица, принимающего решение) по результатам психологических исследований [9].

Метод минимального расстояния позволяет найти компромиссное решение с точки зрения удовлетворения мнений всех экспертов в равной степени путем использования процедуры попарного сравнения альтернатив каждым экспертом и использования этих сравнительных оценок для получения итоговой ранжировки альтернатив [10]. Метод минимального расстояния достаточно легко поддается алгоритмизации, хотя число вычислений может быть велико и зависит от числа альтернатив. Процедуры, используемые экспертами в методе минимального расстояния, являются допустимыми по результатам психологических исследований.

В любом случае процедура принятия решения должна быть автоматизирована, так как требует многократного применения экспертами, не являющимися специалистами в области принятия решений.

Пример решения задачи выбора траектории Smart-обучения

Допустим в оценке участвуют пять экспертов. Экспертами был сформирован набор возможных траекторий (Т1, Т2, Т3, Т4, Т5) с конкретизацией по образовательным элементам. Для определенности назовем этот набор альтернативных траекторий набором 1 альтернатив. Допустим ими были сформированы критерии отбора: К1, К2, К3, К4.

Для автоматизации процедуры выбора/ранжирования альтернатив Т1—Т5 используем метод минимального расстояния на основе расчета медианы Кемени—Снелла, представляющей собой ранжировку альтернатив с минимальным суммарным расстоянием до всех экспертных ранжировок [10]. Искомая матрица и ранжировка покажет предпочтительность альтернативных траекторий для конкретного обучаемого, расположив альтернативные решения от лучшего к худшему.

На первом шаге применения метода каждый эксперт строит матрицы Y попарного сравнения альтернатив из набора 1 по совокупности критериев.

Если эксперт затрудняется в построении подобной матрицы, то можно предложить ему взглянуть на проблему более детально и вначале оценить важность каждой из альтернатив по каждому критерию. Для этого понадобится определить важность каждого критерия в общей оценке альтернативы. Взаимную важность критериев следует предварительно оценить с участием всех экспертов, чтобы получить объективное представление проблемы с разных точек зрения.

Для автоматизации расчетов весов критериев можно применить методы поддержки групповых решений: метод предпочтений, метод ранга.

Рассмотрим процедуру расчета весов критериев на основе метода ранга. Допустим, что ЛПР, например руководитель образовательной программы, назначил группу из пяти экспертов ($kЭ = 5$). Далее ЛПР может определить важность голоса каждого эксперта. Это может быть как равная важность, так и различная. Во втором случае дополнительно потребуется выполнить расчет коэффициентов компетентности экспертов с использованием, например, метода ранга. Согласно данному методу ЛПР задает числовые оценки компетентности экспертов ($C_{Эi}$) по десятибалльной шкале, на основании которых можно вычислить их коэффициенты компетентности $W_{Эi}$ по формуле $W_{Эi} = C_{Эi}/\Sigma$ [9], где Σ — сумма всех числовых оценок всех экспертов. Например, ЛПР определил следующие числовые оценки компетентности голосов экспертов (табл. 1, первая строка), сумма которых составит 43 балла (см. табл. 1 — последний столбец), а коэффициенты компетентности $W_{Эi}$ рассчитаны в строке 3 табл. 1.

Далее пять экспертов должны сформировать веса критериев. Процедура определения весов была рассмотрена авторами в работе [10]. Пусть пять экспертов ($i = 1, \dots, 5$) выбрали для рассмотрения четыре критерия ($j = 1, \dots, 4$) и сформировали следующие десятибалльные оценки (R_{ij}) критериев (табл. 2, целые значения).

Затем рассчитаем взвешенную оценку C_{Kj} критерия K_j с учетом коэффициента $W_{Эi}$ компетентности каждого эксперта по формуле $C_{Kj} = W_{Эi} \cdot R_{ij}$ (см. табл. 2, дробные значения в скобках). Далее рассчитывается сумма взвешенных оценок критериев по формуле $SK = \Sigma C_{Kj} = 29,47$ и веса критериев по формуле $W_{Ki} = C_{Kj}/SK$ (см. табл. 2, последняя строка). Согласно полученным весам можно выстроить критерии по важности (от лучшего к худшему): К3, К1, К2, К4. Пользуясь этими оценками, эксперты могут более детально изучить вопрос взаимной важности альтернативных траекторий обучения.

В конечном счете эксперты должны сформировать матрицы $Y^1 \div Y^5$ попарного сравнения альтернатив из набора 1.

В матрице каждый элемент y_{ij} показывает взаимную важность траектории T_i по отношению к траектории T_j по следующему правилу: лучше (1), хуже (-1), равнозначна (0). Пусть, например, были получены следующие оценочные матрицы:

В матрице каждый элемент y_{ij} показывает взаимную важность траектории T_i по отношению к траектории T_j по следующему правилу: лучше (1), хуже (-1), равнозначна (0). Пусть, например, были получены следующие оценочные матрицы:

$$Y^1 = \begin{pmatrix} 0 & 1 & 1 & 1 & 1 \\ -1 & 0 & -1 & 1 & 1 \\ -1 & 1 & 0 & 1 & 1 \\ -1 & -1 & -1 & 0 & -1 \\ -1 & -1 & -1 & 1 & 0 \end{pmatrix}; Y^2 = \begin{pmatrix} 0 & -1 & 1 & -1 & 0 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & -1 & 0 & -1 & 1 \\ 1 & -1 & 1 & 0 & 1 \\ 0 & -1 & -1 & -1 & 0 \end{pmatrix};$$

$$Y^3 = \begin{pmatrix} 0 & 1 & 0 & 1 & 1 \\ -1 & 0 & -1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ -1 & -1 & -1 & 0 & -1 \\ -1 & -1 & -1 & 1 & 0 \end{pmatrix}; Y^4 = \begin{pmatrix} 0 & -1 & 1 & 1 & -1 \\ 1 & 0 & 1 & 1 & -1 \\ -1 & -1 & 0 & -1 & -1 \\ -1 & -1 & 1 & 0 & -1 \\ 1 & 1 & 1 & 1 & 0 \end{pmatrix};$$

$$Y^5 = \begin{pmatrix} 0 & 1 & -1 & 1 & 0 \\ -1 & 0 & -1 & 0 & -1 \\ 1 & 1 & 0 & 1 & 1 \\ -1 & 0 & -1 & 0 & -1 \\ 0 & 1 & -1 & 1 & 0 \end{pmatrix}. \quad (1)$$

Для каждой матрицы $Y^1 \div Y^5$ рассчитывается ранжировка альтернатив по правилу: все элементы по строкам матриц складываются, и в ранжировке наилучшей считается альтернатива с максимальной суммой, а наихудшей — с минимальной, равнозначными — с одинаковыми суммами. На основании этих матриц можно сделать вывод о том, как каждый из экспертов проранжировал по важности имеющиеся траектории:

- ранжировка эксперта 1 (по матрице Y^1) = T1, T3, T2, T5, T4;
- ранжировка эксперта 2 (по матрице Y^2) = T2, T4, T3, [T1, T5] (альтернативы T1 и T5 равнозначны);
- ранжировка эксперта 3 (по матрице Y^3) = [T1, T3], T2, T5, T4 (альтернативы T1 и T3 равнозначны);
- ранжировка эксперта 4 (по матрице Y^4) = T5, T2, T1, T4, T3;
- ранжировка эксперта 5 (по матрице Y^5) = T3, [T1, T5], [T2, T4] (альтернативы T1 и T5 равнозначны и альтернативы T2 и T4 равнозначны).

Таблица 1

Расчет важности голосов экспертов

i	1	2	3	4	5	Σ
$C_{Эi}$	10	8	9	9	7	43
$W_{Эi} = C_{Эi}/\Sigma$	0,24	0,19	0,21	0,21	0,16	1,00

Таблица 2

Экспертные оценки R_{ij} важности критериев

i	$W_{Эi}$	$R_{ij} (C_{Kj} = W_{Эi} \cdot R_{ij})$			
		$j = 1$	$j = 2$	$j = 3$	$j = 4$
1	0,24	8 (1,92)	9 (2,16)	6 (1,44)	5 (1,25)
2	0,19	5 (0,95)	7 (1,33)	8 (1,52)	9 (1,71)
3	0,21	8 (1,68)	6 (1,26)	8 (1,68)	9 (1,89)
4	0,21	8 (1,68)	7 (1,47)	8 (1,68)	5 (1,05)
5	0,16	8 (1,28)	7 (1,12)	9 (1,44)	6 (0,96)
Взвешенная оценка C_{Kj} критерия K_j		7,51	7,34	7,76	6,86
Веса критериев $W_{Ki} = C_{Kj}/SK_i$		0,25	0,24	0,26	0,23

Далее требуется, учитывая все мнения экспертов, вычислить новую ранжировку альтернатив, наиболее близкую ко всем указанным ранжировкам. Для этого вычислим итоговую матрицу оценок Y , равноудаленную от всех пяти матриц $Y^1 \div Y^5$. Для новой матрицы Y будет справедливым утверждение, что сумма модулей разностей ее элементов и элементов любой из матриц $Y^1 \div Y^5$ должна быть минимальной.

Вычисление такой новой матрицы происходит методом перебора всех существующих матриц с нулевой главной диагональю и кососимметричных относительно нее же (значение "1", которое выше диагонали, симметрично значению "-1", которое ниже диагонали, и наоборот). Такой перебор удобнее всего выполнить программно. По новой рассчитанной матрице Y строится новое ранжирование для приведенных альтернатив по степени предпочтительности.

Однако таких равноудаленных матриц может быть несколько, как и несколько ранжировок альтернатив по предпочтительности.

Согласно исходным данным программные расчеты дают следующий результат. Список наиболее предпочтительных альтернативных траекторий при уменьшении предпочтения будет следующим:

- T1, T3, T2, T5, T4;
- T3, T1, T2, T5, T4.

Значит наиболее предпочтительными (и равнозначными) являются альтернативы T1, T3, т. е. необходимо из нескольких равнозначных альтернатив выбрать единственную — наиболее подходящую для данного обучаемого. Для определенности назовем этот новый список альтернативных, но равных по предпочтительности траекторий набором 2 альтернатив.

Поскольку эксперты в выборе уже принимали участие, то остается предоставить право решать самому обучаемому. Если он затрудняется сделать выбор, то помочь ему может один из методов поддержки принятия индивидуальных решений. Наиболее приемлемым будет метод аналитических иерархий [6], так как он применим в случае малого числа альтернатив (в худшем случае это число будет равно числу экспертов) и в случае небольшого числа критериев отбора (формулирует сам обучаемый, и скорее всего их число тоже не будет слишком велико). Также достоинством этого метода является его наглядность и простота использования не специалистами в области принятия решений.

Рассмотрим пример применения этого метода.

Если запустить программу при следующих исходных данных — ранжировок пяти экспертов:

- ранжировка эксперта 1 = T5, T2, [T1, T3], T4;
- ранжировка эксперта 2 = T2, T4, T3, [T1, T5];
- ранжировка эксперта 3 = [T1, T3], T2, T5, T4;
- ранжировка эксперта 4 = T5, T2, T1, T4, T3;
- ранжировка эксперта 5 = T3, [T1, T5], [T2, T4],

то получим такой результат по искомым наиболее предпочтительным ранжировкам:

- T5, T2, T3, T1, T4;
- T2, T3, T5, T1, T4;
- T3, T5, T2, T1, T4.

Так как приведенные выше ранжировки в качестве наиболее предпочтительной траектории указывают разные альтернативы (T2, T3, T5), то требуются дополнительные процедуры выбора одной из них.

Допустим обучаемый для решения по выбору одной из трех (или более) траекторий составил список наиболее важных для себя критериев отбора:

K1 — минимальная стоимость обучения;

K2 — минимальный срок обучения;

K3 — интенсивность обучения (предпочтительно очные занятия);

K4 — виды информационных источников (предпочтительно электронные учебные пособия и Интернет-ресурсы);

K5 — минимальная численность учебной группы.

Эти критерии могут повторять полностью или частично выше названные экспертами критерии оценки траекторий из набора 1 или могут совершенно отличаться от них. В случае, когда критерии частично или полностью повторяются, возможна ситуация, при которой все отобранные как наилучшие альтернативные траектории из набора 2 удовлетворяют повторяющимся критериям, поскольку эксперты проводили отбор с учетом мнения обучаемого (по результатам входного тестирования). Тогда, проанализировав состав и структуру элементов траекторий из набора 2, можно удалить из списка означенные критерии, тем самым сократив число критериев и упростив задачу выбора.

Допустим для определенности, что все альтернативные траектории набора 2 (T2, T3, T5) удовлетворяют критериям K3 и K4. Тогда вычеркнув указанные критерии из рассмотрения, сформируем окончательный список критериев для выбора наиболее предпочтительной альтернативы из набора 2:

K₁ — минимальная стоимость обучения;

K₂ — минимальный срок обучения;

K₃ — наименьшая численность учебной группы.

Для простоты расчетов переобозначим имеющиеся альтернативы:

A₁ — траектория T2;

A₂ — траектория T3;

A₃ — траектория T5.

Для оставшихся критериев в методе аналитических иерархий [8], как и в большинстве методов поддержки, обучаемому предлагается оценить взаимную важность с его точки зрения каждого из критериев, а по результатам оценки заполнить матрицу X взаимного предпочтения критериев (2). В ней на месте элемента x_{ij} указывается степень превосходства критерия K_i над критерием K_j . Будем использовать шкалу относительной важности кри-

териев K_1, K_2, K_3 : равная важность — 1, умеренное превосходство — 3, существенное и сильное превосходство — 5, значительное превосходство — 7, очень большое превосходство — 9, например, если $x_{ij} = 5$, то $x_{ji} = 1/5$ и всегда $x_{ii} = 1$:

$$X = \begin{pmatrix} 1 & 9 & 5 \\ 1/9 & 1 & 1/7 \\ 1/5 & 7 & 1 \end{pmatrix}. \quad (2)$$

Согласно матрице X (2) найдем цену каждого критерия по формуле $C_{Ki} = \sqrt[3]{\prod_j x_{ij}}$ ($i = 1, \dots, 3$; $j = 1, \dots, 3$). Получим следующие значения: $C_{K1} = 3,56$; $C_{K2} = 0,25$; $C_{K3} = 1,12$. Сумма цен критериев равна $CK = 4,93$. Далее рассчитаем веса каждого критерия по формуле $W_{Ki} = \frac{C_{Ki}}{CK}$. Получим $W_{K1} = 0,72$; $W_{K2} = 0,05$; $W_{K3} = 0,23$.

Теперь проведем сравнение трех альтернатив из набора 2 методом аналитических иерархий, для чего построим еще три матрицы попарного сравнения (Z^1, Z^2, Z^3) взаимного предпочтения альтернатив по каждому из трех критериев:

$$Z^1 = \begin{pmatrix} 1 & 3 & 5 \\ 1/3 & 1 & 1/7 \\ 1/5 & 7 & 1 \end{pmatrix}; Z^2 = \begin{pmatrix} 1 & 1 & 7 \\ 1 & 1 & 1/5 \\ 1/7 & 5 & 1 \end{pmatrix};$$

$$Z^3 = \begin{pmatrix} 1 & 1/3 & 1/5 \\ 3 & 1 & 1/7 \\ 5 & 7 & 1 \end{pmatrix}. \quad (3)$$

В каждой матрице Z^k на месте элемента z_{ij}^k указывается степень превосходства альтернативы A_i над альтернативой A_j по критерию K_k . Будем использовать ту же шкалу относительной важности, что и для сравнения критериев. Затем по аналогии рассчитаем вес (W_i^k) каждой i -й альтернативы по каждому k -му критерию:

$$W_1^1 = 0,63; W_2^1 = 0,09; W_3^1 = 0,28;$$

$$W_1^2 = 0,57; W_2^2 = 0,17; W_3^2 = 0,26;$$

$$W_1^3 = 0,09; W_2^3 = 0,17; W_3^3 = 0,74.$$

Далее следует рассчитать ценность каждой альтернативы с учетом весов критериев по формуле

$$U_{Ai} = \sum_{j=1}^3 (W_{Kj} \cdot W_i^j);$$

$$U_{A1} = 0,50; U_{A2} = 0,11; U_{A3} = 0,39.$$

Сравнив получившиеся взвешенные оценки альтернатив из набора 2, легко можно выбрать наиболее предпочтительную из них по максимальной взвешенной оценке. В данном примере таковой оказалась альтернатива A_1 — траектория Т2 (ценность = 0,50).

Заключение

Использование предлагаемой методики позволяет получить целостную характеристику траектории обучения, представляющую наиболее рациональную комбинацию образовательных элементов, обеспечивающих достижение заданных критериев оценки результатов обучения. Полученные рекомендации позволят педагогам выстроить наиболее рациональным образом процесс обучения как индивидуально с каждым обучающимся, так и с учебной группой.

Важным достоинством предлагаемой методики является возможность автоматизации процесса выбора образовательной траектории с учетом особенностей конкретного обучаемого, а также возможностей и требований университета.

Список литературы

1. Усков В. Л., Иванников А. Д., Усков А. В. Перспективные технологии для электронного образования // Информационные технологии. 2007. № 2. С. 32—38.
2. Тихомирова Н. В. Глобальная стратегия развития Smart-общества. МЭСИ на пути к Smart-университету. URL: <http://smartmesiblogspot.ru/2012/03/smart-smart.html> (дата обращения: 21.01.2016).
3. Шакирова З. Х., Ахметшин Д. А. Smart системы в сфере высшего образования // Современные научные исследования и инновации. 2015. № 7. URL: <http://web.snauka.ru/issues/2015/07/56409> (дата обращения: 20.01.2016).
4. Булгаков М. В., Гридина Е. Г., Иванников А. Д., Старых В. А. Федеральная система информационно-образовательных ресурсов // Информационные ресурсы России. 2009. № 2. С. 25—27.
5. Ефремова Н. Тестовый контроль в образовании. URL: http://www.xliby.ru/nauchnaja_literatura_prochee/testovyi_kontrol_v_obrazovanii/index.php (дата обращения: 10.02.2016).
6. Ларичев О. И. Теория и методы принятия решений, а также Хроника событий в Волшебных странах: учебник. Изд. 2-е, перераб. и доп. М.: Логос, 2002. 392 с.
7. Лобанов А. А. Метод предпочтений как инструмент поддержки принятия решений // Перспективы науки и образования. 2015. № 2 (14). С. 36—43.
8. Иванова Е. М., Вишнеков А. В., Ерохин В. В. Автоматизация процедуры выбора микроконтроллера // Нано- и микро-системная техника. 2014. № 7. С. 14—21.
9. Иванова Е. М., Вишнеков А. В., Ерохин В. В. Верификация СНК: выбор стратегии // Нано- и микро-системная техника. 2014. № 12. С. 30—36.
10. Кемени Дж., Снелл Дж. Кибернетическое моделирование: пер. с англ. М.: Советское радио, 1972. 192 с.

Automation of the Smart-Education Trajectory Choice

The basic Smart-education elements are considered in the article: university facilities; information sources types; the way of the educational process organization; the number of educational group; technology of an information presentation; technology of the organization of current and total knowledge control; knowledge presentation speed according to a particular student; the sequence and the number of the simultaneous studied objects; students preferences. The questions of trajectory choice in the Smart-education environment are discussed. The authors give an approximate content of a possible education trajectory. The article offers a technique of the education trajectory choice based on the decision-making support methods. The specified technique allows using both expert's knowledge and experience in various fields of the organization and ensuring educational process, and requirements of students. This article considers both group and individual decision-making methods. The authors prove the method choice for the task solution. The article shows an example of the minimum distance method application for the expert's opinions coordination while choosing the alternative educational trajectories. The example of the hierarchy analysis method application while choosing one of the alternative educational trajectories recommended by experts is shown. The offered technique allows to automate the process of the most rational education trajectory choice and to reduce decision-making time and costs.

Keywords: Smart-education, education trajectory, decision-making support methods, minimum distance method

References

1. Uskov V. L., Ivannikov A. D., Uskov A. V. Perspektivnye tehnologii dlja jelektronnoogo obrazovaniya (Perspective technologies for electronic education), *Informacionnye tehnologii*, 2007, no. 2, pp. 32–38.
2. Tihomirova N. V. Global'naja strategija razvitiya Smart-obshchestva. MESI na puti k Smart-universitetu (Global strategy of Smart-society development. MESI on the way to Smart-university), URL: <http://smartmesi.blogspot.ru/2012/03/smart-smart.html> (data obrashhenija: 21.01.2016).
3. Shakirova Z. H., Ahmetshin D. A. Smart sistemy v sfere vysshego obrazovaniya (Smart systems in higher education), *Sovremennye nauchnye issledovaniya i innovacii*, 2015, no. 7, URL: <http://web.snauka.ru/issues/2015/07/56409> (data obrashhenija: 20.01.2016).
4. Bulgakov M. V., Gridina E. G., Ivannikov A. D., Staryh V. A. Federal'naja sistema informacionno-obrazovatel'nyh resursov (Federal system of information and education resources), *Informacionnye resursy Rossii*, 2009, no. 2, pp. 25–27.
5. Efremova N. Testovyy kontrol' v obrazovanii (Test control in education). URL: http://www.xliby.ru/nauchnaja_literatura_prochee/testovyi_kontrol_v_obrazovanii/index.php (data obrashhenija: 10.02.2016).
6. Larichev O. I. Teorija i metody prinjatija reshenij, a takzhe Hronika sobytij v Volshebnyh stranah (The decision-making theory and methods, and also Chronicle of events in the Magic countries): uchebnik, izd. 2-e, pererab. i dop., Moscow, Logos, 2002, 392 p.
7. Lobanov A. A. Metod predpochtenij kak instrument podderzhki prinjatija reshenij. (Method of preferences as instrument of decision-making support), *Perspektivy nauki i obrazovaniya*, 2015, no. 2 (14), pp. 36–43.
8. Ivanova E. M., Vishnekov A. V., Erohin V. V. Avtomatizacija procedury vybora mikrokontrollera (Automation of the microcontroller choice procedure), *Nano- i mikrosistemnaja tehnika*, 2014, no. 7, pp. 14–21.
9. Ivanova E. M., Vishnekov A. V., Erohin V. V. Verifikacija SNK: vybor strategii (SOK verification: the choice of strategy), *Nano- i mikrosistemnaja tehnika*, 2014, no. 12, pp. 30–36.
10. Kemeny J. G., Snell J. L., Kiberneticheskoe modelirovanie (Cybernetic modeling): per. s angl. — Moscow, Sovetskoe radio, 1972, 192 p.



С 16 по 18 ноября 2016 г.
в г. Тула на базе Тульского ГУ состоится
Международная научно-техническая конференция
ИНТЕЛЛЕКТУАЛЬНЫЕ
И ИНФОРМАЦИОННЫЕ СИСТЕМЫ
(Интеллект 2016)



ОСНОВНЫЕ ВОПРОСЫ, ПЛАНИРУЕМЫЕ ДЛЯ ОБСУЖДЕНИЯ

- Интеллектуальные системы поддержки принятия решений.
- Управление параллельными и распределенными информационными процессами.
- Методы цифровой обработки сигналов и изображений.
- Системы классификации и распознавания образов.
- Когнитивные технологии.
- Роевые технологии.

Подробную информацию о конференции см. на сайте:
<http://tsu.tula.ru/science/conferences/>

ЦИФРОВАЯ ОБРАБОТКА СИГНАЛОВ И ИЗОБРАЖЕНИЙ DIGITAL PROCESSING OF SIGNALS AND IMAGES

УДК 004.932.1

А. В. Мальцев, канд. физ.-мат. наук, ст. науч. сотр., avmaltcev@mail.ru,

М. В. Михайлюк, д-р физ.-мат. наук, зав. отделом, mix@niisi.ras.ru,

П. Ю. Тимохин, науч. сотр., webpismo@yahoo.de,

М. А. Торгашев, канд. физ.-мат. наук, зав. сектором, mtorg@mail.ru

Федеральный научный центр Научно-исследовательский институт системных исследований
Российской академии наук, г. Москва

Технология разделения видеопотоков в дистанционном обучении с помощью нескольких устройств Kinect

В дистанционном обучении одной из важных задач является передача по сети качественного видеопотока в реальном времени. В работе рассматривается технология, основанная на применении нескольких устройств Microsoft Kinect, в которой исходный видеопоток разделяется на несколько потоков (доска, лектор и маска лектора). Они передаются по сети и объединяются на пользовательском компьютере, что позволяет достичь меньшего битрейта, чем при передаче исходного видеопотока.

Ключевые слова: видеопоток, сегментация, Kinect, дистанционное обучение, карта глубины, сжатие, реальное время

Введение

Одной из важных задач дистанционного обучения является формирование и передача по информационной сети качественного видеопотока в масштабе реального времени. Этот масштаб предполагает работу с видеопотоком, который имеет частоту не менее 25 кадров в секунду, вследствие чего необходимо использовать эффективные алгоритмы сжатия и распаковки видеопотока. Одним из методов решения данной задачи является разделение видеопотока на основе сегментации [1, 2]. Например, если лектор читает лекцию с использованием электронной доски или обычной доски и маркеров, то исходный поток можно разделить на два: первый будет содержать только доску, а второй — только лектора. Содержимое доски во время лекции меняется медленно, поэтому она будет сжиматься сильнее. Вместе с тем лектора можно визуализировать с пониженным качеством. В результате такая работа с отдельными потоками будет более эффективной, чем с одним исходным потоком.

В работе [3] для разделения видеопотоков предложено использовать устройство *Kinect*. Разделение потоков проводится с использованием карты глубины, формируемой *Kinect* в масштабе реального времени. Так как лектор всегда находится перед доской, то его глубина (расстояние до камеры) меньше глубины доски. Разделенные потоки требуют корректировки, а именно, устранения парал-

лакса между изображениями, выполнение обрезания по границе эффективной области, заполнения "дыр" в карте глубины и т. д. Такая технология позволяет обеспечить угол зрения (по горизонтали) $57,5^\circ$ и разрешение 640×480 (1304×1080 для *Kinect* 2.0). Для сжатия полученных потоков в работе [4] предложено использовать кодеки стандартов h.264/h.265, работающие как на центральном процессоре, так и поддерживаемые аппаратно на современных GPU компании NVIDIA.

Поскольку ширина доски может достигать нескольких метров и углы обзора камеры и инфракрасного сенсора *Kinect* фиксированы, то для охвата доски с помощью описанного выше подхода необходимо устанавливать *Kinect* достаточно далеко. Однако при этом уменьшается детализация получаемых с *Kinect* изображений, так как на один пиксель приходится большая площадь доски. Кроме того, существенно увеличивается площадь неэффективной области выше и ниже доски, которую придется отсекать для сокращения времени сжатия и пересылки по сети. Для решения этой проблемы в данной работе предлагается мультикинектная технология разделения видеопотоков, основанная на применении нескольких устройств *Kinect* и позволяющая существенно увеличить угол зрения без ухудшения детализации получаемого видеопотока. Далее рассмотрим суть технологии на примере установки, состоящей из двух таких устройств.

1. Схема экспериментальной установки

Рассмотрим принципиально возможные схемы установок двух устройств *Kinect*, позволяющие увеличить общее разрешение результирующего видеопотока (рис. 1).

На рис. 1, а приведена схема со смещением камер по горизонтали и их разворотом на центральную область. Можно заметить, что эта схема не обеспечивает существенного увеличения разрешения, поскольку камеры направлены на одну и ту же область со значительным перекрытием. Недостатком схемы также является то, что между изображениями имеется значительный горизонтальный параллакс, устранение которого затруднительно.

Рис. 1, б иллюстрирует схему параллельного размещения камер со смещением по горизонтали. Проблемой данной схемы также является значительный горизонтальный параллакс. Кроме того, рабочая область, в которой может находиться лектор, ограничена, в ней имеются "мертвые" зоны, при нахождении в которых человек не попадает в поле зрения ни одной из камер. Для нивелирования этой проблемы требуется значительное сближение камер, что в свою очередь, уменьшает эффективное разрешение.

На рис. 1, в показана схема размещения устройств *Kinect* вблизи друг друга с их разворотом относительно вертикальной оси. В данной схеме более эффективно используется общее разрешение камер и существенно увеличено общее поле зрения. Мертвые зоны могут быть уменьшены за счет максимально близкого размещения камер. Горизонтальный параллакс при этом также уменьшается. В качестве недостатка схемы можно назвать перспективные искажения. Это вызвано тем, что экраны плоскости камер развернуты относительно друг друга, что приводит к изломам прямых линий на границах изображений отдельных камер. Отметим, однако, что этот недостаток может быть ус-

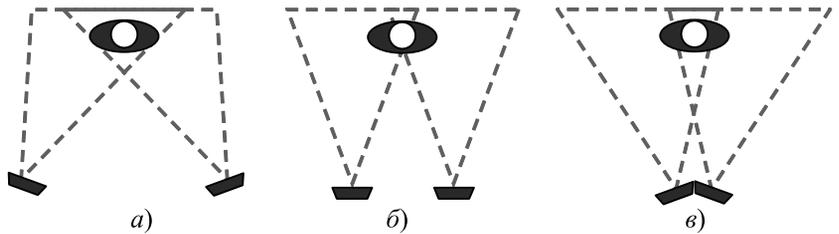


Рис. 1. Возможные схемы установки из нескольких устройств *Kinect*

пешно преодолен с помощью коррекции перспективы, что будет показано далее. В данной работе за основу была выбрана схема, изображенная на рис. 1, в. Схема была немного модифицирована (рис. 2). Для полного устранения горизонтального параллакса камеры размещены строго друг над другом и развернуты в разные стороны с небольшим перекрытием полей зрения. При этом возникает небольшой вертикальный параллакс, который не критичен и в дальнейшем может быть скомпенсирован.

Каждое из устройств *Kinect* работает со своей половиной доски, формируя левый и правый каналы данных, которые обрабатываются независимо. Затем оба канала объединяются в один, содержащий широкоформатные RGB-изображение и черно-белую маску. На основе последней проводится выделение из RGB-изображения кадра потока с лектором и кадра потока с доской, далее сжимаемых и передаваемых по сети на клиентский компьютер.

Далее подробнее рассмотрим получение и обработку видеопотоков с помощью предложенной мультикинектной технологии.

2. Получение и обработка видеопотоков

2.1. Получение данных с устройств *Kinect*

Вначале необходимо получить с нашей установки кадры изображения RGB и карты глубины (DM, depth map) отдельно для левого и правого каналов (рис. 3, см. четвертую сторону обложки). В данной работе это осуществляется путем опроса обоих устройств *Kinect* с использованием *Kinect SDK* [5]. Каждое из запрашиваемых изображений хранится в своем собственном буфере соответствующего устройства. Доступ к этим буферам осуществляется независимо друг от друга. Поэтому для выполнения синхронизации всех четырех изображений (т. е. обеспечения соответствия их одному и тому же моменту съемки) необходимо перед считыванием данных заблокировать все буферы для записи. Это можно сделать путем последовательного вызова для каждого из четырех буферов асинхронной функции *LockRect* (см. [5]).

Заметим, что в каждом из каналов существует параллакс между считанными RGB- и DM-изображениями, возникающий вследствие различного про-



Рис. 2. Размещение двух устройств *Kinect*

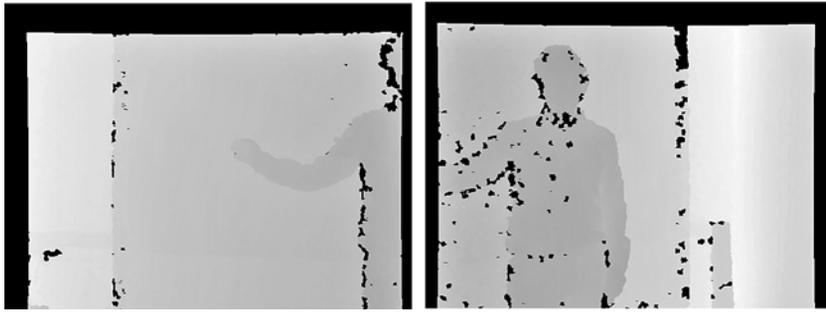


Рис. 4. Карты глубины, приведенные к областям видимости RGB-камер

странственного размещения RGB-камеры и инфракрасного сенсора глубины внутри устройства *Kinect*. К тому же векторы взгляда и углы раствора камеры и IR-сенсора совпадают только с некоторой погрешностью. Параллакс выражается в том, что один и тот же объект на обоих изображениях располагается со смещением (как горизонтальным, так и вертикальным), варьирующимся в зависимости от расстояния от объекта до *Kinect*, а также в наличии областей, где информация получается только из одного источника (RGB-камеры или IR-сенсора). Метод компенсации имеющегося параллакса, основанный на коррекции карты глубины с учетом значения расстояния в каждом пикселе, описан нами в работе [3]. В результате его применения к левому и правому каналам мы получим скорректированные карты глубины (рис. 4), приведенные к областям видимости соответствующих RGB-камер.

2.2. Восстановление утраченной информации о глубине

Полученная в каждом канале карта глубины содержит области, в которых устройство *Kinect* не смогло распознать глубину — так называемые "дыры" (на карте глубины они отмечены черным цветом). Для полноценной работы с этими картами необходимо восстановить утраченную информацию в данных областях. В нашей задаче (отделение силуэта лектора от фона) при таком восстановлении целесообразно использовать лишь значения глубины, соответствующие переднему и заднему плану. Эффективно реализовать такое восстановление по-



Рис. 5. Карты глубины с восстановленными данными

могает выявление характера расположения дыр на изображении. В работе [3] описаны основные типы "дыр", появление которых возможно при работе с *Kinect*. Для рассматриваемой мультикинектной системы характерно образование дыр I-го типа (инфракрасная тень от облучаемого объекта), дыр II-го типа (поглощение или рассеяние IR-излучения поверхностями объектов), а также дыр IV-го типа (интерференция IR-излучения от нескольких источников). По своей природе отдель-

ная типовая дыра располагается либо на облучаемом объекте, либо около него (за исключением дыр IV-го типа, место появления которых непредсказуемо). Однако на практике возникают случаи, когда в карте глубины дыры различных типов соприкасаются друг с другом, образуя дыры сложных форм ("смешанные" дыры), которые покрывают объекты и переднего, и заднего планов. Разбив такие дыры в местах их сочленения, мы получаем вместе с остальными дырами набор одноплановых дыр, каждую из которых заливаем значением глубины, выбранным из пикселей ее внешнего контура. Для этого предлагается следующий алгоритм восстановления данных о глубине.

1. Разбиение "смешанных" дыр и устранение мелких дыр с помощью медианного фильтра с крупным окном.
2. Добавление к карте глубины опоясывающего "бордюра" для корректного выделения и обработки контуров дыр, прилегающих к границам карты глубины (ширина бордюра — 2 пикселя, цвет — максимальное значение диапазона глубины).
3. Выделение внешних контуров дыр с помощью алгоритма, описанного в работе [6].
4. Вычисление цвета (глубины) заливки для каждого контура (более подробно описано в работе [3]).
5. Заливка внутренних областей контуров дыр.
6. Удаление опоясывающего бордюра.
7. Сглаживание границ силуэта лектора с помощью медианного фильтра с малым окном.

Во избежание искажения контура лектора шаги 1—6 алгоритма выполняются над копией исходной карты глубины, из которой затем восстановленные значения глубин переносятся в исходную карту глубины. В рассматриваемой мультикинектной системе разработанный алгоритм применяется по очереди к карте глубины левого и правого канала (рис. 5). В целях сокращения общего времени обработки обеих карт выполнение шагов 1, 2, 6 и 7 алгоритма реализуется на графическом процессоре с помощью технологии распараллеливания вычислений OpenCL. По сравнению с реализацией алгоритма полностью на

CPU предложенная реализация позволила выполнять восстановление данных в картах глубин приблизительно в 3 раза быстрее, при этом время обработки кадра составило около 15 мс.

2.3. Отделение силуэта лектора от фона

В контексте рассматриваемой задачи эффективного сжатия видеопотока отделение силуэта лектора от фона необходимо для достижения двух целей: удаление лектора из общего видеопотока для формирования отдельного потока с доской (высокого качества) и удаление фона из общего видеопотока для формирования отдельного потока с лектором (пониженного качества сжатия). Несмотря на кажущуюся взаимоисключаемость получаемых в результате видеопотоков, эксперименты, проведенные в настоящей работе, показывают, что для эффективного сжатия видеопотока с доской необходимо кроме силуэта лектора также удалять из общего видеопотока и тень от лектора. Ввиду этого в рассматриваемой мультимедийной системе для каждого канала мы формируем пару масок: маску лектора и маску лектора с его тенью.

Построение маски лектора выполняется путем сравнения восстановленной карты глубины, снятой с лектором в кадре, с картой глубины, снятой до входа лектора в кадр (картой пороговых значений). Такой подход позволяет учитывать разброс глубин вдоль плоскости доски, возникающий в результате направления устройств *Kinect* к облучаемой сцене под косым углом, а также избегать ложного включения в передний план выступающих элементов фона (доски). Более подробно формирование и применение карты пороговых значений описано в работе [3]. Отметим, что в отличие от установки с одним устройством *Kinect*, рассмотренной в работе [3], в мультимедийной системе при формировании карты пороговых значений для одного канала необходимо отключать ИР-излучатель устройства *Kinect* другого канала. Это позволяет предотвратить запись в формируемую карту пороговых значений ложных значений глубин, которые могут возникнуть в области наложения ИР-излучения от двух одновременно работающих излучателей *Kinect*. На рис. 6 показаны сформированные для двух каналов карты пороговых значений, а на рис. 7 — маски лектора, построенные с помощью этих карт.

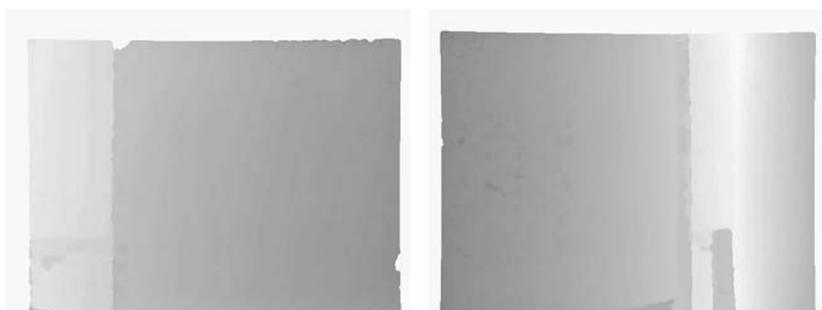


Рис. 6. Карты пороговых значений



Рис. 7. Маска лектора для левого и правого каналов



Рис. 8. Маска лектора с тенью для левого и правого каналов

Маска лектора с тенью создается путем добавления к полученной выше маске лектора маски тени, выделенной из RGB-изображения, считанного с устройства *Kinect*. Для выделения тени в настоящей работе используется алгоритм цветовой сегментации фона/переднего плана [7]. Тенью полагаются пиксели, цвет каждого из которых является более темной версией цвета фона в этом участке изображения относительно заданного пользователем порогового значения [8]. На рис. 8 проиллюстрированы маски лектора с тенью, построенные для левого и правого канала рассматриваемой мультимедийной системы.

2.4. Коррекция перспективы

Поскольку направления взгляда камер и сенсоров устройств *Kinect* в нашем случае не перпендикулярны доске, то в обоих каналах будет наблюдаться перспективное искажение получаемых с устройств

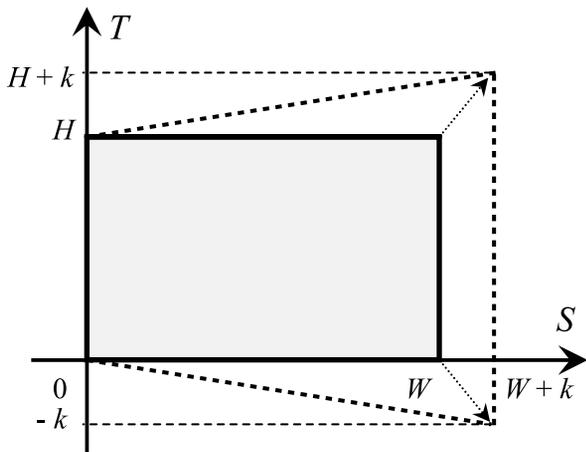


Рис. 9. Коррекция перспективы в правом канале

RGB-изображений и черно-белых масок, подготавливаемых на основе DM-карт. Оно возникает вследствие того, что RGB-камеры и IR-сенсоры осуществляют перспективное проецирование своей трехмерной рабочей области на картинную плоскость. Результатом этого является искажение изображения лектора и информации, находящейся на доске. Например, нарисованный на доске прямоугольник будет выглядеть трапецией.

Для решения этой проблемы мы предлагаем использовать цифровую коррекцию перспективы для RGB-кадров и кадров маски в каждом из каналов. Ее суть заключается в вертикальном растяжении изображения по двум его углам на такое число k пикселей (для каждого угла), которое обеспечивает устранение на этом изображении схождения начерченных на доске горизонтальных линий, в реальности являющихся параллельными. Коэффициент растяжения для столбцов пикселей при этом линейно уменьшается в сторону противоположных углов, где становится равным 1. Чтобы не получить ложное увеличение вертикального размера объектов в кадре, выполним также масштабирование изображения, получившегося после коррекции перспективы, в $1 + k/W$ раз по горизонтали, где W — ширина исходного кадра в пикселях. В результате проведенной коррекции мы получаем кадр, форма

которого имеет вид трапеции. Для дальнейшей работы необходимо провести его кадрирование в границах исходного изображения.

На рис. 9 показан пример коррекции перспективы для кадра размером $W \times H$ пикселей из правого канала. Растяжение в данном случае необходимо выполнять по крайним правым углам. Результат применения коррекции показан на рис. 10 на примере RGB-изображений для обоих каналов.

2.5. Объединение каналов

После того как изображения RGB и масок для обоих каналов подготовлены и скорректированы, необходимо провести объединение этих каналов в один, т. е. склеить левые и правые части RGB-кадров и кадров масок. Для сглаживания перехода между двумя частями результирующего RGB-изображения склеим эти части с пересечением шириной в n пикселей (рис. 11). Цвет C каждого пикселя объединенного RGB-кадра в области пересечения получается путем интерполяции цветов C_L и C_R соответствующих ему пикселей левого и правого каналов:

$$C = (1 - \alpha)C_L + \alpha C_R, \quad \alpha = i/n,$$

где i — номер столбца пикселей области пересечения, считая от ее левой границы. Значение n выбираем эмпирическим путем и в нашем случае оно составляет 20 пикселей.

Маски из левого и правого каналов также объединим с пересечением. Однако в данном случае выполнять интерполяцию нельзя, поскольку она может привести к появлению пикселей, имеющих цвет в градациях серого, что недопустимо для черно-белой маски. Поэтому цвет C пикселя объединенной маски в области пересечения будем считать равным белому, если хотя бы один из двух цветов соответствующих пикселей из масок левого и правого каналов — белый. На рис. 12 показаны объединенные изображения для RGB-кадра и кадров масок.

Итоговое разрешение RGB-кадров и кадров масок с учетом выбранного нами значения n составляет 1100×480 (2240×1080 для *Kinect 2.0*), а угол обзора равен 105° .

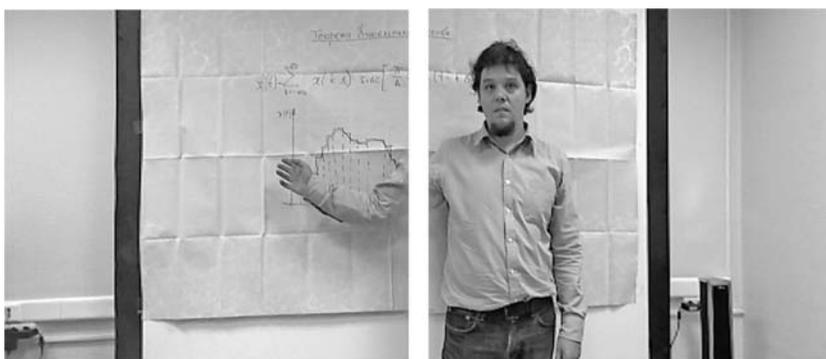


Рис. 10. RGB-изображения после коррекции перспективы

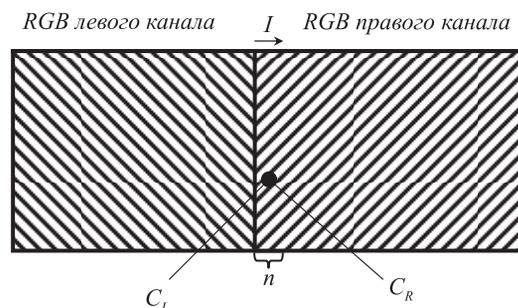


Рис. 11. Объединение RGB-кадров двух каналов

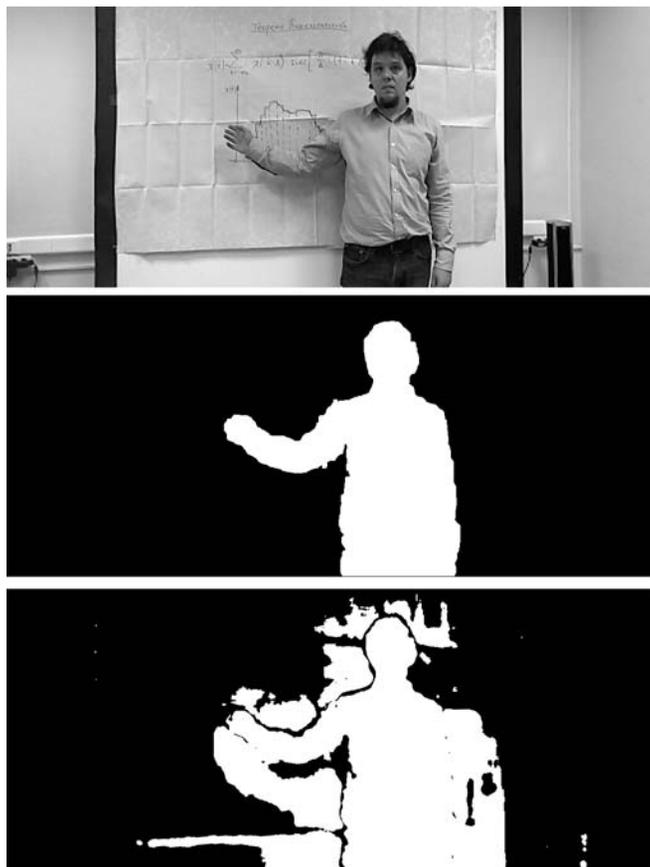


Рис. 12. RGB-кадры и кадры масок после слияния каналов

С помощью полученной объединенной маски лектора выделяем из объединенного RGB-изображения кадр с изображением лектора на черном фоне, а с помощью объединенной маски лектора с тенью — кадр, содержащий только изображение доски (рис. 13, см. четвертую сторону обложки). При формировании кадра с доской области с недостающими данными (после отделения лектора) мы восполняем из предыдущих кадров, как описано в работе [3].

3. Сжатие видеопотоков

После обработки изображений с двух устройств *Kinect* получаем три видеопотока — RGB-потоки доски и лектора, а также видеопоток, содержащий маску лектора. Рассмотрим используемые в данном исследовании методы сжатия полученных данных. Для сжатия RGB-кадров были использованы наиболее современные и эффективные на настоящий момент алгоритмы сжатия видеoinформации с потерями на основе стандартов AVC (Advanced Video Coding, обозначается также H.264) и HEVC (High Efficiency Video Coding, имеет также обозначение H.265). Сжатие видео с использованием данных стандартов было реализовано двумя способами: с использованием кодеков x264/x265, работающих на центральном процессоре, а также с

использованием аппаратных кодеков компании NVIDIA. В исследовании [4] было установлено, что программные кодеки x264/x265 выигрывают по эффективности сжатия по сравнению с аппаратными кодеками NVIDIA, однако существенно проигрывают по скорости и не всегда могут обеспечить режим реального времени. Таким образом, было установлено, что при наличии аппаратной поддержки целесообразней использовать аппаратные кодеки, особенно кодек H.265, который обеспечивает существенно лучший (до 2—3 раз) коэффициент сжатия по сравнению с H.264.

При сжатии видеопотока маски, в отличие от сжатия потоков лектора и фона, желательно обеспечить отсутствие значимых потерь, поскольку в противном случае могут наблюдаться заметные артефакты на границах силуэта лектора при совмещении изображений фона и лектора. С учетом природы обрабатываемых бинарных изображений, которые представляют собой белый силуэт лектора на черном фоне, был выбран метод кодирования маски на основе контуров, задающих силуэт лектора [4]. Поиск контуров выполняется на основе метода Сузуки и Абе [6], позволяющего получить иерархическую информацию о контурах. Далее выполняется предварительная обработка, позволяющая выделить только наиболее значимые контуры и отфильтровать шумы. Для описания контуров был использован цепочный код DDCC (*directional difference chain code*), основанный на кодировании разности направлений. Для сжатия получаемых цепочных кодов был применен алгоритм Deflate. Эксперименты по сжатию тестовых видеопоследовательностей маски лектора позволили получить оценку битрейта на кодирование маски в 85—160 кбит/с.

Заключение

В работе предложен метод мультикинектного разделения видеопотоков для задач дистанционного обучения. Использование нескольких устройств *Kinect* позволяет существенно увеличить поле зрения съемки с сохранением детализации изображения. Разделение потоков выполняется на основе сегментации по карте глубины. Метод включает компенсацию параллакса между изображениями, устранение "дыр" в карте глубины, удаление теней, коррекцию перспективы, объединение каналов и результирующее разделение потоков. Эксперименты по оценке эффективности сжатия разделенных потоков по сравнению с исходным потоком позволили получить оценку коэффициента сжатия в 1,6...1,9. Высокие коэффициенты сжатия обеспечиваются за счет того, что один из потоков (содержащий доску) близок к статическому, поэтому сжимается очень эффективно, второй поток (содержащий лектора) сжимается с немного пониженным качеством, а третий (содержащий маску) эффективно кодируется на основе выделения контура лектора.

Список литературы

1. **Комаров Е. В., Лосев А. С., Орлов Н. С., Чистяков Д. В.** Ближайшее будущее научно-образовательного видео // Информационные технологии в обеспечении федеральных государственных образовательных стандартов: сб. трудов. 2014. Т. 1. С. 307—312.
2. **Wang C., Chan S. C., Ho C. H., Liu A. L., Shum H. Y.** A real-time image-based rendering and compression system with Kinect depth camera // Proc. of the 19th International Conference on Digital Signal Processing. 2014. P. 626—630.
3. **Кушниренко А. Г., Мальцев А. В., Михайлюк М. В., Прилико А. А., Тимохин П. Ю., Торгашев М. А.** Метод разделения видеопотоков в задачах дистанционного обучения // *Mathematica Montisnigri*. 2015. Vol. XXXII. P. 149—157.
4. **Кушниренко А. Г., Мальцев А. В., Михайлюк М. В., Прилико А. А., Тимохин П. Ю., Торгашев М. А.** Сжатие разделенных видеопотоков в задачах дистанционного обучения // *Известия Академии инженерных наук им. А. М. Прохорова*. 2015. № 2. С. 3—10.
5. **Kinect** for Windows SDK. URL: <https://msdn.microsoft.com/enus/library/hh855347.aspx> (date of review: 01.02.2016).
6. **Suzuki S.** et al. Topological structural analysis of digitized binary images by border following // *Computer Vision, Graphics, and Image Processing*. 1985. Vol. 30, N. 1. P. 32—46.
7. **Zivkovic Z.** Improved adaptive Gaussian mixture model for background subtraction // *International Conference Pattern Recognition*. 2004. Vol. 2. P. 28—31.
8. **Prati A., Mikic I., Trivedi M. M., Cucchiara R.** Detecting moving shadows: Algorithms and evaluation // *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2003. Vol. 25, N. 7. P. 918—923.

A. V. Maltsev, Ph. D. (Mathematics), Senior Researcher, avmaltcev@mail.ru,
M. V. Mikhaylyuk, Dr. Sc. (Mathematics), Head of Department, mix@niisi.ras.ru,
P. Yu. Timokhin, Scientific Researcher, webpismo@yahoo.de,
M. A. Torgashev, Ph. D. (Mathematics), Head of Sector, mtorg@mail.ru
Scientific Research Institute of System Analysis (Russian Academy of Sciences)

The Technology of Video Stream Separation by Means of Multiple Kinect Devices in Distance Education

One of the important issues in the field of distance education is to transmit over network a qualitative video stream in real-time mode. In this paper we propose the technology based on using several Microsoft Kinect devices, in which initial video stream is separated into three new streams. The first of them contains only the board, the second — the lecturer, the third — a mask for selecting the lecturer. Separate compression, transmission and merging of these streams into one on user's computer allow us to achieve lower bitrate compared to direct transmission of the source video. Proposed compression methods work in real-time mode and provide high compression ratio for video transmission over data networks with low bandwidth. Using multiple Kinect devices in our technology allows to significantly increase an angle of view compared to single device.

Keywords: video stream, segmentation, Kinect, distance education, depth map, compression, real-time

References

1. **Komarov E. V., Losev A. S., Orlov N. S., Chistyakov D. V.** Blizhayshee budushchee nauchno-obrazovatel'nogo video, *Informatsionnye tekhnologii v obespechenii federal'nykh gosudarstvennykh obrazovatel'nykh standartov. Sbornik trudov*, 2014, vol. 1, pp. 307—312. [Near future of scientific-educational video // In Proceedings Information technologies in providing of federal state educational standards].
2. **Wang C., Chan S. C., Ho C. H., Liu A. L., Shum H. Y.** A real-time image-based rendering and compression system with Kinect depth camera, *Proceedings of the 19th International Conference on Digital Signal Processing*, 2014, pp. 626—630.
3. **Kushnirenko A. G., Maltsev A. V., Mihajljuk M. V., Prilikov A. A., Timokhin P. Yu., Torgashev M. A.** Metod razdeleniya videopotokov v zadachakh distantsionnogo obucheniya, *Mathematica Montisnigri*, 2015, vol. XXXII, pp. 149—157. [The method for separation of video streams in distant education // *Mathematica Montisnigri*].
4. **Kushnirenko A. G., Maltsev A. V., Mihajljuk M. V., Prilikov A. A., Timokhin P. Yu., Torgashev M. A.** Szhatie razdelennykh video potokov v zadachakh distantsionnogo obucheniya, *Izvestiya Akademii inzhenernykh nauk im. A. M. Prokhorova*, 2015, no. 2, pp. 3—10. [The compression of separated video streams for distance education tasks // *News of the Academy of Engineering Sciences named after A. M. Prokhorov*].
5. **Kinect** for Windows SDK, <https://msdn.microsoft.com/enus/library/hh855347.aspx> (date of review: 01.02.2016)].
6. **Suzuki S.** et al. Topological structural analysis of digitized binary images by border following, *Computer Vision, Graphics, and Image Processing*, 1985, vol. 30, no. 1, pp. 32—46.
7. **Zivkovic Z.** Improved adaptive Gaussian mixture model for background subtraction, *International Conference Pattern Recognition*, 2004, vol. 2, pp. 28—31.
8. **Prati A., Mikic I., Trivedi M. M., Cucchiara R.** Detecting moving shadows: Algorithms and evaluation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2003, vol. 25, no. 7, pp. 918—923.

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В БИМЕДИЦИНСКИХ СИСТЕМАХ INFORMATION TECHNOLOGIES IN BIOMEDICAL SYSTEMS

УДК 615.478:681.2

Н. Т. Абдуллаев¹, канд. техн. наук, доц., e-mail: a.namik46@mail.ru,
Н. Я. Мамедов², канд. техн. наук, доц., e-mail: nr.nuraddin47@mail.ru,
Г. С. Агаева², диссертант, e-mail: gunel_aso@yahoo.com

¹ Азербайджанский технический университет

² Азербайджанский государственный университет нефти и промышленности, г. Баку

Применение метода спектрального анализа для дифференциальной диагностики заболевания органов желудочно-кишечного тракта

Рассматривается возможность использования предложенного алгоритма спектрального анализа измерительных сигналов для дифференциальной диагностики функционального состояния органов желудочно-кишечного тракта. Исследованы электрогастроэнтерографические сигналы для нормального состояния при язвенных поражениях желудка и двенадцатиперстной кишки. Для дифференциальной диагностики этой болезни используются средние значения комплексных коэффициентов Фурье, приведены зависимости этих коэффициентов от числа гармоник исследуемого сигнала. Предложены структуры искусственных нейронных сетей для классификации образцов.

Ключевые слова: желудочно-кишечный тракт, электрогастроэнтерографические сигналы, спектральный анализ, коэффициенты Фурье, нейронная сеть

Введение

Спектральный анализ электрофизиологических сигналов является одним из самых распространенных методов обработки измерительных сигналов в целях определения их особенностей и выявления информативных диагностических признаков.

В работе [1] для периодического непрерывного сигнала $y = f(t)$, характеризуемого выражением

$$f(t) = \sum_{n \rightarrow -\infty}^{\infty} C_n e^{jn\omega t},$$

где $\omega = 2\pi/T$ (T — период сигнала), комплексные коэффициенты Фурье C_n определяются как

$$C_n = \frac{1}{T} \int_0^T f(t) e^{-jn\omega t} dt.$$

Далее, используя заданную зависимость $y = f(t)$, определим новую неявную функцию y в зависимости от t :

$$y = f(t + \alpha y), \quad (1)$$

где α — малая постоянная величина.

Зависимость (1) определяет y как однозначную целевую функцию от t ; если $f(t)$ дифференцируема, имеет ограниченную первую производную и удов-

летворяет условию существования и единственности неявной функции, то в этом случае выражение (1) позволяет определить некоторую однозначную функцию вида

$$y = F(t, \alpha). \quad (2)$$

Тогда для преобразованной зависимости (2) разложение в ряд Фурье будет иметь следующий вид:

$$F(t, \alpha) = \sum_{n \rightarrow -\infty}^{\infty} G_n e^{-jn\omega t}. \quad (3)$$

Поскольку для определения коэффициентов G_n не всегда представляется возможным зависимость (1) представить в явном виде (2), то выражение (1) следует представить в параметрической форме.

Для преобразованной функции $F(t, \alpha)$ коэффициенты Фурье G_n были определены как [1]

$$G_0 = \frac{1}{T} \int_0^T f(t + \alpha y) dt = \frac{1}{T} \int_0^T f(t) dt;$$

$$G_n = \frac{1}{T} \int_0^T f(t + \alpha y) e^{-jn\omega t} dt = \frac{1}{2\pi j n \alpha} \int_0^T e^{-jn\omega(t - \alpha f(t))} dt, \quad n \neq 0. \quad (4)$$

В работах [1, 2] было показано, что при соответствующем выборе числа α расчет коэффициентов C_n можно заменить расчетом коэффициентов G_n , так как разность $F(t, \alpha) - f(t)$ невелика. Используя выражение (4) и непосредственно вычисляя предел $\lim_{\alpha \rightarrow 0} G_n$, получаем $\lim_{\alpha \rightarrow 0} G_n = C_n$. Кроме того, из соотношения (4) следует, что $G_0 = C_0$; это означает, что при выполнении преобразования (1) постоянная составляющая сигнала не меняется. При этом выражения для расчета комплексных коэффициентов Фурье G_n позволяют исключить необходимость операций умножения исходной функции на ортогональные составляющие функции $e^{-jn\omega t}$, используемых в алгоритме БПФ. Это, в свою очередь, упрощает расчеты по предложенному алгоритму вычисления коэффициентов Фурье.

Если сигнал $f(t)$ представлен дискретными значениями в M точках отрезка $[0, T]$, что характерно для цифровой обработки сигналов, то для вычисления комплексных коэффициентов Фурье может быть использовано выражение

$$G_n = \frac{T}{2\pi j M n \alpha} \sum_{i=1}^M e^{-jn\omega \alpha_i}, \quad (5)$$

где $\alpha_i = t_i - \alpha f(t_i)$; $t_i = i \frac{T}{M}$ — моменты отсчета мгновенных значений сигнала.

Эффективность рассмотренного алгоритма повышается, если исследуемый сигнал относится к классу низких и инфранизких частот.

Постановка проблемы

Поскольку электрофизиологические сигналы относятся к классу низких и инфранизких частот [3], то предлагается использование комплексных коэффициентов Фурье G_i , $i = \overline{1, n}$, в качестве информативных параметров для дифференциальной диагностики функционального состояния исследуемого органа человеческого организма. В качестве биоэлектрических сигналов были выбраны электрогастроэнтерографические сигналы, характеризующие электрическую активность желудка и кишечника [4–8]. Специфичность частот базисных ритмов

Таблица 1

Частоты, характерные для различных отделов желудочно-кишечного тракта

Отдел желудочно-кишечного тракта	Частота, Гц	Номер частотного отдела
Толстая кишка	0,01...0,03	5
Желудок	0,03...0,07	1
Подвздошная кишка	0,07...0,13	4
Тощая кишка	0,13...0,18	3
Двенадцатиперстная кишка	0,18...0,25	2

для каждого отдела желудочно-кишечного тракта легла в основу электрофизиологических методов исследования его различных отделов (табл. 1).

Результаты вычислительного эксперимента и их анализ

В качестве исходных данных были рассмотрены реально измеренные тошачковые сигналы для нормального состояния и язвенной болезни желудка. Электрогастрограммы соответствующих состояний органов желудочно-кишечного тракта были приведены в статье [9]. Обработке подвергались оцифрованные значения этих сигналов с частотой дискретизации 100 Гц. Расчеты были повторены по 10 пациентам одной гендерной группы (мужчины) примерно одной возрастной группы (35...47 лет) с указанными диагностическими заключениями, подтвержденными эндоскопическими и рентгенологическими обследованиями.

Поскольку электрогастроэнтерографический сигнал является нестационарным, его спектр не следует рассчитывать за весь временной интервал исследования (в нашем случае $T = 16$ мин). Вместо этого выбирается отрезок сигнала постоянной длительности (например, четырехминутный) и методом "скользящего окна" со сдвигом в одну минуту просматривается весь интервал измерения. При этом спектр сигнала рассчитывается для каждого временного отрезка. Таким образом метод "скользящего окна" помогает сглаживанию тренда сигнала. Выбор отрезка такой длительности обусловлен тем, что сокращения различных органов желудочно-кишечного тракта являются низкочастотными. Сложный характер электрогастроэнтерографического сигнала является результатом сложения многих гармоник. Выбор такой длительности временного отрезка обеспечивает хорошее разрешение по частоте спектральных составляющих, что очень важно при расчетах информативных диагностических показателей.

Распределение усредненных значений комплексных коэффициентов Фурье по гармоникам, рассчитанным согласно выражению (5), приведено в табл. 2 и на рис. 1.

Аналогичные расчеты были проведены по электрогастрограмме для нормального состояния и язвенной болезни двенадцатиперстной кишки. Для этих состояний двенадцатиперстной кишки распределение усредненных значений комплексных коэффициентов Фурье приведены в табл. 3 и рис. 2.

Из представленных графиков можно судить о различии значений комплексных коэффициентов Фурье для нормы и язвенной болезни желудка и двенадцатиперстной кишки, что позволяет использовать их в качестве информативных признаков для дифференциальной диагностики функционального состояния исследуемого органа желудочно-кишечного тракта. При расчетах число гармоник

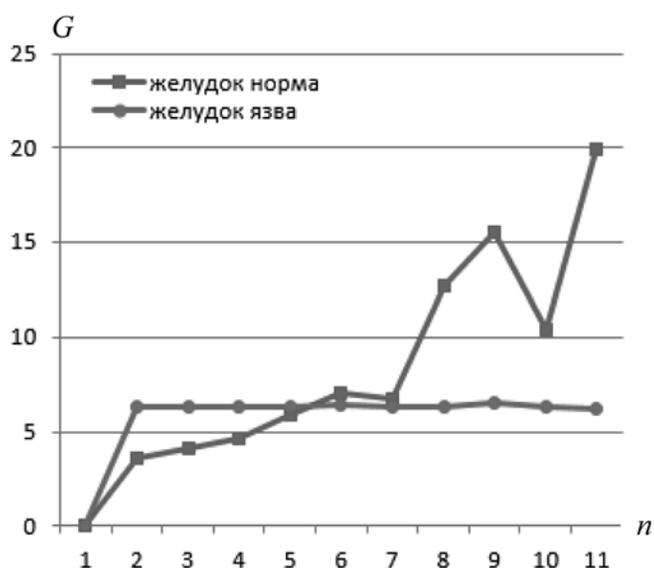
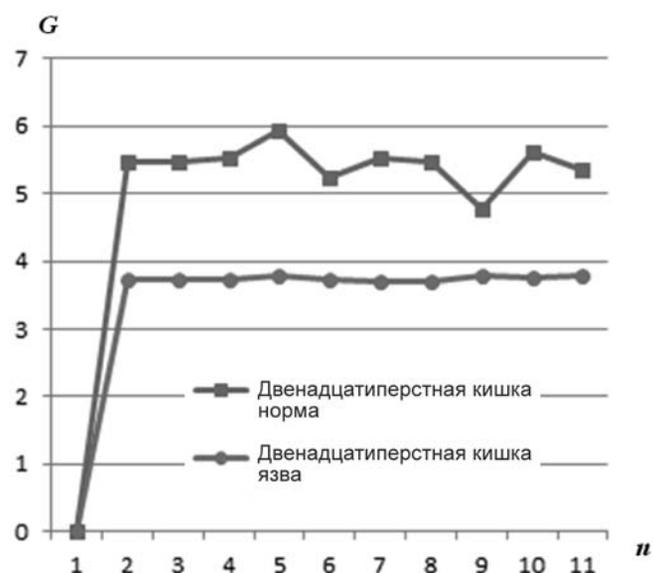
Средние значения комплексных коэффициентов G для желудка

Номер гармоники, n	G_0	G_1	G_2	G_3	G_4	G_5	G_6	G_7	G_8	G_9	G_{10}
Желудок (норма)	0,00614	3,5653	4,14436	4,59584	5,86207	7,00504	6,75691	12,6898	15,5735	10,3688	19,9622
Желудок (язва)	$9,9 \cdot 10^{-8}$	6,33458	6,3261	6,31718	6,35878	6,38944	6,35681	6,34567	6,49474	6,31693	6,23503

Таблица 3

Средние значения комплексных коэффициентов G для двенадцатиперстной кишки

Номер гармоники, n	G_0	G_1	G_2	G_3	G_4	G_5	G_6	G_7	G_8	G_9	G_{10}
Двенадцатиперстная кишка (норма)	$5 \cdot 10^{-0.5}$	5,4689	5,4646	5,52696	5,92111	5,24062	5,52971	5,47688	4,78089	5,60848	5,35389
Двенадцатиперстная кишка (язва)	0,00018	3,71958	3,71888	3,71501	3,76997	3,72728	3,70776	3,6942	3,79403	3,74646	3,77711

Рис. 1. Зависимость средних значений комплексных коэффициентов G от номера гармоникиРис. 2. Зависимость средних значений комплексных коэффициентов G от номера гармоники

исследуемого электрогастроэнтерографического сигнала может варьироваться по желанию исследователя в целях определения характера изменения получаемых значений коэффициентов. Полученные в целях определения характера поведения предлагаемых усредненных диагностически значимых параметров зависимости по всем исследуемым пациентам подтвердили результаты проведенных исследований.

Таким образом, можно утверждать, что возможна дифференциальная диагностика органов желудочно-кишечного тракта по полученным показателям.

Полученные информативные показатели в целях классификации функционального состояния органов желудочно-кишечного тракта могут быть использованы в качестве входных параметров искусственной нейронной сети. Анализ топологий и структур искусственной нейронной сети, функций активации, методов и алгоритмов обучения нейросетей, а также видов нормализации входных параметров позволяет отдать предпочтение такой структуре, как многослойный персептрон, для которой характерно последовательное выделение признаков из исходного образа, что способствует эффективному распознаванию [10]. Сопоставительный анализ основных алгоритмов обучения нейросети позволяет отдать предпочтение алгоритму обратного распространения ошибки "Back Propagation", обладающему способностью минимизировать ошибки и являющемуся наиболее подходящим для нейросетей с технологией обучения с преподавателем в задачах классификации образов. Для рассматриваемой проблемы структура многослойного персептрона приведена на рис. 3. Одним из главных параметров нейросетей является число нейронов в скрытых слоях, которое определяет соотношение точности и обобщающей способности системы [11].

Для классификации патологий в электрофизиологии в работе [12] предложен модульный вариант структуры построения нейросетевых блоков, включающий в себя несколько параллельно располо-

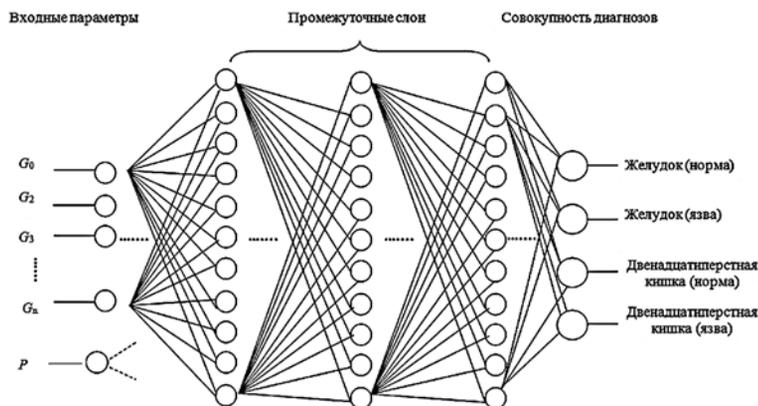


Рис. 3. Структура многослойного персептрона для распознавания патологий желудочно-кишечного тракта

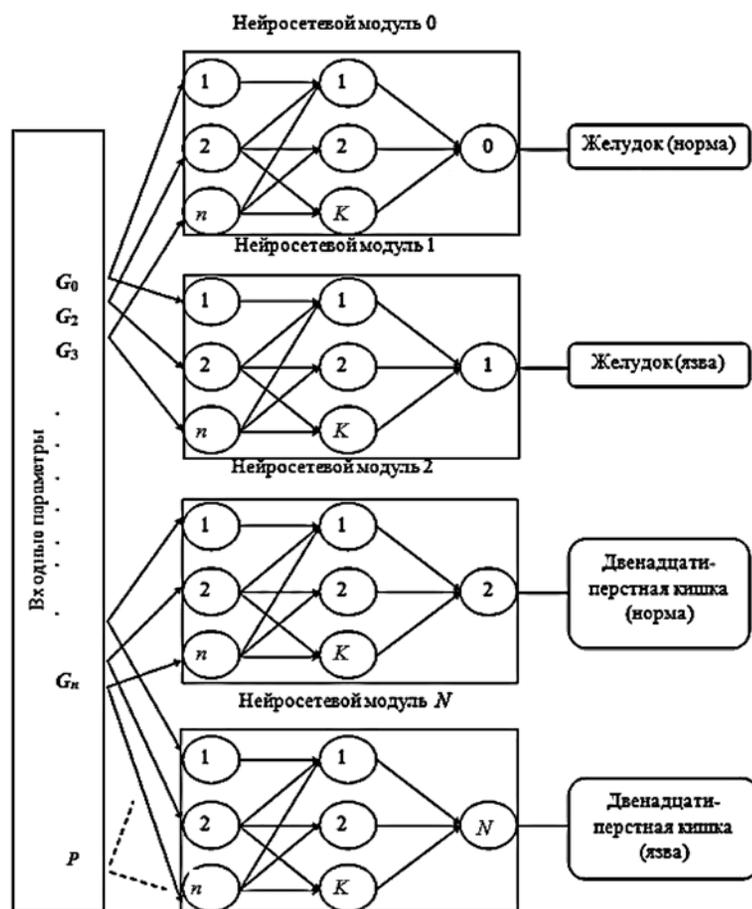


Рис. 4. Модульный вариант построения нейронной сети для распознавания патологий желудочно-кишечного тракта (n — число входных параметров; K — число нейронов промежуточного слоя; N — число анализируемых патологий)

женных нейросетевых модулей, построенных на основе структуры многослойного персептрона. Для классификации заболеваний органов желудочно-кишечного тракта на основании входных параметров, представляющих собой комплексные ко-

эффициенты Фурье G_j , такая структура может иметь вид, представленный на рис. 4.

В качестве дополнительного входного параметра нейросети может быть использовано эффективное значение исследуемого сигнала P_j , что также позволит минимизировать ошибку диагностирования.

Преимуществом данной структуры является концентрация ресурсов каждого модуля на распознавание только одного заболевания, что способствует уменьшению вероятности ошибки неверного заключения для всей системы в целом, а также позволяет расширить функциональные возможности искусственной нейронной сети путем увеличения числа нейросетевых модулей для распознавания новых патологий без переобучения всей системы.

Список литературы

1. Мамедов Н. Я., Абдуллаев Н. Т., Агаева Г. С. Численный алгоритм спектрального анализа измерительных сигналов // Изв. вузов "Приборостроение", 2014. Т. 57, № 7. С. 37–41.
2. Мамедов Н. Я., Абдуллаев Н. Т., Агаева Г. С., Джафарова А. Н. Оценка погрешности метода быстрого спектрального анализа // Изв. вузов "Приборостроение", 2015. Т. 58, № 6. С. 428–435.
3. Дмитриева Н. В. Системная электрофизиология: системный анализ электрофизиологических процессов. М.: Либроком, 2015. 251 с.
4. Ребров В. Г., Станковский Б. А., Кулинина Г. И. Особенности регистрации электрической активности желудка и кишечника с поверхности тела пациента // Российский журнал гастроэнтерологии, гепатологии, колопроктологии. 1995. № 2. С. 48–52.
5. Смирнова Г. О., Силуянов С. В. Периферическая электрогастроэнтерография в клинической практике. Пособие для врачей / Под ред. проф. В. А. Ступина. М.: Издат. дом "Медпрактика-М", 2009. 20 с.
6. Электрогастроэнтерография: исследование электрической активности желудка и кишечника. Функциональная гастроэнтерология. М.: ЗАО НПП "Исток-Система". URL: <http://www.gastroscan.ru/physician/egg/>.
7. Модели и алгоритмы обработки электрогастроэнтерографического сигнала. URL: <http://www.masters.dounta.edu.ua/2009/kita/boutiti/diss/index.html>.
8. Вавринчук С. А., Косенко П. М. Системный анализ показателей периферической электрогастроэнтерографии у больных с осложненной язвенной болезнью. Хабаровск: Центр ИПКСЗ, 2012. 203 с.
9. Абдуллаев Н. Т., Дышин О. А., Керимова М. И. Дифференциальная диагностика заболевания желудочно-кишечного тракта на основе мультифрактального анализа электрогастроэнтерографических сигналов // Информационные технологии. 2015. Т. 21, № 11. С. 871–877.
10. Осовский С. Нейронные сети для обработки информации. М.: Финансы и статистика. 2002. 344 с.
11. Дмитриев Р. Н., Котин В. В. Моделирование временных рядов заболеваемости с использованием искусственных нейронных сетей // Медицинская техника. 2013. № 1. С. 35–38.
12. Сушкова Л. Т., Исаков Р. В., Альмабрук М., Лукьянова Ю. А. Результаты исследования нейронных сетей в задачах распознавания патологических изменений электрической активности сердца // Биомедицинская радиоэлектроника. 2010. № 7. С. 9–13.

N. T. Abdullayev¹, Associate Professor, e-mail: a.namik46@mail.ru,
N. Ya. Mamedov², Associate Professor, e-mail: mr.nuraddin47@mail.ru,
G. S. Agayeva¹, Researcher, e-mail: gunel_aso@yahoo.com

¹ Azerbaijan Technical University, ² Azerbaijan State University of Oil and Industry

Application of the Method the Spectral Analysis for Differential Diagnosis the Disease Bodies of the Digestive Tract

The possibility use of the offered algorithm the spectral analysis of measuring signals for differential diagnostics a functional condition bodies of the enterogastric highway is considered. Elektrogastroenterografic signals for a normal state are investigated and at ulcer damages of a stomach and duodenum. For differential diagnosis this illness average values of complex coefficients Fourier are used, dependences these coefficients on number harmonicas of the studied signal are given. Structures of artificial neural networks for classification samples are offered.

Keywords: digestive tract, elektrogastroenterografic signals, spectral analysis, Fourier's coefficients, neural network

References

1. Mamedov N. Ya., Abdullaev N. T., Agaeva G. S. Chislennyiy algoritm spektralnogo analiza izmeritelnykh signalov, *Izv. vuzov "Priborostroenie"*, 2014, vol. 57, no. 7, pp. 37–41.

2. Mamedov N. Ya., Abdullaev N. T., Agaeva G. S., Dzhafarova A. N. Otsenka pogreshnosti metoda byistrogo spektralnogo analiza, *Izv. vuzov "Priborostroenie"*, 2015, vol. 58, no. 6, pp. 428–435.

3. Dmitrieva N. V. *Sistemnaya elektrofiziologiya: sistemnyiy analiz elektrofiziologicheskikh protsessov*, Moscow, Librokom, 2015, 251 p.

4. Rebrov V. G., Stankovskiy B. A., Kulinina G. I. Osobennosti registratsii elektricheskoy aktivnosti zheludka i kishechnika s poverhnosti tela patsienta, *Rossiyskiy zhurnal gastroenterologii, gepatologii, koloproktologii*, 1995, no. 2, pp. 48–52.

5. Smirnova G. O., Siluyanov S. V. *Perifericheskaya elektrogastroenterografiya v klinicheskoy praktike. Posobie dlya vrachey*. Pod red. prof. V. A. Stupina, Moscow: Izdat. dom "Mediapraktika-M", 2009, 20 p.

6. *Elektrogastroenterografiya: issledovanie elektricheskoy aktivnosti zheludka i kishechnika*. Funktsionalnaya gastroenterologiya, Moscow: ZAO NPP "Istok-Sistema". URL: <http://www.gastroscan.ru/physician/egg/>.

7. *Modeli i algoritmy obrabotki elektrogastroenterograficheskogo signala*. URL: <http://www.masters.dounta.edu.ua/2009/kita/boutiti/diss/index.html>.

8. Vavrinchuk S. A., Kosenko P. M. *Sistemnyiy analiz pokazateley perifericheskoy elektrogastroenterografii u bolnykh s oslozhnennoy yavlennoy boleznyu*. Habarovsk: IPKSH, 2012, 203 p.

9. Abdullaev N. T., Dyishin O. A., Kerimova M. I. Differentsialnaya diagnostika zabollevaniya zheludochno-kishechnogo trakta na osnove multifraktalnogo analiza elektrogastroenterograficheskikh signalov, *Informatsionnyie tehnologii*, 2015, vol. 21, no. 11, pp. 871–877.

10. Ossovskiy S. *Neyronnyie seti dlya obrabotki informatsii*, Moscow, Finansyi i statistika, 2002, 344 p.

11. Dmitriev R. N., Kotin V. V. Modelirovanie vremennykh ryadov zabollevaemosti s ispolzovaniem iskusstvennykh neyronnykh setey, *Meditsinskaya tehnika*, 2013, no. 1, pp. 35–38.

12. Sushkova L. T., Isakov R. V., Almabruk M., Lukyanova Yu. A. Rezultaty issledovaniya neyronnykh setey v zadachah raspoznavaniya patologicheskikh izmeneniy elektricheskoy aktivnosti serdtsa, *Biomeditsinskaya radioelektronika*, 2010, no. 7, pp. 9–13.

VI международный форум



"BROADBAND RUSSIA FORUM 2016 —
Развитие сетей нового поколения в России"



24–25 ноября 2016 г.,

Москва, отель "Холидей Инн Лесная"

Broadband Russia Forum — это крупнейшее событие рынка широкополосного доступа (ШПД) в России

ОСНОВНЫЕ ТЕМЫ ФОРУМА:

- Настоящее и будущее сетей ШПД: перспективы развития рынка Инновации в архитектуре, технологиях и бизнес-моделях развития сетей ШПД
- Информационная безопасность в сетях ШПД
- Сервисы и контент в сетях широкополосного доступа
- Стратегии операторов ШПД в области применения облачных технологий
- Эволюция бизнес-модели операторов связи: от традиционных услуг к IoT-сервисам
- Предпосылки повышения M & A-активности в регионах
- Будущее уже сегодня: SDN и NFV в сетях ШПД
- Умные сети: мониторинг сетевой инфраструктуры и управление трафиком
- Региональные аспекты развития рынка ШПД

Подробную информации о форуме см. на сайте
<http://www.comnews-conferences.ru/ru/conference/bb2016>

ЖУРНАЛ В ЖУРНАЛЕ



**НЕЙРОСЕТЕВЫЕ
ТЕХНОЛОГИИ**

№ 9
СЕНТЯБРЬ
2016

Главный редактор:

ГАЛУШКИН А. И.

Редакционная коллегия:

АВЕДЬЯН Э. Д.
БАЗИЯН Б. Х.
БЕНЕВОЛЕНСКИЙ С. Б.
БОРИСОВ В. В.
ГОРБАЧЕНКО В. И.
ЖДАНОВ А. А.
ЗЕФИРОВ Н. С.
ЗОЗУЛЯ Ю. И.
КРИЖИЖАНОВСКИЙ Б. В.
КУДРЯВЦЕВ В. Б.
КУЛИК С. Д.
КУРАВСКИЙ Л. С.
РЕДЬКО В. Г.
РУДИНСКИЙ А. В.
СИМОРОВ С. Н.
ФЕДУЛОВ А. С.
ЧЕРВЯКОВ Н. И.

**Иностранные
члены редколлегии:**

БОЯНОВ К.
ВЕЛИЧКОВСКИЙ Б. М.
ГРАБАРЧУК В.
РУТКОВСКИЙ Л.

Редакция:

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

Кулагин В. П., Иванов А. И., Серикова Ю. И.

Корректировка методических и случайных составляющих погрешностей вычисления коэффициентов корреляции, возникающих на малых выборках биометрических данных 705

Аведьян Э. Д., Ле Т. Ч. Л.

Двухуровневая система обнаружения DoS-атак и их компонентов на основе нейронных сетей СМАС 711

В. П. Кулагин, д-р техн. наук, проф., руководитель научно-исследовательской лаборатории космических исследований в области технологий, систем и процессов, e-mail: vkulagin@hse.ru, Московский институт электроники и математики НИУ ВШЭ,

А. И. Иванов, д-р техн. наук, доц., нач. лаборатории биометрических и нейросетевых технологий АО "Пензенский научно-исследовательский электротехнический институт", г. Пенза, e-mail: ivan@pniei.penza.ru.,

Ю. И. Серикова, студент, e-mail: julia_ska@yandex.ru
ФБГОУ ВПО "Пензенский государственный университет"

Корректировка методических и случайных составляющих погрешностей вычисления коэффициентов корреляции, возникающих на малых выборках биометрических данных

Показано, что вычисление математических ожиданий, стандартных отклонений и коэффициентов корреляции дает значительные ошибки при использовании малых выборок. Ошибка вычисления коэффициентов корреляции существенно больше, чем ошибки вычисления математических ожиданий и стандартных отклонений. Причиной ошибок является квантование континуумов исходных данных через их представление небольшой выборкой. Приводятся графики распределения вероятностей появления ошибок квантования и их следствия — ошибок, возникающих при вычислении коэффициентов корреляции на малых выборках. Даны таблицы значений параметров для двух величин коэффициентов корреляции при разных размерах выборок биометрических данных. Доказано наличие методических ошибок измерительно-вычислительных процедур определения коэффициентов корреляции. Даны таблицы описания значений методических ошибок, обусловленных малой тестовой выборкой, и корректирующих методическую ошибку гипербола.

Ключевые слова: методическая ошибка, коэффициент корреляции, малая выборка, обработка биометрических данных

Постановка задачи

В настоящее время активно идут процессы информатизации современного общества. Появилась необходимость в создании программируемых автоматов, способных безошибочно узнавать своего хозяина и автоматически обучаться на малом числе примеров биометрического образа человека-хозяина. За рубежом для этой цели используют так называемые "нечеткие экстракторы" [1–4]. В России эти же задачи решаются с использованием искусственных нейронных сетей [5, 6]. Нейросетевые преобразователи биометрии в код доступа почти во всех отношениях оказываются лучше "нечетких экстракторов". Единственный показатель, по которому "нечеткие экстракторы" превосходят нейронные сети, — это объем обучающей выборки. Настраивать "нечеткие экстракторы" удается даже на одном примере распознаваемого биометрического образа, тогда как при обучении нейросетевого преобразователя алгоритмом ГОСТ Р 52633.5—2011 [6] требуется порядка 20 примеров биометрического образа "Свой". При этом стандартный алгоритм обучения [6] устойчив и имеет линейную вычислительную сложность. Низкая вычислительная сложность стандартного алгоритма обучения и его высокая устойчивость обусловлены тем, что он не является итерационным. Алгоритм построен на вычислении весовых коэффициентов нейронной сети путем преобразо-

вания математических ожиданий биометрических параметров и их стандартных отклонений:

$$\begin{cases} E(v) = \frac{1}{n} \sum_{i=1}^n v_i; \\ \sigma(v) = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (E(v) - v_i)^2}, \end{cases} \quad (1)$$

где n — число примеров в обучающей выборке; i — номер примера в обучающей выборке; $E(v)$ — математическое ожидание одного из контролируемых биометрических параметров (v); $\sigma(v)$ — стандартное отклонение контролируемого биометрического параметра.

Предположительно, стандартный алгоритм обучения искусственных нейронных сетей [6] может быть улучшен, если он будет усовершенствован путем использования большего числа статистических параметров. В частности, ожидается, что дополнительно при обучении нейронных сетей можно будет использовать коэффициенты парной корреляции биометрических данных [7], например, будет использоваться коэффициент корреляции между первым и вторым биометрическими параметрами:

$$r(v_1, v_2) = \frac{1}{n} \sum_{i=1}^n \frac{(E(v_1) - v_{1,i})(E(v_2) - v_{2,i})}{\sigma(v_1)\sigma(v_2)}. \quad (2)$$

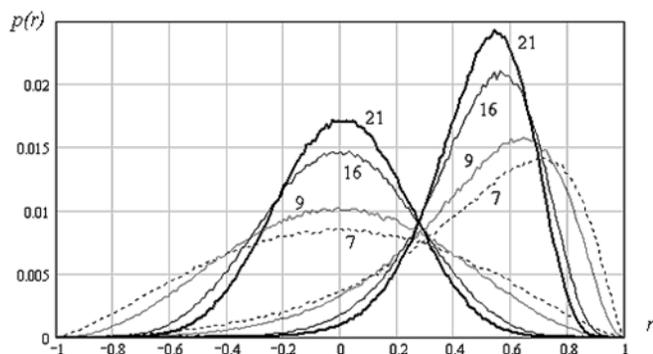


Рис. 1. Распределения значений коэффициентов корреляции для выборок из 7, 9, 16, 21 примеров при двух заданных значениях коэффициентов корреляции $r = 0$ и $r = 0,5$

Дополнительный учет коэффициентов корреляции [7], как дополнительных контролируемых биометрических параметров, позволяет увеличить число входов нейронной сети с величины N до величины $(N + N^2/2)$. Так, при числе контролируемых биометрических параметров $N = 400$ появляется теоретическая возможность перейти от нейронных сетей с 400 входами к нейронным сетям с 80 000 входов.

К сожалению, на малых тестовых выборках, состоящих из $n \approx 21$ примеров, ошибки вычисления коэффициентов корреляции (2) оказываются велики. Причиной больших ошибок является то, что при вычислении коэффициентов парной корреляции используются четыре статистических момента $E(v_1)$, $E(v_2)$, $\sigma(v_1)$, $\sigma(v_2)$. Каждый из этих моментов будет иметь собственную погрешность вычисления $\Delta E(v_1)$, $\Delta E(v_2)$, $\Delta \sigma(v_1)$, $\Delta \sigma(v_2)$. Любые вычисления склонны накапливать ошибки входящих в них величин. На рис. 1 приведены распределения значений коэффициентов корреляции, полученные при разных размерах тестовых выборок.

Из рис. 1 видно, что методические ошибки, обусловленные малым размером тестовой выборки, могут быть весьма значительны. Так, при 21 примере в тестовой выборке и отсутствии корреляции между параметрами вычисления по формуле (2) вместо ожидаемого значения $r = 0$ можно получить значения $r = \pm 0,65$. Чем больше число примеров в выборке, тем точнее результат расчетов, однако увеличивать размер выборки не всегда возможно. Как следствие, актуальной становится задача повышения устойчивости вычислений по формуле (2) либо поиск других более устойчивых формул для оценки парной корреляции для малых выборок биометрических данных.

Шумы квантования статистических функций биометрических данных, возникающие из-за малого числа примеров в тестовой выборке

Причиной ошибок вычислений является то, что биометрические данные по своей природе контину-

альны (непрерывны). Многие из них хорошо описываются нормальным законом распределения значений. Когда мы представляем континуум возможных значений всего 21 примером, возникает ошибка квантования статистических функций. В верхней части рис. 2 приведен график ступенчатой эмпирической функции вероятности биометрического параметра $P(v)$. Ее график монотонно растет с амплитудой каждого скачка — $\Delta P(\Delta v) = \frac{1}{21}$.

В нижней части рис. 2 приведена ошибка квантования или шум квантования непрерывных данных. Очевидно, что влияние шумов квантования может быть снижено путем применения специальных алгоритмов уменьшения (подавления) шумов.

Следует подчеркнуть, что хорошо изученные статистические критерии имеют простую интерпретацию, если их рассматривать как некоторую процедуру обработки шумов квантования. В частности, статистический критерий Колмогорова—Смирнова [8]

$$\sup_{-\infty < v < +\infty} |P(v) - \tilde{P}(v)| = \max\{\max|\Delta P(v)|\} \quad (3)$$

следует рассматривать как оценку сверху амплитуды шума квантования непрерывной функции вероятности $P(v)$ или как глобальный максимум значений модуля ошибок квантования.

С этих позиций статистический критерий Крамера фон Мизеса [8] является оценкой стандартного отклонения шума квантования непрерывной функции вероятности:

$$\begin{aligned} & \int_{-\infty}^{\infty} \{P(v) - \tilde{P}(v)\}^2 dv = \\ & = \int_{-\infty}^{\infty} \{E(\Delta P) - \Delta P(v)\}^2 dv = \sigma^2(\Delta P). \end{aligned} \quad (4)$$

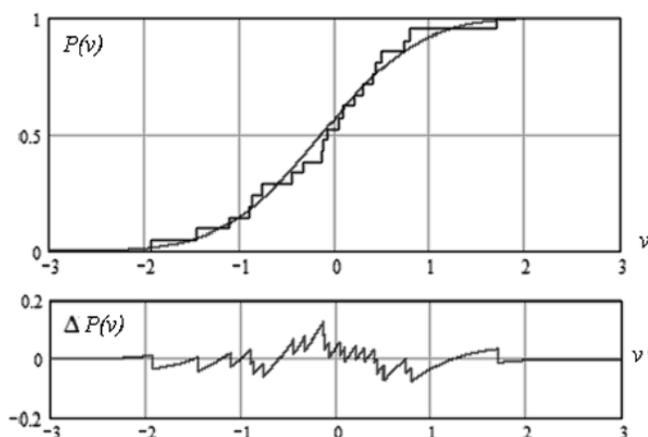


Рис. 2. Эффекты квантования непрерывной функции вероятности при представлении континуума данных 21 примером

По сути дела, каждый из известных критериев проверки статистических гипотез [8] можно рассматривать как некоторый способ подавления шумов квантования. В этом смысле можно сравнивать между собой различные статистические критерии по их способности подавлять шумы квантования. В частности, критерий Крамера фон Мизеса подавляет влияние шумов квантования много эффективнее, чем критерий хи-квадрат Пирсона.

Могут использоваться и другие процедуры снижения влияния шумов квантования, возникающих из-за малого объема тестовой выборки. За рубежом при обработке биометрических данных часто используют так называемый "бут-страп" метод [9], суть которого состоит в искусственном дополнении недостающих данных, например, их размещением в пустых интервалах эмпирической гистограммы. Хороший результат дает также сглаживание эмпирических гистограмм одномерными цифровыми фильтрами [10] и переход к многомерной обработке данных [11].

Можно предположить, что одномерные алгоритмы снижения влияния шумов квантования [9–11] могут быть обобщены до двухмерных алгоритмов более устойчивого вычисления коэффициентов парной корреляции. Таких алгоритмов вычисления коэффициентов корреляции может быть создано достаточно много. Так как каждый такой алгоритм вычисления коэффициентов корреляции будет иметь собственную погрешность, появляется возможность направленного поиска наиболее устойчивых алгоритмов вычислений, способных эффективно противодействовать шумам квантования.

Моделирование зависимых равнокоррелированных данных

Известно, что моделировать случайные многомерные процессы крайне сложно [12]. Технически вполне возможно вычислить симметричную матрицу размерностью 256×256 , описывающую корреляционные связи между выходами нейросетевого

преобразователя (между 256 входами нейронной сети, наблюдающей 256 биометрических параметров). Однако построить генератор случайных чисел, точно воспроизводящий корреляционные связи столь высокой размерности, технически невозможно.

Формально можно использовать 256 генераторов независимых случайных данных ξ , умножив их на некоторую связывающую матрицу A . Однако найти подобную матрицу, которая даст нужные корреляционные связи $r(y_k, y_j)$, достаточно трудно. Это обратная задача, которая относится к плохо обусловленным.

Для решения поставленной задачи воспользуемся симметричной связывающей матрицей, которая имеет единичную диагональ и одинаковые элементы вне диагонали:

$$\begin{bmatrix} 1 & a & \dots & a \\ a & 1 & \dots & a \\ \dots & \dots & \dots & \dots \\ a & a & \dots & 1 \end{bmatrix} \times \begin{bmatrix} \xi_{1,i} \\ \xi_{2,i} \\ \dots \\ \xi_{m,i} \end{bmatrix} = \begin{bmatrix} y_{1,i} \\ y_{2,i} \\ \dots \\ y_{m,i} \end{bmatrix} \Rightarrow$$

$$\Rightarrow R_m = \begin{bmatrix} 1 & r & \dots & r \\ r & 1 & \dots & r \\ \dots & \dots & \dots & \dots \\ r & r & \dots & 1 \end{bmatrix}. \quad (5)$$

В этом случае данные оказываются равнокоррелированными. Если плавно изменять параметр a , связывающей матрицы A от 0 до 1, то коэффициенты равной коррелированности r также будут плавно меняться в пределах от 0 до 1.

Оценка методической ошибки

Используя формулу (2), проведем вычисления, меняя значения коэффициента корреляции r при разных значениях размера выборки n . Усредненные результаты одного миллиона вычислений приведены в табл. 1.

Таблица 1

Значения математических ожиданий $E(r)$ при разных размерах тестовой выборки n

n	Число степеней свободы (размер тестовой выборки)											
	7	8	9	10	11	12	14	16	21	32	64	
Равная коррелированность данных (r)	0,01	0,01	0,01	0,01	0,01	0,01	0,01	0,01	0,01	0,01	0,01	0,01
	0,1	0,092	0,092	0,093	0,094	0,095	0,095	0,096	0,096	0,097	0,098	0,099
	0,2	0,185	0,186	0,189	0,189	0,189	0,191	0,193	0,194	0,195	0,197	0,198
	0,3	0,278	0,281	0,284	0,286	0,287	0,288	0,290	0,292	0,294	0,296	0,298
	0,4	0,373	0,377	0,379	0,382	0,384	0,386	0,388	0,390	0,393	0,396	0,398
	0,5	0,467	0,473	0,476	0,479	0,481	0,483	0,485	0,487	0,491	0,494	0,497
	0,6	0,567	0,572	0,575	0,578	0,581	0,582	0,585	0,588	0,591	0,594	0,598
	0,7	0,667	0,672	0,676	0,679	0,681	0,683	0,686	0,688	0,691	0,695	0,698
	0,8	0,771	0,775	0,778	0,781	0,783	0,785	0,787	0,789	0,792	0,795	0,797
	0,9	0,883	0,886	0,888	0,890	0,891	0,892	0,894	0,895	0,897	0,898	0,899
0,99	0,988	0,988	0,989	0,989	0,989	0,989	0,989	0,99	0,99	0,99	0,99	

Из табл. 1 видно, что заданные при моделировании значения коэффициента корреляции (левый столбец таблицы) существенно отличаются от математических ожиданий результатов моделирования $E(r)$. Данная методическая ошибка Δr уменьшается с увеличением размера тестовой выборки (разность соответствующих значений левого и правого столбцов табл. 1 стремится к нулю).

Построчная корректировка методических ошибок

Рассмотрим метод коррекции методической ошибки. Если рассматривать строки табл. 1, то можно заметить, что значения элементов строки удобно аппроксимировать гиперболой степени 1,5:

$$\Delta r(n) = \left(\alpha + \frac{\lambda}{n^{1,5}} \right), \quad (6)$$

где n — число опытов в знаменателе гиперболы; α и λ — подбираемые при аппроксимации коэффициенты.

Мультипликативная корректировка методической ошибки может быть выполнена на основе выражения

$$\tilde{r}(n) = r(n) \left(1 + \alpha + \frac{\lambda}{n^{1,5}} \right), \quad (7)$$

где $r(n)$ — значение коэффициента корреляции, вычисленное по формуле (2), $\tilde{r}(n)$ — значение коэффициента корреляции с учетом компенсации методической ошибки. Результаты аппроксимации данных гиперболой приведены в табл. 2.

Примеры реальных графиков методических ошибок (сплошные линии) и их приближений (пунктирные линии), полученные с использованием данных табл. 2, приведены на рис. 3.

В итоге можно отметить, что корректировка методической погрешности вычислений возможна как при использовании соответствующих корректирующих таблиц, так и при использовании аналитической аппроксимации в виде гипербол (6), (7).

Корректировка методической ошибки при фиксированных объемах тестовой выборки

Рассмотрим другой способ корректировки методической ошибки, основанный на аппроксимации значений столбцов табл. 1. На рис. 4 приведены соответствующие графики.

Приближение данных численного моделирования (пунктирные линии рис. 4) осуществлено с использованием взвешенной функции бета-распределения значений [8]:

$$\Delta r = \text{sign}(r) \frac{\lambda(\alpha + \beta + 1)!}{\alpha! \beta!} |r|^\alpha (1 - |r|)^\beta, \quad (8)$$

где λ, α, β — подбираемые при аппроксимации параметры, $\text{sign}(r)$ — знак модуля учитываемого значения коэффициента парной корреляции.

Данные о параметрах аппроксимации методической ошибки вида (8) для разных размеров выборки n приведены в табл. 3.

Из рис. 4 видно, что корректировка с использованием аппроксимации (8) позволяет уменьшить

Таблица 2

Данные аппроксимации методической ошибки гиперболой при объемах выборки от 7 до 64 примеров

$\tilde{r}(n)$	0,01	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	0,99
α	0,00	0,0008	0,0013	0,0008	0,001	0,002	0,002	0,002	0,002	0,0003	0,00
λ	0,00	0,18	0,29	0,45	0,5	0,6	0,6	0,6	0,52	0,3	0,035

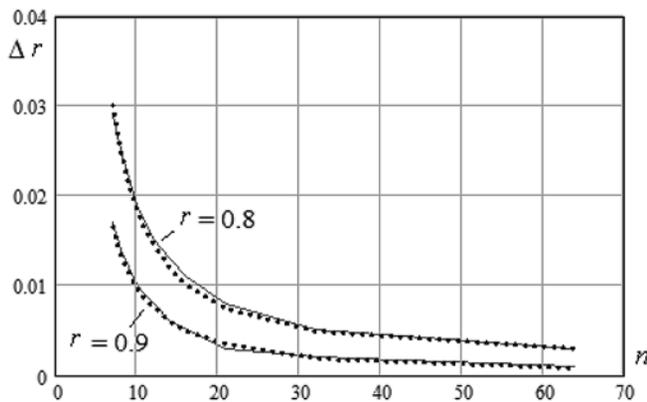


Рис. 3. Примеры графиков методических ошибок (аппроксимация дана пунктиром) как функции размеров тестовой выборки

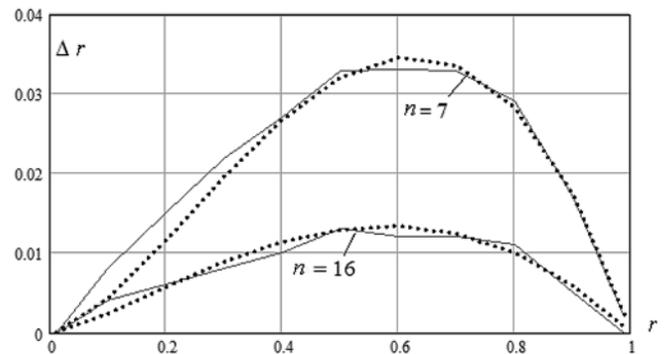


Рис. 4. Примеры графиков методических ошибок (аппроксимация дана пунктиром) как функции значений коэффициентов корреляции

Данные аппроксимации методической ошибки взвешенным бета-распределением для разных объемов выборки n

n	7	8	9	10	11	12	14	16	21	32	64
λ	0,021	0,018	0,0166	0,0147	0,013	0,0114	0,010	0,0084	0,0061	0,0043	0,0021
α	2,45	2,45	2,45	2,45	2,45	2,45	2,45	2,45	2,45	2,45	2,45
β	1,86	1,88	1,90	1,93	1,96	1,98	1,99	2,11	2,04	2,06	2,08

методическую ошибку от 5 до 10 раз, если далее применять аддитивную коррекцию:

$$\tilde{r}(n) = r(n) + \text{sign}(r) \frac{\lambda(\alpha + \beta + 1)!}{\alpha! \beta!} |r(n)|^\alpha (1 - |r(n)|)^\beta. \quad (9)$$

Объединение корректора методической погрешности по строкам табл. 1 в виде соотношения (7) с корректором методической ошибки по столбцам табл. 1 в виде выражения (9) позволяет получить двухмерный цифровой корректор методических погрешностей определения коэффициентов корреляции, вычисляемых на малых тестовых выборках объемом от 7 до 64 примеров. На данный момент сравнение цифровых корректоров, полученных сплайн-аппроксимацией исходной табл. 1 с аналитическими приближениями (7) и (9), не проведено. Необходимы дополнительные исследования.

Заключение

Чем меньше выборка примеров, представляющих биометрический образ, тем больше амплитуда шумов квантования (см. рис. 2). Кажется, что это тупик и обучать преобразователи биометрия—код на малых выборках невозможно. На самом деле это не так. Люди и животные способны обучаться на очень малом числе примеров нового образа. Это связано с тем, что их естественные нейронные сети обладают очень большой входной размерностью и способны учитывать до 10 000 параметров. Чем выше размерность решаемой задачи, тем эффективнее оказываются подавление случайных составляющих погрешностей наблюдаемых значений параметров нейросетевыми преобразователями биометрия—код [5–7]. В данной статье показано, что учет множества парных коэффициентов корреляции теоретически позволяет поднять размерность решаемой нейронной сетью задачи с 400 до 80 000 контролируемых параметров. Такой 20-кратный рост размерности решаемой задачи должен в конечном итоге обеспечить значительное повышение подавления случайных составляющих погрешностей исходных данных.

Приведенные в данной статье примеры показывают, что методическая составляющая погрешности вычисления коэффициентов корреляции может быть снижена от 5 до 10 раз. После корректировки методических составляющих погрешности доминирующей оказывается случайная погрешность,

которая подавляется сглаживанием данных малой выборки по каждому контролируемому параметру [10] или по всей совокупности учитываемых параметров за счет применения рекордно устойчивого алгоритма обучения искусственных нейронных сетей [6], обладающего к тому же линейной вычислительной сложностью. Применять иные алгоритмы обучения нельзя, так как они имеют более высокую вычислительную сложность и неустойчивы. На данный момент только алгоритм обучения ГОСТ Р 52633.5—2011 [6] способен обучать большие нейронные сети, имеющие от 400 до 80 000 входов на малых выборках (от 12 до 21 примера) за приемлемое время.

Список литературы

1. **Dodis Y., Reyzin L., Smith A.** Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy // EUROCRYPT, April 13, 2004. P. 523–540.
2. **Monrose F., Reiter M., Li Q., Wetzel S.** Cryptographic key generation from voice // Proc. IEEE Symp. on Security and Privacy, pp. 202–213, 2001.
3. **Ramírez-Ruiz J., Pfeiffer C., Nolzaco-Flores J.** Cryptographic Keys Generation Using FingerCodes // Advances in Artificial Intelligence — IBERAMIA-SBIA 2006 (LNCS 4140), p. 178–187, 2006.
4. **Hao F., Anderson R., Daugman J.** Crypto with Biometrics Effectively // IEEE TRANSACTIONS ON COMPUTERS. 2006. Vol. 55, N. 9. P. 1073–1074.
5. **Волчихин В. И., Иванов А. И., Фунтиков В. А.** Быстрые алгоритмы обучения нейросетевых механизмов биометрико-криптографической защиты информации. Пенза: Изд-во Пензенского государственного университета, 2005. 273 с.
6. **ГОСТ Р 52633.5—2011.** Защита информации. Техника защиты информации. Автоматическое обучение нейросетевых преобразователей биометрия — код доступа.
7. **Иванов А. И., Ложников П. С., Качайкин Е. И.** Идентификация подлинности рукописных автографов сетями Байеса—Хэмминга и сетями квадратичных форм // Вопросы защиты информации, 2015. № 2. С. 28–34.
8. **Кобзарь А. И.** Прикладная математическая статистика. Для инженеров и научных работников. М.: ФИЗМАТЛИТ, 2006. 816 с.
9. **Болл Руд, Коннел Джонатан Х., Панканти Шарат, Ратха Налини К., Сеньор Эндрю У.** Руководство по биометрии / Пер. с англ. М.: Техносфера, 2007. 368 с.
10. **Серикова Н. И., Иванов А. И., Качалин С. В.** Биометрическая статистика: сглаживание гистограмм, построенных на малой обучающей выборке // Вестник СибГАУ. 2014. № 3 (55). С. 146–150.
11. **Волчихин В. И., Иванов А. И., Серикова Н. И., Фунтикова Ю. В.** Эффект снижения размера тестовой выборки за счет перехода к многомерному статистическому анализу биометрических данных // Изв. высших учебных заведений. Поволжский регион. Технические науки. Пенза: ПГУ. 2015, № 2. С. 86–91.
12. **Шальгин А. С., Палагин Ю. И.** Прикладные методы статистического моделирования. Л.: Машиностроение, 1986. 320 с.

V. P. Kulagin, Professor, Head of Research Laboratory, e-mail vkulagi@hse.ru,
Moscow Institute of Electronics and Mathematics, Higher School of Economics, Russia, Moscow
A. I. Ivanov, Associate Professor, Head of Laboratory, JSC "Penza Research Electrotechnical Institute"
Russia, Penza, (841-2) 59-33-10, e-mail ivan@pniei.penza.ru.
Yu. I. Serikova, Undergraduate 1st year, FBGOU VPO "Penza State University", Russia,
Penza, e-mail: gosh64@mail.ru

Correction of Methodical and Casual Components of Errors of Calculation of the Coefficients of Correlation Arising on Small Selections of Biometric Data

The paper makes it clear that calculating expectation values, standard deviations and correlation rates gives appreciable errors when using small samples. The correlation rate miscalculation far exceeds those of expectation values and standard deviations. Such errors happen all along of the given data continua quantization through their representation via small sample. We give the probability distribution plots of quantization errors and errors arising from calculating correlation rates in small samples. The allowance for association of correlation rate miscalculations with the test sample sizes makes it possible to apply for simulation modelling of several variables conditioned upon their equal mutual correlation. The table lists the expectation values for correlation rates derived from different sizes of test sample. These values indicate the presence of significant systematic errors arising while evaluating the correlation rates. Small samples show appreciable systematic error, yet decreasing rapidly with the increase in a test sample size. The paper stands for correcting systematic error in the additive and / or multiplicative form, as well as considers two approaches to correction data analytic description by adjusting the detected systematic error. The first approach to analytic description provides for approximating correction data rows by hyperbolic curves with dim fractional exponent. The second approach to analytic description provides for approximating correction data columns by applying to the analytical function describing beta distribution.

Keywords: methodological error, coefficient of correlation, small sample, processing of biometric data

References

1. **Dodis Y., Reyzin L., Smith A.** Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy, Data April 13, *EUROCRYPT*, 2004, pp. 523–540.
2. **Monrose F., Reiter M., Li Q., Wetzel S.** Cryptographic key generation from voice, *Proc. IEEE Symp. on Security and Privacy*, 2001.
3. **Ramirez-Ruiz J., Pfeiffer C., Nolzco-Flores J.** Cryptographic Keys Generation Using FingerCodes, *Advances in Artificial Intelligence — IBERAMIA-SBIA 2006* (LNCS 4140), 2006, pp. 178–187.
4. **Feng Hao, Ross Anderson, John Daugman.** Crypto with Biometrics Effectively, *IEEE TRANSACTIONS ON COMPUTERS*, 2006, vol. 55, no. 9.
5. **Volchihin V. I., Ivanov A. I., Funtikov V. A.** *Bystrye algoritmy obucheniia nei' rosetevy'kh mehanizmov biometriko-kriptograficheskoi' zashchity' informacii.* Penza—2005 g. Izdatel'stvo Penzenskogo gosudarstvennogo universiteta, 2005, pp. 273.
6. **GOST R 52633.5—2011** *Zashchita informacii. Tekhnika zashchity' informacii. Avtomaticheskoe obuchenie nei' rosetevy'kh preobrazovatelei' biometrii — kod dostupa.*
7. **Ivanov A. I., Lozhnikov P. S., Kachai'kin E. I.** Identifikatsiia podlinnosti rukopisny'kh avtografov setiami Bai'esa-KHe'mminga i setiami kvadrachny'kh form, *Voprosy' zashchity' informacii*, 2015, no. 2, pp. 28–34.
8. **Kobzar' A. I.** *Prikladnaia matematicheskaia statistika. Dlia inzhenerov i nauchny'kh rabotneykov.* Moscow: FIZMATLIT, 2006, 816 p.
9. **Boll Rud, Konnel Jonathan Kh., Pankanti Sharat, Ratha Nalini K., Sen'or Andrew U.** *Rukovodstvo po biometrii.* Moscow: Tekhnosfera, 2007, pp. 368.
10. **Serikova N. I., Ivanov A. I., Kachalin S. V.** Biometricheskaia statistika: sglazhivanie gistogramm, postroenny'kh na maloi' obuchaiushchei' vy'borke, *Vestnyk SibGAU*, 2014, no. 3 (55), pp. 146–150.
11. **Volchihin V. I., Ivanov A. I., Serikova N. I., Funtikova Iu. V.** E'ffekt snizheniia razmera testovoi' vy'borki za schet perehoda k mnogomernomu statisticheskomu analizu biometricheskikh danny'kh, *Izvestiia vy'sshikh uchebny'kh zavedenii'. Povolzhskii' region. Tekhnicheskie nauki.* Penza: PGU, 2015, no. 2, pp. 86–91.
12. **Shalygin A. S., Palagin Iu. I.** *Prikladnye metody' statisticheskogo modelirovaniia.* Leningrad: Mashinostroenie, 1986 g. 320 p.

Э. Д. Аведьян^{1, 2, 3}, д-р техн. наук, гл.-науч. сотр.¹, зам нач.², профессор³, e-mail: avedian@mail.ru,
Т. Ч. Л. Ле³, аспирант, e-mail: tranglinh2011@gmail.com

¹Центр информационных технологий и систем органов исполнительной власти, Москва

²Международный центр по информатике и электронике, Москва

³Московский физико-технический институт, Москва

Двухуровневая система обнаружения DoS-атак и их компонентов на основе нейронных сетей СМАС

Приведены результаты применения системы нейронных сетей СМАС (НС СМАС) для обнаружения DoS-атак и их компонентов, выполненные на всех записях базы данных атак KDD Cup 99. Система состоит из двух уровней. Верхний уровень предназначен для обнаружения DoS-атак с помощью обученной НС СМАС, нижний уровень — для выделения из обнаруженных DoS-атак шести их компонентов: Back, Neptune, Land, Pod, Teardrop и Smurf с помощью шести обученных НС СМАС. Ошибка пропуска DoS-атаки и ложной тревоги не превышает 0,2 %.

Ключевые слова: обнаружение DoS-атак, нейронная сеть СМАС, база данных атак KDD Cup 99, компоненты DoS-атак

Введение

В работах Дж. Кеннеди [1] приведены результаты, которые показывают, что в качестве классификатора DoS-атак на информационные ресурсы может быть успешно использована нейронная сеть СМАС (НС СМАС). В этой же работе подчеркивается, что НС СМАС послужит аналитическим компонентом создаваемой полномасштабной интегрированной системы обнаружения атак. К сожалению, результаты создания подобной системы не опубликованы. Следует также отметить, что в работе [1] отсутствует важная информация об используемых автором признаках атак, параметрах НС СМАС, обучающей и тестовой последовательностях, которая позволила бы воспроизвести описанные результаты моделирования. В работе [2] была предпринята попытка подтвердить или опровергнуть утверждения Дж. Кеннеди [1] о возможности успешного применения НС СМАС в качестве классификатора DoS-атак, для чего была разработана и описана технология обнаружения DoS-атак на основе НС СМАС. С помощью этой технологии проведено обучение и тестирование нейронной сети по обнаружению DoS-атак на записях базы данных атак KDD Cup 99. Результаты тестирования системы продемонстрировали высокую вероятность обнаружения DoS-атак и подтвердили утверждение Дж. Кеннеди [1] о том, что НС СМАС может быть успешным классификатором DoS-атак.

В данной работе приведены новые результаты по применению системы нейронных сетей СМАС для обнаружения как DoS-атак, так и их компонентов, выполненные на всех данных базы атак KDD Cup 99 [3]. Система обнаружения состоит из двух уровней. Верхний уровень предназначен для обнаружения DoS-атак с помощью обученной НС СМАС так, как это реализовано в работе [2] с ис-

пользованием уточненного набора из пяти признаков трафика. Нижний уровень предназначен для выделения из обнаруженных DoS-атак шести их компонентов: Back, Neptune, Land, Pod, Teardrop и Smurf [4]. Выделение указанных компонентов выполняется с помощью шести обученных НС СМАС, в каждой из которых используется свой набор, состоящий из двух признаков трафика. Высокая точность обнаружения как DoS-атак, так и их компонентов достигнута в результате оптимизации параметров нейронных сетей СМАС, входящих в систему обнаружения.

Цель настоящей статьи — описание важнейших моментов, которые следует учитывать при использовании НС СМАС при решении задачи обнаружения атак, и иллюстрация применения НС СМАС на примере обнаружения DoS-атак и их компонентов.

1. Постановка задачи и состав базы данных атак KDD Cup 99

Задача, которая рассматривается в настоящей работе, заключается в создании обучающейся системы обнаружения DoS-атак и их компонентов, основными аналитическими элементами которой являются нейронные сети СМАС.

Как отмечалось выше, обучение и тестирование системы выполнено на всем наборе записей о сетевых соединениях, которые имеются в общедоступной базе данных атак KDD Cup 99 [3], общее число которых равно 4 898 431. Поскольку, однако, данную базу данных характеризует избыточный характер [5], то в ней удалены повторы, в результате чего объем данных для обучения и тестирования в сжатой базе данных составил 1 074 992 записей, из которых 247 267 записей являются DoS-атаками. Каждая запись состоит из 41 признака сетевого трафика трех типов: символьные, логические и

Состав DoS-атак в сжатой БД KDD Cup 99

№	Компоненты DoS-атак	Число записей компонентов
1	Back	968
2	Neptune	242 149
3	Teardrop	918
4	Land	19
5	Pod	206
6	Smurf	3007
	Итого	247 267

числовые. Последний 42-й элемент записи содержит информацию о том, к какому одному из пяти классов относится соединение (нормальное соединение и четыре вида атак):

1. *Denial of Service* (DoS) — отказ в обслуживании, при котором происходит генерация большого объема трафика, в результате чего происходит перегрузка сервера и пользователи не могут получить доступ к предоставляемым вычислительной системой ресурсам.

2. *Remote to User* (R2L) — доступ незарегистрированного пользователя к компьютеру со стороны удаленной машины.

3. *User to Root* (U2R) — получение зарегистрированным пользователем привилегий администратора.

4. *Probing* — сканирование портов в целях получения конфиденциальной информации.

Отметим, что при этом каждый из перечисленных видов атак включает в себя несколько компонентов. Для DoS-атак, которые и являются в настоящей работе объектами обнаружения, такими компонентами в базе данных KDD Cup 99 являются атаки [4] Back, Neptune, Teardrop, Land, Pod и Smurf.

Back-атака. Нападающий отправляет запросы на Apache веб-сервер с URL, перед которым стоит большое число слешей. Сервер, пытаясь обработать эти запросы, замедляется и оказывается неспособным обработать другие запросы.

Neptune-атака (SYN Flood). Атакующий посылает на сервер жертвы запросы с открытым флагом SYN, игнорируя и не отвечая на ответные пакеты (режим полуконечного соединения). В результате на сервере происходит переполнение очереди на подключение, и он оказывается неспособным отвечать на запросы пользователей.

Teardrop-атака. Атака отказа в обслуживании, которая эксплуатирует недостаток в реализации старых стеков TCP/IP. Некоторые реализации кода повторной сборки фрагментированных IP-пакетов на этих платформах не обрабатывают должным образом пересекающиеся IP-фрагменты, в результате чего возникает необходимость в перезагрузке сервера.

Land-атака. Атакующий посылает сфальсифицированный пакет SYN, в котором адрес источника совпадает с адресом получателя. В результате сервер оказывается полностью заблокированным, и для восстановления требуется физически его выключить и вновь включить.

Pod-атака (Ping of death). Компьютер-жертва получает от атакующего подделанный эхо-запрос (*ping*), размер которого превышает допустимый. В результате компьютер-жертва вообще перестает отвечать на запросы. Современные операционные системы защищены от этого вида атак.

Smurf-атака. Атакующий посылает пакеты эхо-запроса ICMP по широковещательному адресу посредника, в котором адрес отправителя заменен на адрес жертвы. Большое число компьютеров по-

средника в ответ посылают пакеты по адресу жертвы, что приводит к выходу из строя сервера жертвы.

Число неповторяющихся компонентов DoS-атак в базе данных KDD Cup 99 приведено в табл. 1.

2. Структура системы обнаружения DoS-атак и их компонентов

Система обнаружения DoS-атак и их компонентов имеет двухуровневую структуру, верхний уровень которой предназначен для обнаружения DoS-атак, нижний уровень — для отнесения обнаруженной DoS-атаки к тому или иному компоненту. Элементами обнаружения являются обученные нейронные сети СМАС. Структура системы обнаружения представлена на рис. 1.

Система функционирует следующим образом. Из базы данных KDD Cup 99 извлекается запись о сетевом соединении, содержащая значения 41 признака. Из записи выделяют пять признаков с номерами {3, 4, 5, 23, 30}, названия которых приведены в табл. 2.

Значения этих признаков квантуются, принимая целочисленные значения, и в таком виде поступают на вход обученной распознаванию DoS-атак НС СМАС. По реакции нейронной сети принимается решение о наличии или отсутствии DoS-атаки. Если система принимает решение о наличии DoS-атаки, то данная запись поступает на вход второго уровня системы, в котором находятся обученные распознаванию компонентов DoS-атак шесть НС СМАС. Из записи выделяются значения признаков компонентов DoS-атак с номерами:

Таблица 2

Названия признаков для обнаружения DoS-атак

Номер признака	Название признака	Описание признака
3	Service	Служба
4	Flag	Флаг состояния соединения
5	Source byte	Число байтов, переданных от источника к месту назначения
23	Count	Число соединений на одном хосте текущего соединения в течение последних 2 с
30	Diff srv rate	% подключений к различным услугам

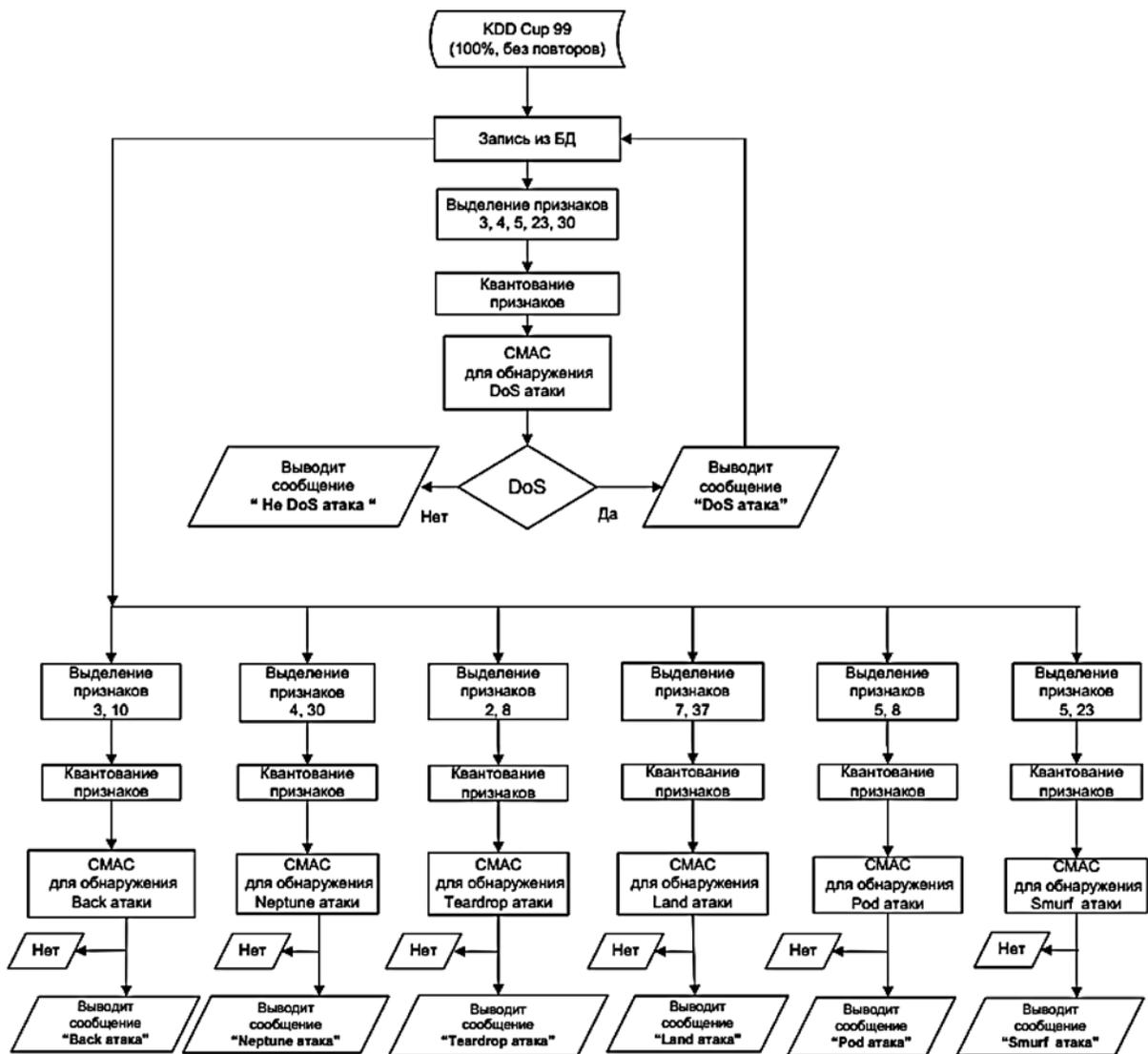


Рис. 1. Двухуровневая структура системы обнаружения DoS-атак и их компонентов

Таблица 3
Названия признаков для обнаружения компонентов DoS-атак

Номер признака	Название признака	Описание признака
2	Protocoltype	Протокол связи
3	Service	Служба
4	Flag	Флаг состояния соединения
5	Source byte	Число байтов, переданных от источника к месту назначения
7	Land	1, если соединение с/на тот же хост/порт; 0, в противном случае
8	Wrong fragment	Число неправильных фрагментов
10	Hot	Число "hot" показателей
23	Count	Число соединений на одном хосте текущего соединения в течение последних 2 с
30	Diff srv rate	% подключений к различным услугам
37	Dst host srv diff host rate	% подключений к одной и той же услуге хоста назначения, поступающих от разных хостов

Back — {3,10}, Neptune — {4,30}, Teardrop — {2,8}, Land — {7,37}, Pod — {5,8} и Smurf — {5,23}, признаки квантуются и поступают на входы обученных выделению компонентов DoS-атак соответствующих НС СМАС. Названия и описания признаков, используемых во втором слое системы, приведены в табл. 3.

3. Нейронная сеть СМАС: структура, алгоритмы обучения

Основным аналитическим элементом описываемой системы обнаружения атак является нейронная сеть СМАС, предложенная в работах Дж. Альбуса [6, 7]. Подробное описание нейронной сети СМАС на русском языке можно найти в работах [8, 9]. Далее приводится краткая информация об этом виде нейронных сетей, которая необходима для понимания особенностей применения НС СМАС в задаче обнаружения атак, см. также [2].

Наиболее существенным отличием НС СМАС от других нейронных сетей является следующее:

- аргументы запоминаемой и воспроизводимой функции (входной N -мерный вектор нейронной сети, или вектор признаков) принимают только дискретные значения:

$$X = \{x^{(1)} = \overline{1, x_{\max}^{(1)}}; x^{(2)} = \overline{1, x_{\max}^{(2)}}, \dots; x^{(N)} = \overline{1, x_{\max}^{(N)}}\}; \quad (1)$$

- нелинейное преобразование аргументов функции происходит неявно с помощью алгоритма вычисления адресов ячеек ассоциативной памяти [8], в которых хранятся числа, определяющие значение запоминаемой функции.

В НС СМАС каждый входной N -мерный вектор x (вектор признаков) делает активными ровно ρ ячеек памяти, суммарное содержимое которых равно значению запоминаемой функции. Каждому входному N -мерному вектору x однозначно соответствует ρ -мерный вектор активных номеров ячеек памяти m . Параметр ρ (обобщающий параметр) играет очень важную роль, его значение определяет разрешающую способность НС СМАС и требуемый объем памяти нейронной сети.

Структура НС СМАС показана на рис. 2.

Важными характеристиками НС СМАС являются объем памяти

$$M = \rho^{-N} + 1 \prod_{i=1}^N (x_{\max}^{(i)} + \rho - 1) \quad (2)$$

и значение обобщающего параметра ρ , которое характеризует аппроксимационные свойства: с уве-

личением ρ увеличиваются аппроксимационные свойства сети, но при этом теряется детализация запоминаемой функции. Поэтому для параметра ρ имеется оптимальное значение, при котором запоминаемая функция воспроизводится наилучшим образом. Для полного использования памяти НС СМАС и нахождения оптимального значения параметра ρ далее полагается

$$x_{\max}^{(i)} = 2^{l^{(i)}} + 1, i = \overline{1, N}, \rho = 2^k, \quad (3)$$

где $l^{(i)}$ и k — целые.

Если компоненты входного вектора запоминаемой функции непрерывны, то их следует преобразовать в цифровые значения. На примере одной переменной z это преобразование выполняется следующим образом. Задаются минимальное и максимальное значения z_{\min} , z_{\max} переменной z и число уровней ее квантования x_{\max} . Вычисляется шаг квантования $\Delta = (z_{\max} - z_{\min})/x_{\max}$, и каждому элементу квантования каждой компоненты присваиваются целочисленные номера $x^{(i)} = 1, 2, \dots, x_{\max}$ по формуле

$$x^{(i)} = \text{Round}((z - z_{\min})/\Delta + 0,5), \quad (4)$$

где Round — функция округления до ближайшего целого.

Структура НС СМАС, представленная на рис. 2, предназначена для запоминания произвольных функций многих переменных. Для применения НС СМАС в задачах классификации эта структура дополняется пороговым элементом, выходом которого служат номера классов входной информации. В рассматриваемой задаче верхнего уровня число классов равно двум: класс DoS-атак и класс не DoS-атак. В этом случае выход v порогового элемента имеет вид:

$$v = 1, \text{ если } (y_{\text{out}} - \Delta) \geq 0 \\ \text{и } v = 0, \text{ если } (y_{\text{out}} - \Delta) < 0, \quad (5)$$

при этом значение $v = 1$ соответствует DoS-атаке, $v = 0$ — ее отсутствию. В (5) y_{out} — выход обученной НС СМАС; Δ — порог, значение которого выбирается экспериментально.

4. Обучающие и тестирующие последовательности системы обнаружения атак

Обучение системы обнаружения атак заключается в обучении 7 НС СМАС по информации из базы данных атак KDD Cup 99.

Для обучения НС СМАС верхнего уровня, предназначенной для обнаружения DoS-атак, из имеющихся 247 267 записей DoS-атак случайным образом выделяется 75 % записей (185 448 записей), состав которых представлен в табл. 4.

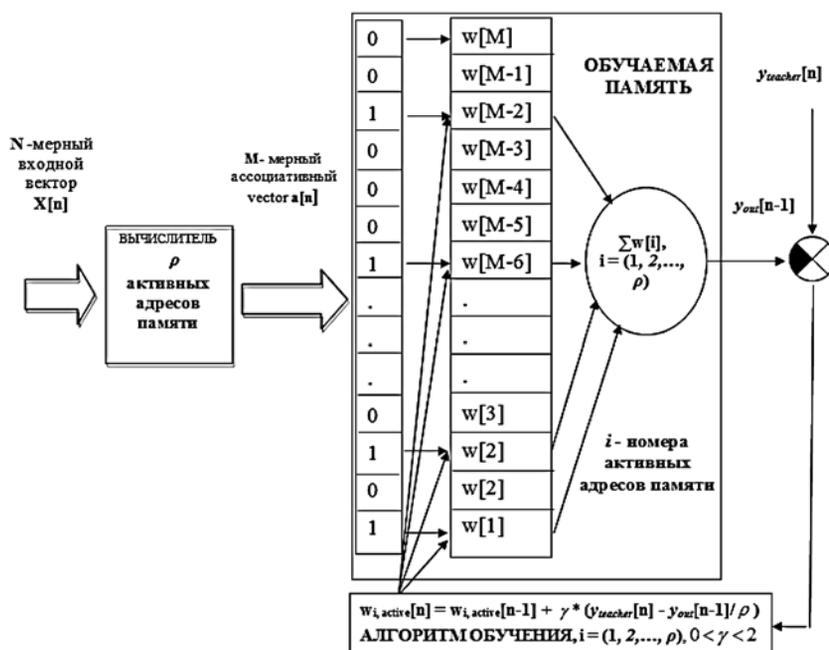


Рис. 2. Структура НС СМАС

Таблица 4

Состав обучающих записей нижнего уровня

Номер	Компоненты DoS-атак	Число обучающих записей компонентов
1	Back	726
2	Neptune	181 611
3	Teardrop	688
4	Land	14
5	Pod	154
6	Smurf	2255
	Итого	185 448

Тестирование обученной системы проводится по всем 889 544 записям из базы данных атак KDD Cup 99, за исключением 75 % записей DoS-атак, которые были использованы для обучения системы. Тестовые данные состоят из $N_{DoS} = 61\ 819$ записей DoS-атак и $N_{NotDoS} = 827\ 725$ записей, которые содержат три класса атак: Probe, R2L и U2R и нормальные соединения.

5. Экспериментальные результаты

5.1. Обучение и тестирование НС СМАС верхнего уровня обнаружению DoS-атак

Обучение. Обучение НС СМАС верхнего уровня обнаружению DoS-атак выполняется с помощью 185 448 записей о DoS-атаках (75 % от общего числа DoS-атак). Из этих записей выделяют значения пяти признаков с номерами {3, 4, 5, 23, 30}. Максимальные значения признаков 3 и 4 согласно выражениям (3) принимают равными $x_{\max}^{(1)} = 129$ и $x_{\max}^{(2)} = 17$, соответственно. Максимальные значения признаков 5, 23, 30 принимают несколько значений. Для признака 5 максимальные значения $x_{\max}^{(3)}$ равны 129, 257, 513, 1025 и 2049, для признака 23 $x_{\max}^{(4)}$ равны 33, 65, 129, 257 и 513, а для признака 30 $x_{\max}^{(5)}$ равны 33, 65 и 129. Признаки квантуются с различными шагами квантования согласно выражениям (3) и (4). После квантования в полученных выборках возникают повторы квантованных записей, которые удаляются. При этом число записей для обучения, которое зависит от введенных шагов квантования по соответствующим переменным, существенно уменьшается, более чем на порядок.

Как отмечалось в разд. 3, на точность обучения оказывает существенное влияние значение обобщающего параметра ρ , которое в экспериментах согласно выражению (3) принимает значения $\rho = 2, 4, 8, 16$. Кроме того, точность обнаружения атак также зависит от значения порога обнаружения Δ (5), который принимает значения 0,2, 0,3, 0,4 и 0,5.

Обучение НС СМАС обнаружению DoS-атак проводится для каждого отдельного набора записей при фиксированных значениях параметров ρ и Δ .

Перед процессом обучения память НС СМАС обнуляется, на ее вход подается 5-мерный вектор дискретных признаков, который случайным образом извлекается из обучающего набора данных. Число шагов обучения N принято заведомо завышенным и равняется $N = 500\ 000$. Значение указаний учителя для DoS-атаки принимается равным 1. После завершения процесса обучения НС СМАС подвергается тестированию. Поскольку в результате тестирования выясняется область возможных оптимальных значений шагов квантования и значений обобщающего параметра, то для нахождения оптимальных параметров НС СМАС используется только часть из 75 наборов выборок с различными шагами квантования при различных значениях обобщающего параметра и порога.

Тестирование. Тестирование обученной НС СМАС верхнего уровня выполняется подачей на ее вход квантованных значений пяти признаков с номерами {3, 4, 5, 23, 30} 889 544 записей из сжатой базы данных атак KDD Cup 99, в которых отсутствуют 75 % использованных для обучения записей DoS-атак. Признаки записей квантуются с теми же шагами квантования, которые использовались для обучения нейронной сети. О каждой записи известно, к какому она относится классу: 0 — отсутствует DoS-атака, 1 — присутствует DoS-атака. Выход нейронной сети позволяет определить оценки вероятностей ошибок первого и второго рода системы обнаружения DoS-атак. При этом ошибка первого рода возникает, когда входная запись не является DoS-атакой, а система относит ее к DoS-атаке (ложная тревога), а ошибка второго рода возникает тогда, когда на вход системы поступает DoS-атака, а система ее не обнаруживает (пропуск атаки). Оценки вероятностей ошибок первого $\hat{\alpha}$ и второго $\hat{\beta}$ рода определяются следующими соотношениями:

$$\hat{\alpha} = \Delta N_{NotDoS} / N_{NotDoS}, \quad \hat{\beta} = \Delta N_{DoS} / N_{DoS}, \quad (6)$$

где $N_{NotDoS} = 827\ 725$ — число не DoS-атак, $N_{DoS} = 61\ 819$ — число DoS-атак для всех случаев тестирования. Величины ΔN_{NotDoS} — число ложных тревог, ΔN_{DoS} — число пропущенных DoS-атак в (6) зависят от выбранных параметров системы обучения, определяют ее точностные характеристики и их вычисляют в результате каждого эксперимента. Кроме значений параметров (6) вычисляется значение однопараметрической характеристики системы

$$\hat{\gamma} = (\hat{\alpha}^2 + \hat{\beta}^2)^{1/2}, \quad (7)$$

в которой ошибки первого и второго рода сворачиваются с одинаковыми весами.

5.2. Результаты обучения и тестирования НС СМАС верхнего уровня

В результате обучения и тестирования 216 вариантов нейронной сети СМАС верхнего уровня, для

**Параметры обучения НС СМАС обнаружению
компонентов DoS-атак**

Компоненты DoS-атак	Номера признаков	Минимальное значение квантованного признака	Максимальное значение квантованного признака	Число записей для обучения	Порог Δ	ρ
Back	3 10	1 1	129 9	4	0,5	2
Neptune	4 30	1 1	33 129	87	0,5	2
Teardrop	2 8	1 1	5 5	3	0,5	2
Land	7 37	1 1	3 129	7	0,5	2
Pod	5 8	1 1	2049 5	3	0,5	2
Smurf	5 23	1 1	2049 513	850	0,5	2

которой шаги квантования, значения обобщающего параметра и значение порога принимали различные значения, были получены значения параметров $\hat{\alpha}$, $\hat{\beta}$ и $\hat{\gamma}$, характеризующие точность системы обнаружения DoS-атак.

В наилучшей системе обнаружения DoS-атак по критерию наименьшего значения параметра $\hat{\gamma}$ значения $\rho = 4$ и $\Delta = 0,5$, а максимальные значения компонент входного вектора признаков (1) принимают значения

$$X_{opt} = \{x^{(1)} = \overline{1,129}; x^{(2)} = \overline{1,17}; x^{(3)} = \overline{1,513}; x^{(4)} = \overline{1,65}; x^{(5)} = \overline{1,33}\}. \quad (8)$$

В этой близкой к оптимальной системе число ложных тревог $\Delta N_{NotDoS} = 1336$, число пропусков DoS-атак $\Delta N_{DoS} = 39$. С учетом выражений (6) и (7) оценки вероятностей ошибок первого $\hat{\alpha}$ и второго рода $\hat{\beta}$ будут $\hat{\alpha} = 0,0016$, $\hat{\beta} = 0,0006$. Значение параметра $\hat{\gamma} = 0,0017$. Из этих результатов следует, что оценка вероятности обнаружения DoS-атак для такой оптимальной системы равна $\hat{P}_{DoS} = 0,9994$. Доля ложных тревог составляет 0,02 от общего числа DoS-атак.

Для другой системы обнаружения DoS-атак, близкой по оптимальности к описанной выше, в которой обобщающий параметр и порог приняли новые значения: $\rho = 8$ и $\Delta = 0,2$, а максимальные значения компонент входного вектора признаков (8) остались без изменения, все атаки были обнаружены: $\Delta N_{DoS} = 0$, однако возросло число ложных тревог $\Delta N_{NotDoS} = 2549$. Для такой системы $\hat{\alpha} = 0,0031$, $\hat{\beta} = 0,0$. Значение параметра $\hat{\gamma} = 0,0031$.

5.3. Результаты обучения и тестирования НС СМАС нижнего уровня обнаружения компонентов DoS-атак

Обучение всех шести НС СМАС нижнего уровня проводится в полной аналогии с описанными выше процессами обучения НС СМАС верхнего уровня.

Из записей DoS-атак для обучения выделяются значения признаков компонентов DoS-атак: Back — {3,10}, Neptune — {4,30}, Teardrop — {2,8}, Land — {7,37}, Pod — {5,8} и Smurf — {5,23}, признаки квантуются и из них удаляются повторы. В результате этих процедур для обучения обнаружению компонентов DoS-атак число записей для компонентов Back, Neptune, Teardrop, Land, Pod и Smurf существенно уменьшается, в том числе вследствие наличия только двух признаков для каждого компонента, и оказывается равным 3, 87, 3, 3, 7 и 850, соответственно. В силу небольшого числа обучающих записей для нижнего уровня число шагов обучения, которое зависит от номеров компонентов DoS-атак, существенно ниже, чем для верхнего уровня. Проведенный предварительный анализ возможных значений параметров нижнего уровня обнаружения компонентов DoS-атак определил

максимальные значения признаков, значения обобщающего параметра и порога, значения которых приведены в табл. 5. Указание учителя в режиме обучения принимается равным единице. Отметим, что параметры нижнего уровня не оптимизированы, оптимизация последних является предметом последующих исследований. Следует также учесть, что в базе данных атак KDD Cup 99 число некоторых компонентов DoS-атак незначительно, например для компонентов Land и Pod (см. табл. 4).

Тестирование несколько отличается от тестирования верхнего уровня: в роли тестовых данных выступают те записи, которые НС СМАС верхнего уровня приняла за DoS-атаку. Из этих записей выделяются соответствующие каждому компоненту DoS-атак признаки, они квантуются с теми же шагами квантования, которые были использованы при обучении каждой из шести НС СМАС, и эти данные подаются на входы каждой нейронной сети. Если несколько НС дают сообщение относительно обнаружения компонента DoS-атаки, то предпочтение отдается той сети, у которой выше аналоговый выход.

В табл. 6 приведены результаты тестирования нижнего уровня системы обнаружения DoS-атак для случая, когда параметры верхнего уровня оптимальны и соответствуют описанным в разделе 5.2 значениям, для которых $\hat{\gamma} = 0,0017$. Число записей для тестирования равно 63 116, из которых 61780 представляют записи DoS-атак.

Из табл. 6 следует, что значения параметров нижнего уровня системы обнаружения DoS-атак и их компонентов близки к оптимальным. Вероятность обнаружения большинства компонентов DoS-атак (Teardrop, Land, Pod, Smurf) для такой системы практически равна единице. Несколько хуже ре-

Результаты тестирования нижнего уровня системы обнаружения компонентов DoS-атак

Компоненты DoS-атак	Точное число компонентов в выборке	Точное число некомпонентов в выборке	Число принятых за компонент	Число пропущенных компонентов	$\hat{\alpha}$	$\hat{\beta}$	$\hat{\gamma}$
Back	242	62 874	5	3	0,0001	0,0124	0,0124
Neptune	60 499	2617	461	61	0,1762	0,0010	0,1762
Teardrop	230	62 886	0	0	0	0	0
Land	5	63 111	6	0	0,0001	0	0,0001
Pod	52	63 064	0	0	0	0	0
Smurf	752	62 364	129	0	0,0021	0	0,0021

ультаты для компонента Back, для которого эта оценка равна $P_{Back} = 0,9876$. Отметим также несколько высокую вероятность ложной тревоги для компонента Neptune, которая равна $\hat{\alpha} = 0,1762$.

6. Заключение

Приведенные выше результаты обучения и тестирования системы обнаружения DoS-атак и их компонентов на базе нейронных сетей СМАС расширяют результаты, приведенные в работе [2], и подтверждают утверждения Дж. Кеннеди [1] о большой перспективности НС СМАС как аналитического инструмента обнаружения атак на информационные ресурсы. Для развития полученных в работе результатов следует провести исследования, используя информацию из других баз данных и другие наборы признаков применительно не только к DoS-атакам, но и к другим видам атак.

Список литературы

1. Cannady J. Next Generation Intrusion Detection: Autonomous Reinforcement Learning of Network Attacks // Proc. of the 23-rd

National Information Systems Security Conference, October 16–19, 2000, Baltimore, MD, USA, 2000. (<http://csrc.nist.gov/nissc/2000/proceedings/toc.html>).

2. Авдьян Э. Д., Ле Т. Ч. Л. Нейронная сеть СМАС в задаче обнаружения атак на информационные ресурсы // Информатизация и связь. 2015. № 4. С. 93–98.

3. KDD Cup 1999 Data: <http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html>.

4. Bhorja P., Garg K. Determining feature set of DoS-attacks // International Journal of Advanced Research in Computer Science and Software Engineering. 2013. Vol. 3. N. 5. P. 875–878.

5. Tavallae M., Bagheri E., Lu W., Ghorbani A. A. A Detailed analysis of the KDD Cup 99 data set // Proc. of the Second IEEE Symposium on Computational Intelligence for Security and Defense Applications 2009, Ottawa, 2009. P. 53–58.

6. Albus J. S. A new approach to manipulator control: the cerebellar model articulation controller // ASME Trans., J. Dynamic Systems, Measurement and Control. 1975. Vol. 97. N. 3. P. 220–227.

7. Albus J. S. Data storage in the cerebellar model articulation controller (CMAC) // ASME Trans., J. Dynamic Systems, Measurement and Control. 1975. Vol. 97. N. 3. P. 228–233.

8. Авдьян Э. Д. Ассоциативная нейронная сеть СМАС. Часть I. Структура, объем памяти, обучение и базисные функции // Информационные технологии. 1997. № 5. С. 6–14.

9. Авдьян Э. Д. Ассоциативная нейронная сеть СМАС. Часть II. Процессы обучения, ускоренное обучение, влияние помех, устранение влияния помех в двухслойной сети // Информационные технологии. 1997. № 6. С. 16–27.

E. D. Aved'yan^{1, 2, 3}, Senior research Fellow¹, Deputy Head², Professor³, T. T. L. Le³, Postgraduate Student

¹Department of Advanced Research and special Projects of the Federal State Autonomous Research Institution CIT&S, Moscow, Russia

²The Neural Network Technology Centre of the International Centre of Informatics and Electronics, Moscow, Russia

³Department of Radio Engineering and Cybernetics of the Moscow Institute of Physics & Technology (State University), Moscow, Russia

A Two-Level System for DoS attacks and their Components Detection based on the Neural Networks CMAC

The results of the application of the system of neural networks CMAC (NN CMAC) to detect DoS attacks and their components are given. The system used all the data in the database KDD Cup 99. The system consists of two levels. The first level is designed to detect DoS attacks using trained NN CMAC. The second level is designed to separate from the detected DoS attacks all the 6 components: Back, Neptune, Land, Pod, Teardrop and Smurf using 6 trained NN CMAC. Miss rate of DoS attacks and false alarm in the first level does not exceed 0,2 %.

Keywords: Detection of DoS attacks, the neural network CMAC, attacks, database KDD Cup 99, the components of DoS attacks

References

1. **Cannady J.**, Next Generation Intrusion Detection: Autonomous Reinforcement Learning of Network Attacks, *Proceedings of the 23-rd National Information Systems Security Conference, October 16–19, 2000, Baltimore, MD, USA, 2000* (<http://csrc.nist.gov/nissc/2000/proceedings/toc.html>).
2. **Aved'yan E. D., Le T. Ch. L.**, Nejronnaja set' SMAS v zadache obnaruzhenija atak na informacionnye resursy (The Neural network CMAC in the Problem of Intrusion Detection on the Information Resources), *Informatizacija i svjaz*, 2015, no. 4, pp. 93–98.
3. **KDD Cup 1999** Data: <http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html>.
4. **Bhoria P, Garg K.**, Determining feature set of DoS attacks, *International Journal of Advanced Research in Computer Science and Software Engineering*, 2013, vol. 3, no. 5, pp. 875–878.
5. **Tavallaee M., Bagheri E., Lu W., Ghorbani A. A.** A Detailed analysis of the KDD Cup 99 data set, *Proceedings of the Second IEEE Symposium on Computational Intelligence for Security and Defense Applications 2009, Ottawa, 2009*, pp. 53–58.
6. **Albus J. S.**, A new approach to manipulator control: the cerebellar model articulation controller, *ASME Trans., J. Dynamic Systems, Measurement and Control*, 1975, vol. 97, no. 3, pp. 220–227.
7. **Albus J. S.**, Data storage in the cerebellar model articulation controller (CMAC), *ASME Trans., J. Dynamic Systems, Measurement and Control*, 1975, vol. 97, no. 3, pp. 228–233.
8. **Aved'yan E. D.**, Associativnaja nejronnaja set' SMAS. Chast' I Struktura, ob#em pamjati, obuchenie i bazisnye funkicii (The Associative CMAC Neural Network. Part I. The Structure, Memory, Learning and Basis functions), *Informacionnye tehnologii*, 1997, no. 5, pp. 6–14.
9. **Aved'yan E. D.**, Associativnaja nejronnaja set' SMAS. Chast' II. Processy obuchenija, uskorennoe obuchenie, vlijanie pomeh, us-tranenie vlijanija pomeh v dvuhslojnoj seti (The Associative CMAC Neural Network. Part II. The Learning Processes, Accelerated Learning, Noise Influences, Noise Elimination in the Two-Layer CMAC Network). *Informacionnye tehnologii*, 1997, no. 6, pp. 16–27.

Адрес редакции:

107076, Москва, Стромьинский пер., 4

Телефон редакции журнала (499) 269-5510

E-mail: it@novtex.ru

Технический редактор *Е. В. Конова*.

Корректор *Е. В. Комиссарова*.

Сдано в набор 08.07.2016. Подписано в печать 23.08.2016. Формат 60×88 1/8. Бумага офсетная.

Усл. печ. л. 8,86. Заказ IT916. Цена договорная.

Журнал зарегистрирован в Министерстве Российской Федерации по делам печати, телерадиовещания и средств массовых коммуникаций.

Свидетельство о регистрации ПИ № 77-15565 от 02 июня 2003 г.

Оригинал-макет ООО "Авансед солюшнз". Отпечатано в ООО "Авансед солюшнз".

119071, г. Москва, Ленинский пр-т, д. 19, стр. 1.



С 28 по 30 октября 2016 г. в Москве состоится

двенадцатая конференция

CEE-SEC / РАЗРАБОТКА ПО —

это одно из главных ежегодных ИТ событий России, имеющих статус "In-Cooperation with ACM SIGSOFT".

В Программу конференции войдут приглашенные и конкурсные доклады, панельные дискуссии, а также открытые обсуждения. Программа ориентирована как на опытных специалистов, так и на тех, кто стремится набрать начальный опыт, в том числе в смежных с основной специализацией областях.

ТРЕНДОВЫЕ ТЕМЫ

- Распределенные приложения на сети устройств и сервис-ориентированная архитектура.
- Интернет вещей, робототехника и встраиваемые системы.
- Наука о данных, большие данные, высоконагруженные системы.
- Облачные модели использования ресурсов.
- Улучшенное машинное обучение и контекстно-зависимые системы.
- Применение веб-технологий в разработке крупных проектов.
- ИТ в медицине и смежных областях (биотехнологии, нейро-science)
- Наукоемкие стартапы.

ТРАДИЦИОННЫЕ ТЕМЫ

- Технологии программирования.
- Системный анализ и инженерия требований.
- Управление проектом и продуктом: классические и гибкие методологии.
- Тестирование, верификация и анализ ПО.
- Архитектура программных систем.
- Безопасность и надежность систем и ПО.
- Специфика разработки мобильных и кросс-платформенных приложений.
- Взаимодействие человека и компьютера.
- Открытый код: технологии и сообщество.
- Человеческий капитал, поиск и подбор сотрудников.
- Бизнес и предпринимательство в сфере разработки ПО.
- Коммуникации — эффективность и управление.
- ИТ как средство государственного управления.

В программу войдут:

- Выступления приглашенных экспертов российского и мирового ИТ-рынка;
- Доклады, отобранные на конкурсной основе;
- Дискуссии и мастер-классы;
- Специальная Networking Party.

Подробную информацию о конференции см. на сайте
<http://2016.secr.ru/>



с 7 по 8 декабря 2016 года,
в павильоне № 75 ВВЦ состоится

VI Международная научно-практическая конференция "АВТОМАТИЗАЦИЯ И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ЭНЕРГЕТИКЕ 2016"

Тема конференции:

"Современное состояние и тенденции развития информационно-управляющих систем и телекоммуникаций в энергетике (контроль, учет, управление, телекоммуникации, безопасность)"

НАПРАВЛЕНИЯ КОНФЕРЕНЦИИ:

- Информационно-технологические системы реального времени, включая системы управления, телемеханики и связи, мониторинга и диагностики (АСТУ, АИИС КУЭ, АСДУ, РЗА, интеллектуальные активно-адаптивные сети Smart Grid, цифровые подстанции и т.п.).
- Обеспечение безопасности информационных систем объектов топливно-энергетического комплекса, защита от внешних угроз, киберугрозы и пр.
- Моделирование и оптимизация производства, распределения и потребления энергоресурсов.
- Вопросы импортозамещения при создании современных информационно-управляющих и телекоммуникационных систем в энергетике.
- Эксплуатация информационных и телекоммуникационных систем, применение "необслуживаемого" энергоэффективного оборудования.
- Современные летающие лаборатории на базе БПЛА для мониторинга объектов ТЭК, опыт применения.

Круглые столы:

1. "Обеспечение информационной безопасности — глобальная проблема Человечества, мощный стимул в развитии использования информационных технологий в энергетике".
2. "Системы связи — основа информационных технологий и телекоммуникаций в электроэнергетике. Проблемы, решения, векторы и драйверы развития".

Подробную информацию о конференции см. на сайте
www.avite.ru