

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Том 22
2016
№ 4

ТЕОРЕТИЧЕСКИЙ И ПРИКЛАДНОЙ НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ

Издается с ноября 1995 г.

УЧРЕДИТЕЛЬ
Издательство "Новые технологии"

СОДЕРЖАНИЕ

ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

- Тарасов А. Д. Эффективность работы генетического алгоритма в задаче проектирования систем физической защиты 243
- Николаев А. И. Эффективный подход на основе машинного обучения к решению задачи о максимальной клике 249

МОДЕЛИРОВАНИЕ И ОПТИМИЗАЦИЯ

- Дмитриев А. В., Мальцева С. В., Цуканова О. А. Моделирование и качественный анализ социальной микроблогинговой сети как динамической системы 255
- Дягилев В. И., Коковин В. А., Увайсов С. У., Увайсова С. С. Компьютерное моделирование работы силового преобразователя с выходным синусоидальным напряжением. 261

СИСТЕМЫ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ

- Олейник П. П., Гурьянов В. И. UML-профиль для метамодельно-ориентированного проектирования программных приложений баз данных. 267
- Бибило П. Н., Логинова И. П. Формирование энергоемких тестов для комбинационных логических схем по результатам оценки их энергопотребления. 277

БЕЗОПАСНОСТЬ ИНФОРМАЦИИ

- Авдошин С. М., Лазаренко А. В. Технология анонимных сетей. 284

ЦИФРОВАЯ ОБРАБОТКА СИГНАЛОВ И ИЗОБРАЖЕНИЙ

- Лютикова Л. А., Шматова Е. В. Анализ и синтез алгоритмов распознавания образов с использованием переменного-значной логики. 292

ДИСКУССИОННЫЙ КЛУБ

- Цветков В. Я. Полисемия информации 298

Журнал в журнале НЕЙРОСЕТЕВЫЕ ТЕХНОЛОГИИ

- Галушкин А. И., Пантюхин Д. В. СуперЭВМ и мемристоры 304
- Задаянчук А. И., Попова М. С., Стрижов В. В. Выбор оптимальной модели классификации временных рядов. 313

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.

INFORMATION TECHNOLOGIES

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Vol. 22
2016
No. 4

THEORETICAL AND APPLIED SCIENTIFIC AND TECHNICAL JOURNAL

Published since November 1995

ISSN 1684-6400

CONTENTS

INTELLIGENT SYSTEMS AND TECHNOLOGIES

- Tarasov A. D.** Genetic Algorithm Effectiveness Used in Problem of Physical Defense System Designing 243
- Nikolaev A. I.** Efficient Approach for the Maximum Clique Problem Based on Machine Learning 249

MODELING AND OPTIMIZATION

- Dmitriev A. V., Maltseva S. V., Tsukanova O. A.** Modeling and Qualitative Analysis of a Social Microblogging Network as a Dynamical System 255
- Diagilev V. I., Kokovin V. A., Uvaysov S. U., Uvaysova S. S.** Computer Simulation of the Power Converter with Harmonic Wave Output 261

CAD-SYSTEMS

- Oleynik P. P., Gurianov V. I.** UML-Profile for Metamodel-Driven Design of Database Applications 267
- Bibilo P. N., Loginova I. P.** The Creation of Energy-Intensive Tests for Combinational Logic Circuits According to the Results of Evaluation of Their Power Consumption 277

CRYPTOSAFETY INFORMATION

- Avdoshin S. M., Lazarenko A. V.** Technology of Anonymous Networks. 284

DIGITAL PROCESSING OF SIGNALS AND IMAGES

- Lyutikova L. A., Shmatova E. V.** Recognition Algorithms Analysis and Synthesis with Varied Values Logic 292

DISCUSSION CLUB

- Tsvetkov V. Ya.** Polysemy Information. 298

Journal-in-journal NEUROTECHNOLOGIES

- Galushkin A. I., Pantiukhin D. V.** Supercomputers and Memristors. 304
- Zadayanchuk A. I., Popova M. S., Strijov V. V.** Selection of Optimal Time Series Classification Model. 313

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"

А. Д. Тарасов, ст. преподаватель каф. "Автоматизированные системы обработки информации и управления", e-mail: adtarasov@mail.ru
Оренбургский государственный аграрный университет

Эффективность работы генетического алгоритма в задаче проектирования систем физической защиты

Рассмотрен процесс решения задачи проектирования систем физической защиты с помощью стандартного генетического алгоритма. Выделены параметры алгоритма с наибольшим влиянием на процесс решения. Описаны эксперименты, доказывающие необходимость определения значений этих параметров с помощью адаптивного генетического алгоритма.

Ключевые слова: система физической защиты, поддержка принятия решений, стандартный генетический алгоритм, адаптивный генетический алгоритм

Введение

Под системами физической защиты (СФЗ) понимается совокупность инженерно-технических, сигнализационных, программно-аппаратных и иных средств для защиты имущества или объектов от хищений, диверсий и других неправомерных действий, позволяющая на заданном уровне создавать трудности, ограничивать возможности и увеличивать время проникновения нарушителя на объект охраны [1].

Критически важными объектами (КВО) являются объекты, нарушение функционирования которых приводит к потере управления, разрушению инфраструктуры, необратимому негативному изменению экономики субъекта или административной территории либо существенному ухудшению на длительный период времени безопасности жизнедеятельности населения, проживающего на этих территориях [2].

Обеспечение безопасности КВО является важнейшей задачей, и процесс создания СФЗ КВО требует для проектировщика использования любых доступных средств, которые могут предоставить современные информационные технологии. Одним из таких средств является поддержка принятия решений.

Задача поддержки принятия решений при проектировании СФЗ и их анализе требует информации, отражающей опыт и знания экспертов, предоставления возможных вариантов решения, оценки этих решений и обоснованного выбора из них. В задаче принятия решения о составе СФЗ самым сложным и плохо формализуемым этапом, требующим применения процедур, которые используют знания экспертов, является концептуальное проектирование, под которым понимается анализ защищенности объекта, разработка практических рекоменда-

ций по созданию СФЗ, выбор варианта и состава инженерно-технических средств охраны (ИТСО).

Для решения задач такого рода предлагается использовать генетические алгоритмы (ГА). Так как задача обеспечения защиты критически важных объектов требует использования высокоэффективных методов решения, то при разработке ГА необходимо обеспечить максимальное качество работы алгоритма.

Генетические алгоритмы имеют множество преимуществ по сравнению с другими методами оптимизации [3], но также не лишены недостатков, один из которых — сильная зависимость эффективности работы алгоритма от выбранных параметров ГА. Такими параметрами являются вероятность кроссинговера, вероятность мутаций, число точек деления хромосом при кроссинговере и т. п. Обычно поиск лучших значений параметров проводится интуитивно, методом проб и ошибок, что приводит к выбору не самых удачных вариантов. Проведение экспериментов с оценкой работоспособности алгоритма при различных значениях параметров часто решает эту проблему.

Дополнительная сложность при определении наилучших значений параметров алгоритма вызывается следующей особенностью ГА: влияние многих параметров на эффективность работы изменяется на протяжении всего времени работы ГА. Следовательно, нужно определить не постоянные значения, а правила изменения параметров в процессе поиска решения. Например, желательно уменьшать вероятность мутации в зависимости от времени работы генетического алгоритма [4]. Кроме того, изменение параметров ГА может потребоваться при смене исходных данных задачи.

Таким образом, поиск наилучших значений параметров ГА надо проводить при каждом его за-

пуске, что возможно при использовании так называемых адаптивных генетических алгоритмов.

Выдвигаем предположение, что для решения задачи проектирования СФЗ критически важных объектов требуется разработка ГА с обязательной реализацией механизма адаптации.

В данной работе анализируется процесс решения задачи выбора состава ИТСО с помощью стандартного ГА, описанный в работе [5]. Выделены параметры ГА, значения которых сильно влияют на эффективность работы алгоритма, и проведены эксперименты, доказывающие необходимость определения значений этих параметров с помощью адаптивного ГА.

1. Использование ГА в задаче выбора состава ИТСО объекта

Понятие генетического алгоритма и особенности стандартного ГА описаны в работах [3] и [6].

Для выбора состава ИТСО с помощью стандартного ГА применяется логическая модель, которая представляет структуру объекта в виде графа, где вершины — это зоны объекта, а ребра — рубежи. Часть зон является критическими элементами (КЭ) объекта, требующими защиты. Защита КЭ обеспечивается комплексом ИТСО, состав которого представляется в виде множества точек контроля (ТК). Физически каждая ТК может включать в себя несколько средств комплекса ИТСО, выполняющих одну общую функцию: точка обнаружения (ТО); точка доступа (ТД); точка видеонаблюдения (ТВ); точка задержки (ТЗ). Итого можно выделить четыре типа точек контроля, причем для защиты одной зоны может потребоваться несколько точек. Решение задачи нахождения оптимального расположения ТК на графе объекта подробно описано в работе [5].

Целевая функция ГА данной задачи состоит из двух частей. Первая часть функции отвечает за то, чтобы набор ТК на каждом из всех возможных путей к КЭ соответствовал требуемому набору ТК для данного КЭ или превышал его, т. е. все КЭ должны быть защищены. Вторая часть функции отвечает за минимизацию общего числа ТК на объекте (чем меньше ТК, тем меньше затраты на создание комплекса ИТСО).

Для получения оптимального решения соблюдение условий первой части целевой функции важнее, чем соблюдение условий второй части, так как если не все КЭ защищены, проект системы защиты непригоден независимо от затрат. Поэтому в задаче используются так называемые уровни влияния частей целевой функции [5]. Два числа A и B , соотношение которых задает превышение значения одной части целевой функции над другой, участвуют в расчете эффективностей хромосом:

$$\mu(h_i) = (\mu_1(h_i) \times A + \mu_2(h_i) \times B) / (A + B),$$

где $\mu_1(h_i)$ — значение эффективности i -й хромосомы для первой части целевой функции, $\mu_2(h_i)$ — значение эффективности для второй части целевой функции.

Чем больше разница между числами A и B , тем сильнее одна часть целевой функции влияет на эффективность хромосом (и на получаемые решения). Делаем вывод, что уровни влияния частей целевой функции являются одними из параметров ГА, от которых сильно зависит эффективность работы ГА. Эксперименты, представленные ниже, показали, насколько меняется скорость поиска оптимального решения при изменении отношения уровней влияния.

Вероятность мутации также является одним из значимых параметров ГА, оптимальное значение которого непредсказуемо, но влияет как на скорость работы, так и на саму возможность получения оптимального решения.

Таким образом, в процессе решения задачи выбора состава ИТСО объекта с использованием стандартного ГА определяем два параметра, значения которых, предположительно, являются наиболее важными для эффективной работы алгоритма. Это уровни влияния частей целевой функции и вероятность мутации хромосом.

2. Оценка оптимальности решения, полученного с помощью ГА

Набор исходных данных для тестирования работоспособности алгоритма описывает условный объект (рис. 1). Граф объекта достаточно простой для поиска путей вручную, что будет использовано при оценке оптимальности решения. Исходные данные следующие: точки проникновения — зона 1 и 4, критические элементы — зона 10 и 12. Ограничения на число ТК у всех вершин и ребер одинаковое: минимальное — ноль, максимальное — три для ТК каждого типа. Требования по ТК для критических элементов выбраны случайным образом (табл. 1).

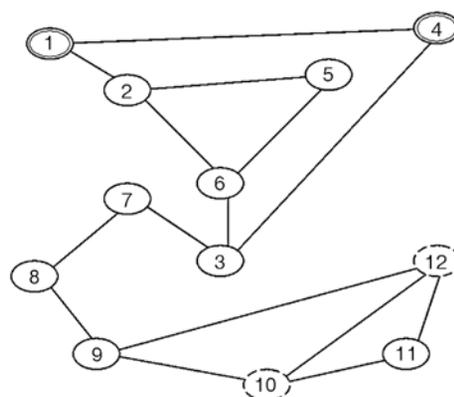


Рис. 1. Граф условного объекта

Таблица 1

Требования по числу ТК для условного объекта

Критические элементы	Точки обнаружения (ТО)	Точки доступа (ТД)	Точки видеонаблюдения (ТВ)	Точки задержки (ТЗ)
Зона 10	25	1	17	4
Зона 12	10	17	16	3

Набор точек контроля в оптимальном решении для условного объекта

Зона или рубеж	ТО	ТД	ТВ	ТЗ
Зона 3	3	3	2	0
Зона 4	1	0	0	0
Зона 7	3	3	1	1
Зона 8	3	3	3	0
Зона 9	3	3	3	1
Зона 10	3	0	1	1
Зона 12	0	2	0	0
Рубеж 6—3	1	0	0	0
Рубеж 3—7	3	1	3	1
Рубеж 7—8	3	2	1	0
Рубеж 8—9	3	0	3	0
Сумма	26	17	17	4

Для проверки оптимальности полученного генетическим алгоритмом результата нужно найти решение задачи вручную. Если существует несколько оптимальных решений (с одинаковым соответствием целевой функции), то необходимо определить, какие общие свойства должны иметь решения, чтобы считаться оптимальными. На основе рассчитанных свойств можно доказать, что результат, полученный ГА, оптимален.

Проанализируем задачу с текущим набором исходных данных. Оптимальное решение должно максимально возможно соответствовать целевой функции (защита всех КЭ при минимально возможном числе ТК). Определяем свойства оптимального решения через следующие умозаключения.

Число ТК на всех путях (на всем объекте) в идеальном случае строго равно требуемому числу ТК для КЭ (меньшее число не сможет защитить все КЭ, большее нежелательно по правилам второй части целевой функции). Если КЭ несколько, то в идеальном наборе ТК число ТК каждого типа равно максимальному среди требований для защиты каждого КЭ. Другими словами, минимальное число ТК на объекте должно соответствовать или превышать требуемое для защиты каждого КЭ, а значит, в идеале равно наибольшему числу среди заданных наборов ТК, требуемых для всех КЭ. По табл. 1 определяем идеальный набор ТК на всем объекте: 25 ТО, 17 ТД, 17 ТВ, 4 ТЗ.

Теперь проверим, возможно ли защитить все пути таким минимально допустимым набором точек. Фактически требуется найти вершины или ребра, через которые идут все пути, и расставить на них нужное число ТК. Таких вершин и ребер в графе семь. Все они расположены на участке графа между вершинами 3 и 9: 3—7—8—9. Ограничения на число ТК в вершинах и ребрах позволяют разместить на этом участке не более 21 ТК каждого вида (по три на вершинах и ребрах). Этого достаточно для ТД, ТВ и ТЗ, значит, их число в оптимальном решении равно 17, 17 и 4 соответственно.

Нужное число точек обнаружения не помещается на выделенном участке графа. 21 ТО в идеальном решении расположены на этом участке. Оставшиеся четыре ТО требуют поиска оптимального расположения в других участках. Эти точки должны защищать только зону 10, так как для защиты зоны 12 достаточно 21 точки, которые уже размещены. Расположим три допустимые по ограничениям ТО в самой зоне 10. Остается одна недостающая ТО.

Согласно графу объекта одна ТО в любых вершинах и ребрах, оставшихся не заполненными точками обнаружения, не сможет защитить все пути. Граф разветвляется на два направления до и после участка 3—7—8—9. От-

сюда вывод: необходимо разместить на графе минимум две ТО для защиты всех путей. Эти точки могут быть расположены практически в любых оставшихся незанятыми вершинах или ребрах, но только с условием, что любой путь пройдет хотя бы через одну из двух точек. Например, участки графа 9—12; 9—10 или 3—6; 3—4 или обе точки проникновения — зоны 1 и 4.

Таким образом, в оптимальном решении должно быть на одну ТО больше, чем в идеальном наборе: 26 ТО, 17 ТД, 17 ТВ, 4 ТЗ — всего 64 точки контроля. Если генетическим алгоритмом будет найдена хромосома с подобным набором ТК, значит, алгоритм пригоден для поиска оптимальных решений.

Решение было найдено программой [7] несколько раз с различными настройками ГА (см. следующий раздел). Полученные решения не полностью соответствуют предсказанному расположению, но набор ТК всегда равен оптимальному: 26 ТО, 17 ТД, 17 ТВ, 4 ТЗ. Пример решения в интерфейсе программы показан на рис. 2 и в табл. 2.

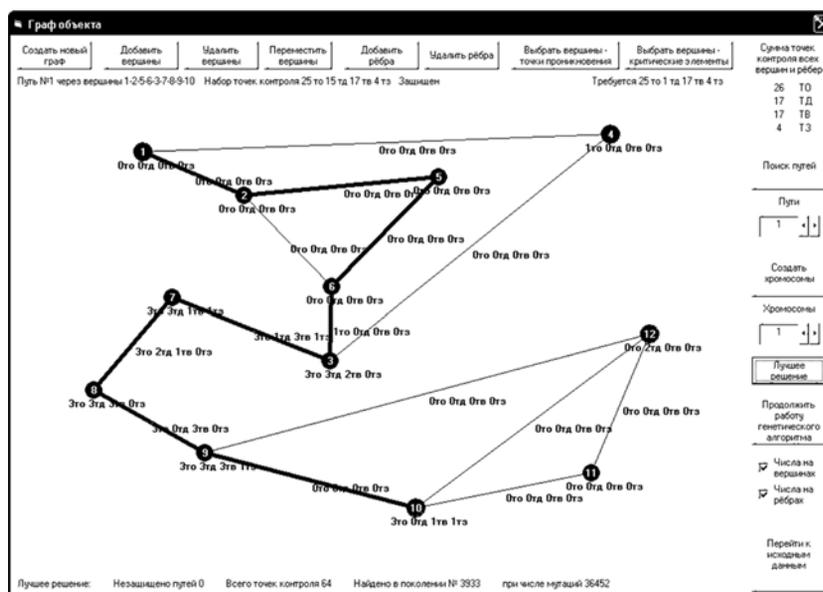


Рис. 2. Оптимальное решение для условного объекта

3. Проверка зависимости работы ГА от уровня влияния частей целевой функции и вероятности мутации

Проведем эксперименты для выяснения наилучших значений параметров задачи, при которых скорость поиска решения наибольшая.

В проводимых экспериментах доказывается, что неравные уровни влияния для двух частей целевой функции значительно ускоряют поиск решения по сравнению с равным (при большем влиянии первой части функции). Также проверяется скорость работы при различных соотношениях уровней влияния.

Вторым проверяемым параметром задачи является вероятность мутации. Чем выше эта вероятность, тем быстрее среди хромосом появляются решения, более близкие к оптимальному решению. Если же мутации появляются часто, то происходит слишком быстрое изменение хромосом. В этом случае решения, близкие к оптимальному, не успевают пройти процедуру кроссинговера и закрепить свои свойства в будущих поколениях. Это замедляет или делает полностью невозможным поиск оптимального решения.

Были проведены эксперименты с использованием описанного выше набора исходных данных. Цель:

определить скорость поиска решения при различных параметрах. Скорость определялась по числу итераций алгоритма, прошедших до появления хромосомы с оптимальным решением. Номер итерации соответствует номеру поколения хромосом. Дополнительно анализировали число произошедших мутаций.

В первой части экспериментов постоянным параметром была вероятность мутации, равная 0,3, и переменным параметром — уровни влияния частей целевой функции. Проверяли 10 вариантов отношения уровней влияния второй части целевой функции к первой, начиная от 0,5 (первая влияет в 2 раза сильнее второй) до 1 (уровни влияния одинаковы).

Во второй части экспериментов постоянный параметр — отношение уровней влияния — равен 0,75. Переменный параметр — вероятность мутации — изменялся в диапазоне от 0,05 до 0,5. С данной вероятностью каждая хромосома в поколении подвергается мутации одного гена.

В работе с генетическими алгоритмами слишком велик фактор случайности, и, чтобы уменьшить непредсказуемость результатов, эксперимент с каждым вариантом проводили трижды (табл. 3, 4).

Делаем следующий вывод: максимальная скорость поиска решения получена при значении от-

Таблица 3

Зависимость скорости поиска оптимального решения от отношения уровней влияния частей целевой функции

№	Отношение уровня влияния 2-й части целевой функции к 1-й	Вероятность мутации	Номер поколения с оптимальным решением	Число произошедших мутаций
1	0,5	0,3	8 955	79 208
2	0,5	0,3	13 180	116 532
3	0,5	0,3	12 536	110 620
4	0,6	0,3	4 017	37 003
5	0,6	0,3	5 896	53 940
6	0,6	0,3	6 692	59 474
7	0,7	0,3	2 913	26 785
8	0,7	0,3	3 462	31 596
9	0,7	0,3	4 852	43 596
10	0,75	0,3	2 632	24 783
11	0,75	0,3	3 314	29 518
12	0,75	0,3	2 586	23 024
13	0,8	0,3	3 493	31 969
14	0,8	0,3	2 184	20 868
15	0,8	0,3	2 898	26 991
16	0,85	0,3	3 623	32 275
17	0,85	0,3	2 454	22 563
18	0,85	0,3	3 450	31 619
19	0,9	0,3	5 360	47 210
20	0,9	0,3	3 292	30 222
21	0,9	0,3	2 793	25 514
22	0,95	0,3	5 654	47 670
23	0,95	0,3	7 344	63 477
24	0,95	0,3	7 956	69 446
25	0,97	0,3	9 480	83 922
26	0,97	0,3	11 706	103 320
27	0,97	0,3	11 505	101 910
28	1	0,3	16 663	145 513
29	1	0,3	14 534	126 493
30	1	0,3	13 015	115 193

Таблица 4

Зависимость скорости поиска оптимального решения от вероятности мутации

№	Отношение уровня влияния 2-й части целевой функции к 1-й	Вероятность мутации	Номер поколения с оптимальным решением	Число произошедших мутаций
1	0,75	0,05	10 401	16 047
2	0,75	0,05	10 942	16 646
3	0,75	0,05	7 911	12 097
4	0,75	0,1	5 183	15 214
5	0,75	0,1	6 276	19 417
6	0,75	0,1	5 187	15 794
7	0,75	0,15	4 402	20 192
8	0,75	0,15	3 851	17 985
9	0,75	0,15	4 391	20 196
10	0,75	0,2	4 270	25 332
11	0,75	0,2	4 314	25 659
12	0,75	0,2	6 217	36 262
13	0,75	0,25	2 447	19 360
14	0,75	0,25	3 888	29 448
15	0,75	0,25	3 219	25 601
16	0,75	0,3	2 632	24 783
17	0,75	0,3	3 281	30 287
18	0,75	0,3	2 345	22 017
19	0,75	0,35	2 677	28 779
20	0,75	0,35	3 046	31 642
21	0,75	0,35	3 132	33 249
22	0,75	0,4	2 919	39 981
23	0,75	0,4	5 999	80 892
24	0,75	0,4	4 411	53 757
25	0,75	0,45	5 975	71 911
26	0,75	0,45	3 940	54 151
27	0,75	0,45	9 002	108 710
28	0,75	0,5	19 940	298 566
29	0,75	0,5	17 502	261 759
30	0,75	0,5	9 377	141 192

ношения уровня влияния второй части целевой функции к первой в диапазоне $0,75 \dots 0,9$. То есть часть функции, отвечающая за соблюдение защищенности КЭ, должна влиять сильнее, чем часть функции, минимизирующая число ТК на объекте на $10 \dots 25$ %. Наилучший вариант, когда влияние второй части функции составляет $0,75$ от влияния первой. В пределах погрешности, вызываемой случайными процессами, вариант отношения $0,9$ также показывает высокие результаты.

Как и предсказывалось, равные уровни влияния значительно замедляют работу алгоритма (практически в 3 раза). На рис. 3 показано начало поиска решений для модельного объекта с отношением уровней влияния частей целевой функции, равным единице. Графики отображают суммарное несоответствие хромосом частям целевой функции: нижний график — недостаток ТК для защиты КЭ, верхний — суммарное число ТК. Видно, что несоответствия хромосом частям функции становятся практически одинаковыми, что замедляет приближение к оптимальному решению, так как в оптимальном решении значение первой части функции должно быть нулевым, а от второй части — намного больше нуля [5]. Близкие к единице отношения $0,95$ и $0,97$ также требуют больше итераций алгоритма для поиска решения.

Отношения, меньшие $0,75$, показывают замедление работы: чем больше разница между уровнями влияния частей функции, тем дольше идет поиск решения. Это объясняется следующим: хромосомы, оптимальные по первой части целевой функции (отклонение равно нулю), при различных отклонениях по второй части целевой функции получают примерно равные значения вероятности попадания в родительский пул (так как влияние второй части функции слишком мало).

Вероятность попадания оптимального решения среди хромосом в родительский пул получается низкой, и поиск решения замедляется. На рис. 4 и 5 видно, что скорость сходимости по второй части целевой функции (верхний график) меньше при отношении $0,5$, чем при $0,75$.

Вывод по вероятности мутации: максимальная скорость работы алгоритма получена при значениях $0,25 \dots 0,35$. Меньшие вероятности показывают приблизительно линейную зависимость — чем ниже вероятность мутации, тем больше требуется итераций алгоритма для поиска решения. Это, очевидно, объясняется уменьшением вероятности появления хромосомы, которая будет ближе других к оптимальному решению. На графиках можно увидеть, что вероятность, равная $0,05$, вызывает замедление поиска решения задачи (рис. 6). Обнаруживается низкая сходимости по обеим частям целевой функции при сравнении с вероятностью мутации $0,3$ при том же отношении уровней влияния частей целевой функции (см. рис. 4).

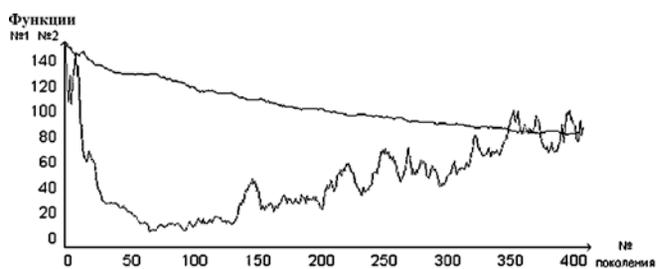


Рис. 3. Графики суммарного несоответствия частям целевой функции (вероятность мутации $0,3$, отношение уровней влияния 1)

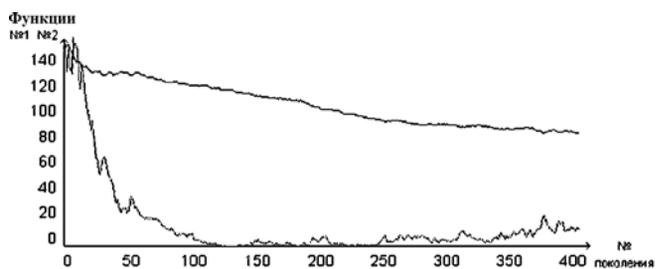


Рис. 4. Графики суммарного несоответствия частям целевой функции (вероятность мутации $0,3$, отношение уровней влияния $0,75$)

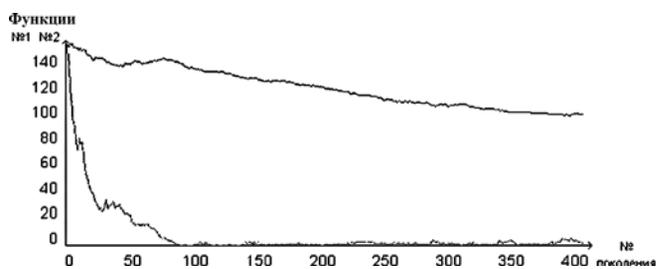


Рис. 5. Графики суммарного несоответствия частям целевой функции (вероятность мутации $0,3$, отношение уровней влияния $0,5$)

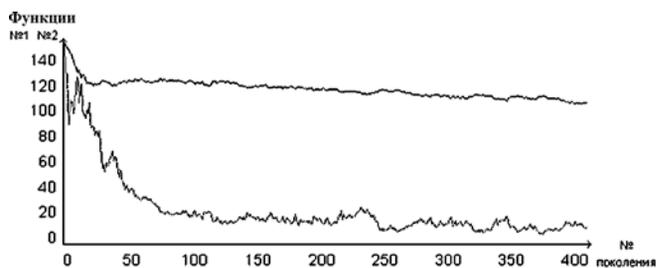


Рис. 6. Графики суммарного несоответствия частям целевой функции (вероятность мутации $0,05$, отношение уровней влияния $0,75$)

Слишком большое число мутаций разрушает хромосомы с эффективными решениями раньше, чем кроссинговер сохранит их свойства в следующих поколениях. Замедление поиска решения при вероятности $0,5$ можно обнаружить в момент приближения свойств хромосом к оптимальному решению (рис. 7). Графики показывают в среднем большее суммарное несоответствие первой части целевой функции (нижний график) по сравнению с вероятностью мутации $0,3$ при том же отношении уровней

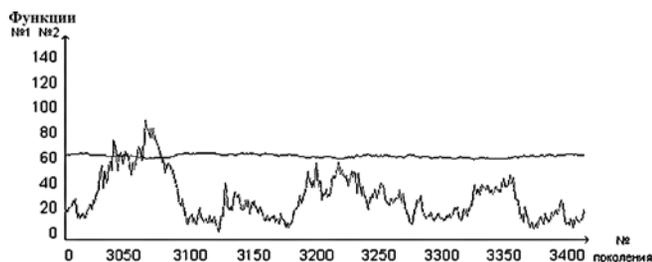


Рис. 7. Графики суммарного несоответствия частям целевой функции при приближении к оптимальному решению (вероятность мутации 0,5, отношение уровней влияния 0,75)

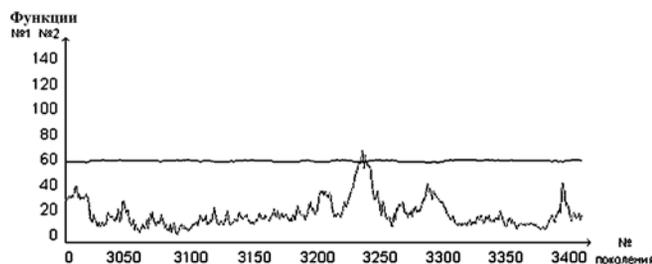


Рис. 8. Графики суммарного несоответствия частям целевой функции при приближении к оптимальному решению (вероятность мутации 0,3, отношение уровней влияния 0,75)

влияния частей целевой функции (рис. 8). При этом в начале работы ГА высокая вероятность мутации увеличивает скорость сходимости.

Делаем вывод, что вероятность мутации необходимо менять при приближении к оптимальному решению: чем дольше работает алгоритм, тем ниже должна быть вероятность.

Заключение

Использование ГА для решения задачи выбора состава СФЗ критически важных объектов требует тщательной настройки параметров алгоритма, так как это влияет на эффективность работы и оптимальность полученных решений. Выбор наилучших значений параметров, подходящих для любых исходных данных, требует постоянного проведения

экспериментов с доказательством оптимальности полученных решений. Кроме того, практически невозможно определить, как правильно должны изменяться параметры во время работы алгоритма для различных исходных данных. Следовательно, необходимо включить в описанный алгоритм механизм адаптации для регулирования значений наиболее важных параметров.

Был проведен анализ процесса решения задачи выбора состава ИТСО с помощью стандартного ГА. В приведенном примере были определены два наиболее важных параметра: уровни влияния частей целевой функции и вероятность мутации. Эксперименты показали ощутимое воздействие значений этих параметров на работоспособность алгоритма.

Определена цель дальнейших исследований: необходимо провести модификацию программы для добавления механизма вычисления значений двух указанных параметров во время работы ГА. Полученный адаптивный генетический алгоритм будет решать задачу выбора состава ИТСО с высокой эффективностью.

Список литературы

1. Магауенов Р. Г. Системы охранной сигнализации: основы теории и принципы построения. М.: Горячая линия — Телеком, 2004. 367 с.
2. Перминов А. Н. О новом подходе к решению задачи мониторинга критически важных объектов и (или) потенциально опасных объектов инфраструктуры Российской Федерации и опасных грузов: цели, задачи и направления развития // Право и безопасность. 2005. № 4. С. 41–46.
3. Рутковская Д., Пилиньский М., Рутковский Л. Нейронные сети, генетические алгоритмы и нечеткие системы: Пер. с польск. И. Д. Рудинского. М.: Горячая линия — Телеком, 2006. 452 с.
4. Холланд Дж. Генетические алгоритмы // В мире науки. 1992. № 9—10. С. 32.
5. Боровский А. С. Обобщенная модель системы физической защиты как объект автоматизированного проектирования // Вестник компьютерных и информационных технологий. 2014. № 10. С. 45–52.
6. Скурихин А. Генетические алгоритмы // Новости искусственного интеллекта. 1995. № 4. С. 6–17.
7. Свидетельство о государственной регистрации программы для ЭВМ № 2014615742 "GenalgSfz". Авторы: Тарасов Андрей Дмитриевич, Боровский Александр Сергеевич. Заявка № 2014613355. Дата поступления 15 апреля 2014 г. Зарегистрировано в Реестре программ для ЭВМ 02 июня 2014 г.

A. D. Tarasov, Senior Lecturer of "Automated systems of information processing and control" department, e-mail: adtarasov@mail.ru, State agrarian university of Orenburg

Genetic Algorithm Effectiveness Used in Problem of Physical Defense System Designing

Is considered the problem of decision-making for potentially-dangerous objects physical defense system structure. The problem stage with experts knowledge using — the conceptual designing, which including object defense analysis, practical recommendations for physical defense system creation, options of engineering defense units structure. It is observed problem solving process of engineering defense units structure designing with Standard Genetic Algorithm using. Method based on object of defense representation in graph form. The example of engineering defense units structure searching with the program is shown. Algorithm parameters with the greatest influence on decision process were found. Experiments are described which show: these parameters must be determined with Adaptive Genetic Algorithm using.

Keywords: physical protection system, decision support system, standard genetic algorithm, adaptive genetic algorithm.

References

1. **Magauenov R. G.** *Sistemy okhrannoi signalizatsii: osnovy teorii i principy postroeniya* (Guard Signalization Systems: Base of Theory and Build Principles), Moscow, Goryachaya liniya — Telekom, 2004, 367 p. (in Russian).
2. **Perminov A. N.** *Pravo i bezopasnost'*, 2005, no. 4, pp. 41—46 (in Russian).
3. **Rutkovskaya D., Pilin'skij M., Rutkovskij L.** *Neironnye seti, geneticheskie algoritmy i nechtotkie sistemi* (Neuron Networks, Genetic

Algorithms and Fuzzy Systems), Moscow, Goryachaya liniya — Telekom, 2006, 452 p. (in Russian).

4. **Holland G.** *V mire nauki*, 1992, no. 9—10, pp. 32 (in Russian).
5. **Borovskij A. S.** *Vestnik komp'uternikh i informatsionnykh tekhnologiy*, 2014, no. 10, pp. 45—52 (in Russian).
6. **Skurikhin A.** *Novosti iskusstvennogo intellekta*, 1995, no. 4, pp. 6—17 (in Russian).
7. **Tarasov A. D., Borovskij A. S.** *Svidetelstvo o gosudarstvennoy registratsii programmy dlya EVM № 2014615742 "GenalgSfz"*, 02.06.2014. (in Russian).

УДК 519.17, 519.8

А. И. Николаев, стажер-исследователь, ainikolaev@hse.ru,

Лаборатория алгоритмов и технологий анализа сетевых структур,

Национальный исследовательский университет Высшая школа экономики, Нижний Новгород

Эффективный подход на основе машинного обучения к решению задачи о максимальной клике

Представлен новый подход к решению задачи о максимальной клике. Предложенный подход состоит в том, что для данного графа с помощью машинного обучения выбирается наиболее быстрый алгоритм из нескольких алгоритмов, решающих задачу о максимальной клике. После чего выбранный алгоритм применяется для решения задачи о максимальной клике в этом графе. Вычислительные эксперименты на графах библиотеки DIMACS показывают, что представленный подход позволяет с высокой точностью выбрать наиболее быстрый алгоритм из нескольких рассматриваемых.

Ключевые слова: задача о максимальной клике, точные алгоритмы, деревья классификации

Введение

Задача о максимальной клике в простом неориентированном графе $G = (V, E)$ является классической задачей теории графов. Клик (или полным подграфом) графа G называется такое подмножество его вершин, в котором любые две вершины соединены ребром. Клика, которая не содержится в клике большего размера, называется *максимальной по включению*. Задача о максимальной клике (ЗМК) состоит в том, чтобы для заданного графа G найти клику максимального размера.

ЗМК является NP -трудной задачей [4]. Данная задача имеет большое число приложений. В частности, эта задача возникает в теории кодирования [8], биоинформатике [2], анализе социальных сетей, анализе сетей фондовых рынков [1].

Для решения ЗМК в литературе предложено большое число как точных алгоритмов, так и приближенных (эвристик). Подробный обзор точных и приближенных алгоритмов для ЗМК может быть найден в работе [10]. В соответствии с результатами, опубликованными в этом обзоре, алгоритмы MaxCLQ [5] и MCS [9] являются наиболее эффективными современными алгоритмами для точного решения ЗМК.

Так как ЗМК относится к классу NP -трудных задач, то теоретический анализ точных алгоритмов

для решения ЗМК затруднителен. Вместо этого проводят эмпирические исследования их характеристик на тестовых графах. Например, для тестирования алгоритмов, решающих ЗМК, используют графы библиотеки DIMACS [12]. При этом во многих работах, посвященных данной задаче, делается вывод о том, что новый алгоритм "лучше" других, если он затрачивает на решение всех тестовых графов меньше времени, чем другие алгоритмы. Хотя тот факт, что одни алгоритмы оказываются лучше других алгоритмов на конкретных графах, зачастую никак не учитывается.

Идеальным решением данной проблемы является обращение к некоему "оракулу", который знает, сколько времени тратит каждый из известных алгоритмов на решение данного примера. После чего оракул выбирает алгоритм, который решает рассматриваемую задачу за минимальное время. К сожалению, совершенного оракула, который за короткое время выбирает наиболее быстрый алгоритм, для таких задач не существует, так как задача определения алгоритма является еще более сложной с вычислительной точки зрения. Поэтому на практике мы можем рассматривать только эвристические алгоритмы для предсказания наиболее быстрого алгоритма из предложенного множества алгоритмов.

В настоящей работе представлен подход, который использует машинное обучение для выбора наиболее быстрого алгоритма из нескольких точных алгоритмов для ЗМК, а именно алгоритма MaxCLQ [5], алгоритма MCS [9] и нескольких вариантов алгоритма RPC [7]. С этой целью для ЗМК были выделены признаки, которые используют при работе метода машинного обучения. Для предложенного подхода проведены вычислительные эксперименты, показывающие его эффективность.

1. Задача о максимальной клике

Задача о максимальной клике в графе $G = (V, E)$, где $|V| = n$, может быть сформулирована как задача булева линейного программирования (БЛП):

$$\max \sum_{i=1}^n x_i; \quad (1)$$

$$x_i + x_j \leq 1, \forall (i, j) \in \bar{E} = \{(i, j) | i, j \in V, i \neq j \wedge (i, j) \notin E\}; \quad (2)$$

$$x_i \in \{0, 1\}, \forall i = 1, \dots, n. \quad (3)$$

В данной модели если вершина i входит в максимальную клику, то переменная x_i равна 1, в противном случае x_i равна 0. Неравенство (2) гарантирует, что только смежные вершины могут входить в искомое решение.

Так как ЗМК может быть сформулирована как задача БЛП, то ее можно решить с помощью общих методов решения задач БЛП. Однако на практике оказывается, что алгоритмы, специально разработанные для решения ЗМК, требуют гораздо меньше времени на поиск точного решения. Поэтому далее будут рассмотрены три современных алгоритма, разработанных для решения ЗМК.

Алгоритм MCS. Алгоритм MCS был предложен в работе [9]. Этот алгоритм реализует метод ветвей и границ. В качестве верхней границы используется *вершинная раскраска* рассматриваемого графа. Вершинной раскраской графа $G = (V, E)$ называется такое отображение $c : V \rightarrow N$, что $c(v) \neq c(w)$ для любых двух смежных вершин $v, w \in V$. Множество вершин, покрашенных в i -й цвет, будем обозначать C_i . Следующее утверждение связывает число вершин в максимальной клике графа G , которое называется *кликковым числом* $\omega(G)$, с числом цветов в вершинной раскраске графа G .

Утверждение 1. Если граф $G = (V, E)$ может быть раскрашен в k цветов, то $\omega(G) \leq k$, причем в максимальную клику входит не более одной вершины из каждого цвета.

Отметим, что задача нахождения раскраски с минимальным числом цветов является NP-трудной задачей. Поэтому на практике для нахождения вершинной раскраски графа часто используют при-

ближенные алгоритмы, в частности, "жадный" алгоритм. "Жадный" алгоритм раскраски работает следующим образом: алгоритм последовательно красит каждую из вершин, учитывая выбранный порядок вершин, в допустимый цвет с минимальным номером. Для повышения качества "жадной" раскраски в алгоритме MCS используется "перекрашивание" [9].

Алгоритм RPC. Алгоритм RPC (*Reusing Parent Coloring*) был предложен в работе [7]. Алгоритм RPC является параметрическим алгоритмом с целым неотрицательным параметром δ . Алгоритм RPC основан на алгоритме MCS [9], и в нем также для вычисления верхней границы используется "жадная" раскраска с перекрашиванием. Основная идея алгоритма RPC заключается в повторном использовании "родительской" раскраски для некоторых подзадач в дереве поиска вместо трудоемкого вычисления "жадной" раскраски с перекрашиванием для каждой подзадачи. Чем больше значение параметра δ , тем чаще повторно используется раскраска, полученная в родительском узле дерева поиска [7]. Отметим, что при $\delta = 0$ алгоритм RPC совпадает с алгоритмом MCS, так как для всех подзадач для вычисления верхней границы используется "жадная" раскраска с перекрашиванием.

Алгоритм MaxCLQ. Алгоритм MaxCLQ впервые был описан в работе [5]. Данный алгоритм сводит ЗМК к задаче максимальной выполнимости булевой формулы в конъюнктивной нормальной форме. Пусть x_1, x_2, \dots, x_n — булевы переменные, принимающие значения 0 (ложь) и 1 (истина). Задача максимальной выполнимости булевой формулы состоит в том, что для данной формулы $F = S_1 \wedge S_2 \wedge \dots \wedge S_k$ (где $S_j = \bigvee_{i \in R_j} x_i$, $R_j \subseteq \{1, 2, \dots, m\}$, $\forall j = \overline{1, k}$), представленной в конъюнктивной нормальной форме, необходимо определить значения булевых переменных x_i ($i = \overline{1, m}$), при которых максимальное число дизъюнктов S_j будет выполняться (принимать значение 1).

Для сведения ЗМК к задаче максимальной выполнимости алгоритм MaxCLQ каждой вершине i графа G ставит в соответствие булеву переменную x_i . Каждой паре несмежных вершин i, j алгоритм ставит в соответствие дизъюнкт $H_r = \bar{x}_i \vee \bar{x}_j$. Вершинная раскраска $\{C_1, C_2, \dots, C_k\}$, полученная с помощью "жадной" раскраски всего графа G , преобразуется в множество дизъюнктов $\{S_1, S_2, \dots, S_k\}$ так, что переменная x_i входит в j -й дизъюнкт S_j ($j = \overline{1, k}$) тогда и только тогда, когда $i \in C_j$. Задача о максимальной клике в графе эквивалента задаче максимальной выполнимости булевой формулы $F = S_1 \wedge S_2 \wedge \dots \wedge S_k$ с дополнительным ограничением, что все дизъюнкты H_r должны выполняться [6]. В такой постановке значение $x_i = 1$ означает, что вершина i входит в максимальную клику.

2. Новый подход к решению задачи о максимальной клике

2.1 Описание подхода

Предложенный подход к решению задачи о максимальной клике выполняется следующим образом. Сначала для заданного графа G выбирают один из алгоритмов MCS, MaxCLQ и нескольких вариантов алгоритма RPC, определяемых значением параметра δ . Необходимо выбрать тот алгоритм, который с наибольшей вероятностью будет самым быстрым для данного графа. После этого выбранный алгоритм применяют для решения ЗМК для графа G .

Задача выбора (предсказания) наиболее быстрого алгоритма из нескольких была рассмотрена как классическая задача машинного обучения — задача классификации. Для этого были выделены признаки, которые используются при работе метода машинного обучения, сгенерирована обучающая выборка и выбран определенный метод классификации.

2.2 Выделенные признаки для входных данных

В соответствии с результатами, опубликованными в работах по точным алгоритмам для решения ЗМК [5, 7—10], можно сделать вывод о том, что наиболее важными характеристиками графов, влияющими на быстродействие алгоритмов ветвей и границ, являются такие параметры, как размеры и плотность графа, степени вершин и их соседей, нижняя и верхняя границы на размер максимальной клики, ширина дерева поиска. Поэтому для выбора наиболее быстрого точного алгоритма для ЗМК из множества алгоритмов были выделены следующие признаки.

1. Число вершин.
2. Число ребер.
3. Плотность графа.
4. Минимальная степень вершин.
5. Максимальная степень вершин.
6. Среднее значение степени вершины.
7. Среднее квадратичное отклонение степени вершины.
8. Минимальная сумма степеней смежных вершин.
9. Максимальная сумма степеней смежных вершин.
10. Среднее значение суммы степеней смежных вершин.
11. Среднее квадратичное отклонение суммы степеней смежных вершин.
12. Размер клики, найденной "жадной" эвристикой (нижняя граница).
13. Число цветов в "жадной" раскраске графа (верхняя граница).
14. Число вершин, которые нужно рассмотреть на первом уровне дерева поиска (ширина дерева поиска).
15. Доля вершин в графе, которые нужно рассмотреть на первом уровне дерева поиска (относительная ширина дерева поиска).

Признаки 1—3 являются классическими признаками для различных задач на графах. Признаки 4—7 описывают характеристики степеней вершин графа, а признаки 8—11 описывают характеристики суммы степеней смежных вершин. Признак 12 дает нижнюю границу на размер максимальной клики в графе. Для нахождения нижней границы используют два "жадных" алгоритма. Первый "жадный" алгоритм сначала добавляет вершину v_1 с максимальной степенью в текущую клику Q . После этого из вершин, смежных вершине v_1 , выбирается вершина v_2 с максимальной степенью в подграфе, образованном этими вершинами, и добавляется в Q . Затем в Q добавляется вершина v_3 с максимальной степенью среди соседей вершин v_1 , v_2 и так далее. Алгоритм заканчивает свою работу, когда клика Q является максимальной по включению.

Второй "жадный" алгоритм из всего графа G удаляет вершину с наименьшей степенью и пересчитывает степени всех вершин в новом графе G' . Далее удаляется вершина с наименьшей степенью в графе G' и так далее. Вершины в графе удаляются до тех пор, пока новый граф не станет кликой. Значение признака 12 равно максимуму из размеров клик, полученных двумя эвристиками. Признак 13 дает верхнюю границу на максимальную клику, опираясь на утверждение 1. Вычисление этой верхней границы выполняется с помощью той же "жадной" эвристики для раскраски графа, которая используется в алгоритмах MCS и RPC. Признак 14 показывает число вершин, которые нужно рассмотреть на первом уровне дерева поиска в алгоритмах MCS и RPC при размере текущей максимальной клики $|Q|$, равном значению признака 12. Это все вершины, которые имеют номер цвета в начальной "жадной" раскраске больше $|Q|$, а значит, потенциально могут входить в клику, которая больше, чем Q . Признак 15 показывает долю таких вершин в графе.

2.3 Обучающая выборка

В качестве обучающей выборки использовали графы, сгенерированные в соответствии с моделью Эрдеша—Реньи [3]. Для генерации графов необходимо задать два параметра n и p , где n — число вершин в генерируемом графе, а p — вероятность того, что любые две вершины i и j будут соединены ребром независимо от всех остальных пар вершин. В табл. 1 представлены значения параметров сгенерированных графов. Вероятности p задавались с шагом 0,05. Для каждой пары значений n и p было сгенерировано 10 графов.

Алгоритмы MCS, RPC и MaxCLQ были реализованы на языке C++ и запущены на 950 сгенерированных графах. Реализация алгоритма MaxCLQ была любезно предоставлена его авторами. Все вычисления проводили на персональном компьютере со следующими характеристиками: процессор Intel Core i5 2,3 ГГц, размер оперативной памяти 8 Гбайт.

Таблица 1

Параметры сгенерированных графов

n	p
150	0,05—0,95
200	0,05—0,95
300	0,05—0,85
400	0,05—0,6
500	0,05—0,6
1000	0,05—0,4
1500	0,05—0,4

Время работы каждого алгоритма было ограничено 600 с. Как результат, только 947 графов были решены за отведенное время. Стоит отметить, что алгоритм RPC запускался с различными параметрами δ начиная с $\delta = 1$. После того как алгоритм RPC закончил свою работу с параметром $\delta = 1$, запускался алгоритм RPC с параметром $\delta = 2$ и так далее. В случае если алгоритм RPC с новым значением параметра $\delta = k$ тратил больше времени, чем алгоритм RPC с параметром $\delta = k - 1$, то алгоритм останавливался и увеличение параметра δ прекращалось. Таким образом, в результате решения ЗМК для всех сгенерированных графов было получено, что $1 \leq \delta \leq 3$.

2.4. Метод классификации

Наиболее быстрый алгоритм выбирали из следующих алгоритмов: MaxCLQ, MCS и RPC со значениями параметра $\delta = 1, 2, 3$. На рисунке (см. третью сторону обложки) представлено распределение графов из обучающей выборки на классы. Здесь 1 — это алгоритм MaxCLQ, который оказался наиболее быстрым для 83 графов; 2 — алгоритм MCS (наиболее быстрый для 320 графов); 3 — алгоритм RPC с параметром $\delta = 1$ (348 графов); 4 — RPC с $\delta = 2$ (188 графов); 5 — RPC с $\delta = 3$ (8 графов). В случае произвольного графа (после вычисления всех его признаков) задачу определения наиболее быстрого алгоритма решения ЗМК для этого графа можно рассматривать как задачу классификации, где номер класса совпадает с номером наиболее быстрого алгоритма.

Задача классификации является известной и широко изучаемой задачей машинного обучения. Для ее решения существует большое число методов, например, метод опорных векторов, байесовский классификатор, нейронная сеть и др. В данной работе был использован метод "деревья классификации" [6], так как этот метод быстро работает и возвращает результаты, которые легко интерпретировать и анализировать. Также несомненным достоинством этого метода является отбор информативных признаков при обучении классификатора. Стоит отметить, что деревья классификации используются для бинарной классификации, но могут быть применены для многоклассовой классификации. Так,

в данной работе использована стратегия "один против всех" для многоклассовой классификации. Стратегия "один против всех" состоит в том, что вместо одного классификатора обучается M классификаторов, где M — общее число классов. Классификатор i (в нашем случае i -е дерево классификации) отделяет объекты i -го класса от объектов остальных классов. Результатом работы дерева классификации является вероятность $P(i | x)$ того, что объект x относится к i -му классу. Затем для объекта x выбирается класс j^* с максимальной вероятностью: $j^* = \arg \max_{j = \overline{1, M}} P(j | x)$.

В нашем случае M равно 5, поэтому с помощью программного обеспечения RapidMiner [13] было обучено пять классификаторов. Стоит отметить, что полученные в результате обучения деревья классификации использовали не все признаки, а только следующие: 2, 3, 6, 8, 10, 12, 13, 14. Это означает, что остальные признаки либо оказались несущественными, либо зависимыми от выбранных признаков. Таким образом, вычисление этих признаков не требуется.

3. Вычислительные результаты

Предложенный алгоритм был обучен на случайных графах, описанных в подразд. 2.2, и протестирован на графах библиотеки DIMACS.

Из библиотеки DIMACS было рассмотрено 25 графов (см. столбец 1 в табл. 3), для которых рассматриваемые алгоритмы решают ЗМК за время, превышающее 1 с (чтобы исключить погрешности компьютерных вычислений) и не превышающее 2 ч (чтобы сократить общее время вычислений). Стоит отметить, что для вычисления признаков, описанных в подразд. 2.1, для любого из графов DIMACS было потрачено не более 20 мс. Результаты работы метода предсказания наиболее быстрого алгоритма для графов библиотеки DIMACS представлены в табл. 2. Для 13 графов наиболее быстрый — алгоритм 4, предсказанный верно для всех 13 графов. Для восьми графов наиболее быстрым является алгоритм 5, но был предсказан алгоритм 4, который, тем не менее, дает близкие результаты и для всех восьми графов является вторым по скорости вы-

Таблица 2

Таблица сопряженности классификатора для графов DIMACS

		Реальный класс					Точность, %
		1	3	4	5	2	
Предсказанный класс	1	2	1	0	0	0	66,67
	3	0	1	0	0	0	100,00
	4	0	0	13	8	0	61,90
	5	0	0	0	0	0	0,00
	2	0	0	0	0	0	0,00
	Полнота, %	100,00	50,00	100,00	0,00	0,00	

Время решения ЗМК для графов библиотеки DIMACS, мс

Граф DIMACS	MaxCLQ	MCS	RPC, $\delta = 1$	RPC, $\delta = 2$	RPC, $\delta = 3$	Наиболее быстрый алгоритм	Предложенный подход
C250.9	344516	1361335	1041168	987836	971229	344516	344517
MANN_a45	34148	43230	31159	34574	85558	31159	34154
brock400_1	259708	284586	244209	234715	235103	234715	234716
brock400_2	118908	125398	107746	103844	103952	103844	103845
brock400_3	204222	193386	164518	158995	159121	158995	158996
brock400_4	130754	92893	79692	76856	77071	76856	76858
brock800_1	5606592	3914626	3430660	3294158	3300223	3294158	3294165
brock800_2	4889039	3395345	2971610	2845765	2842208	2842208	2845772
brock800_3	3222601	3461488	3020965	2899545	2898460	2898460	2899552
brock800_4	2438408	1602096	1405760	1341436	1333342	1333342	1341442
dsjc500.5	3532	1555	1412	1426	1487	1412	1415
dsjc1000.5	317877	140509	127070	123663	127732	123663	123675
frb30-15-1	655244	721617	440375	339713	289979	289979	339715
frb30-15-2	951654	533285	296789	208696	160926	160926	208698
frb30-15-3	580959	473354	278528	210579	176525	176525	210581
frb30-15-4	1155555	1327562	765939	574914	481628	481628	574915
frb30-15-5	873662	489109	266500	187442	147959	147959	187443
p_hat300-3	1387	1245	1051	1044	1089	1044	1045
p_hat500-3	49829	60698	51642	49310	49779	49310	49313
p_hat700-2	3586	2726	2319	2250	2371	2250	2255
p_hat700-3	1082242	1063763	894519	841329	854369	841329	841333
p_hat1000-2	117828	106392	87986	82428	83927	82428	82439
p_hat1500-1	11408	2257	1993	1916	1922	1916	1936
sanr200_0.9	5604	13855	10712	10107	10122	5604	5604
sanr400_0.7	97663	79299	68261	66440	68368	66440	66442
Общее время	23156926	19491609	15792583	14678981	14464450	13750666	14030826

числений (табл. 3). Для двух графов наилучший алгоритм — алгоритм 1, и он был предсказан верно в обоих случаях. И еще для двух графов наилучшим является алгоритм 3, вместо которого в одном случае был предсказан алгоритм 1, дающий второй по скорости вычислений результат, близкий к первому. Общая точность предсказания алгоритма равна 64 % (16 из 25).

В табл. 4 показано среднее сокращение времени вычислений с помощью предложенного подхода (подсчет признаков, выбор одного из пяти алгоритмов на основе деревьев классификации, решение ЗМК с помощью выбранного алгоритма) по сравнению с каждым из рассматриваемых алгоритмов в отдельности. Среднее сокращение времени вычислений относительно каждого отдельного алгоритма подсчитывается следующим образом: для каждого

графа вычисляется, насколько время работы предложенного подхода меньше времени работы отдельного алгоритма, а затем вычисляется среднее арифметическое по всем полученным значениям. Сокращение времени вычислений предложенного

подхода определяется по формуле $\frac{t' - t}{t'} \cdot 100\%$, где

t — время работы предложенного подхода, а t' — время работы отдельного алгоритма. Опираясь на табл. 3 и 4, можно сделать вывод о том, что предложенный подход в среднем быстрее алгоритма MaxCLQ и алгоритма MCS на 35 и 28 % соответственно. Если же сравнивать предложенный подход с заранее неизвестным наиболее быстрым для каждого конкретного графа алгоритмом, то наиболее быстрый алгоритм (разный для каждого графа) тратит на решение ЗМК лишь на 4,97 % времени меньше, чем предложенный подход.

Таблица 4

Среднее сокращение времени вычислений (%) предложенного подхода относительно каждого из рассматриваемых алгоритмов в отдельности для графов библиотеки DIMACS

MaxCLQ	MCS	RPC, $\delta = 1$	RPC, $\delta = 2$	RPC, $\delta = 3$	Наиболее быстрый алгоритм	Предложенный подход
35,74	28,97	11,83	4,41	3,21	-4,97	0,00

Заключение

В данной работе представлен новый подход к решению задачи о максимальной клике. Предложенный подход использует деревья классификации для выбора из нескольких рассматриваемых алгоритмов наиболее быстрого алгоритма для заданного графа. После этого выбранный алгоритм применяется для решения задачи о максимальной клике

в данном графе. С этой целью для ЗМК выделены признаки, которые используют при предсказании наиболее быстрого алгоритма. Вычислительные результаты показывают эффективность предложенного подхода. Так, для графов библиотеки DIMACS точность выбора наиболее быстрого алгоритма составляет 64 %, а в тех случаях, когда выбранный алгоритм не является наиболее быстрым, он является вторым по скорости вычислений. При этом наиболее быстрый алгоритм (разный для каждого графа) тратит на решение ЗМК лишь на 5 % меньше, чем предложенный подход.

Работа поддержана грантом РФФ 14-41-00039.

Список литературы

1. **Boginski V., Butenko S., Pardalos P.** Statistical analysis of financial networks // *Computational Statistics and Data Analysis*. 2005. N. 48 (2). P. 431–443
2. **Butenko S., Wilhelm W. E.** Clique-detection models in computational biochemistry and genomics // *European Journal of Operational Research*. 2006. N. 173. P. 1–17.
3. **Erdos P., Renyi A.** On random graphs I // *Publ. Math. Debrecen*. 1959. N. 6. P. 290–297.
4. **Karp R. M.** Reducibility Among Combinatorial Problems // *Complexity of Computer Computations*. 1972. P. 85–103.
5. **Li C. M., Quan Z.** Combining graph structure exploitation and propositional reasoning for the maximum clique problem // *Proceedings of the 2010 22nd IEEE International Conference on Tools with Artificial Intelligence*. 2010. Vol. 1. P. 344–351.
6. **Loh W.-Y.** Classification and regression trees // *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*. 2011. Vol. 1, N. 1. P. 14–23.
7. **Nikolaev A., Batsyn M., San Segundo P.** Reusing the same coloring in the child nodes of the search tree for the maximum clique problem // *Lecture Notes in Computer Science*. 2015. N. 8994. P. 275–280.
8. **Pardalos P., Rebennack S.** Computational challenges with cliques, quasi-cliques and clique partitions in graphs // *Lecture Notes in Computer Science*. 2010. N. 6049. P. 13–22.
9. **Tomita E., Sutani Y., Higashi T., Takahashi S., Wakatsuki M.** A Simple and Faster Branch-and-Bound Algorithm for Finding a Maximum Clique // *Lecture Notes in Computer Science*. 2010. N. 5942. P. 191–203.
10. **Wu Q., Hao J. K.** A review on algorithms for maximum clique problems // *European Journal of Operational Research*. 2015. N. 242. P. 693–709.
11. **Xu L., Hutter F., Hoos H. H., Leyton-Brown K.** SATzilla: portfolio-based algorithm selection for SAT // *Journal of Artificial Intelligence Research*, 2008, vol. 32, pp. 565–606.
12. <http://dimacs.rutgers.edu/Challenges/>
13. <http://rapidminer.com/>

A. I. Nikolaev, Research Intern, ainikolaev@hse.ru

Laboratory of Algorithms and Technologies for Networks Analysis,
National Research University Higher School of Economics, Nizhny Novgorod

Efficient Approach for the Maximum Clique Problem Based on Machine Learning

In this paper a new approach for solving the maximum clique problem is presented. For a given graph the suggested approach uses machine learning technique to predict the fastest algorithm from several algorithms for the maximum clique problem. Then the chosen algorithm is applied for solving the maximum clique problem in this graph. The computational results show the efficiency of the proposed approach.

Keywords: maximum clique problem; exact algorithms; decision trees

References

1. **Boginski V., Butenko S., Pardalos P.** Statistical analysis of financial networks, *Computational Statistics and Data Analysis*, 2005, no. 48 (2), pp. 431–443.
2. **Butenko S., Wilhelm W. E.** Clique-detection models in computational biochemistry and genomics, *European Journal of Operational Research*, 2006, 173, pp. 1–17.
3. **Erdos P., Renyi A.** On random graphs I, *Publ. Math. Debrecen*, 1959, no. 6, pp. 290–297.
4. **Karp R. M.** Reducibility Among Combinatorial Problems, *Complexity of Computer Computations*, 1972, pp. 85–103.
5. **Li C. M., Quan Z.** Combining graph structure exploitation and propositional reasoning for the maximum clique problem, *Proceedings of the 2010 22nd IEEE International Conference on Tools with Artificial Intelligence*, 2010, no. 1, pp. 344–351.
6. **Loh W.-Y.** Classification and regression trees, *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2011, vol. 1, no. 1, pp. 14–23.
7. **Nikolaev A., Batsyn M., San Segundo P.** Reusing the same coloring in the child nodes of the search tree for the maximum clique problem, *Lecture Notes in Computer Science*, 2015, no. 8994, pp. 275–280.
8. **Pardalos P., Rebennack S.** Computational challenges with cliques, quasi-cliques and clique partitions in graphs, *Lecture Notes in Computer Science*, 2010, 6049, pp. 13–22.
9. **Tomita E., Sutani Y., Higashi T., Takahashi S., Wakatsuki M.** A Simple and Faster Branch-and-Bound Algorithm for Finding a Maximum Clique, *Lecture Notes in Computer Science*, 2010, no. 45942, pp. 191–203.
10. **Wu Q., Hao J. K.** A review on algorithms for maximum clique problems, *European Journal of Operational Research*, 2015, no. 242, p. 693–709.
11. **Xu L., Hutter F., Hoos H. H., Leyton-Brown K.** SATzilla: portfolio-based algorithm selection for SAT, *Journal of Artificial Intelligence Research*, 2008, vol. 32, pp. 565–606.
12. <http://dimacs.rutgers.edu/Challenges/>
13. <http://rapidminer.com/>

МОДЕЛИРОВАНИЕ И ОПТИМИЗАЦИЯ MODELING AND OPTIMIZATION

УДК 517.938

А. В. Дмитриев, д-р физ.-мат. наук, проф., e-mail: a.dmitriev@hse.ru,

С. В. Мальцева, д-р техн. наук, проф., e-mail: smaltseva@hse.ru,

О. А. Цуканова, аспирант, e-mail: otsukanova@hse.ru

Национальный исследовательский университет "Высшая школа экономики"

Моделирование и качественный анализ социальной микроблогинговой сети как динамической системы

Представлены результаты качественного анализа социальной сети, базирующейся на принципе микроблогинга. Сетевое взаимодействие аппроксимировалось дифференцируемой нелинейной динамической системой числа отправителей и получателей заметок (без ограничений и с ограничениями). Установлено, что адекватной является динамическая система с учетом насыщения и взаимодействий между отправителями и получателями. Получены параметрические условия асимптотической устойчивости и вилообразной бифуркации для условий сетевой динамики.

Ключевые слова: математическое моделирование, социальные сети, микроблогинг, динамические системы, нелинейная динамика, асимптотическая устойчивость, вилообразная бифуркация

Введение

Математическому моделированию социальных сетей посвящено большое количество публикаций. Наиболее полный обзор таких моделей представлен в работе [1]: оптимизационные и имитационные модели (модели с порогами, модели просачивания и заражения, модели Изинга, модели на основе цепей Маркова, модели на основе клеточных автоматов и др.), теоретико-игровые модели (модели взаимной информативности, модели согласованных коллективных действий, модели стабильности сети и др.). За исключением модели диффузии информации [2], практически отсутствуют модели, основанные на идеях и формализме теории нелинейных дифференцируемых динамических систем [3]. В последнее время подходы данной теории успешно применяют в моделировании динамических систем разной природы.

В настоящей работе исследование ограничено сетями "микроблогингового характера", наиболее известным примером которых является *Twitter*. Мы полагаем, что подобную социальную сеть можно считать такой динамической системой, которая из совокупного действия индивидуальных интересов вырабатывает агрегированные факторы (потoki), которые начинают проявляться в макромасштабе и действовать по законам детерминированных связей и отношений. Можно предположить, что такая сеть с точки зрения возникших в ней обобщенных макроскопических потоков гомеоморфна динами-

ческим системам так называемого гидродинамического типа. Следовательно, если в таких системах возникают взаимодействующие встречные потоки, то в них (т. е. системах), как правило, возникает явление обобщенной турбулентности, порождающее кризисные режимы развития состояний таких динамических систем.

В терминологии популяционной динамики пользователи микроблогинговых сетей находятся в отношении мутуализма, в одном из типов симбиотических отношений, при котором межпопуляционные взаимодействия являются облигантными (обязательными) [4]. В этом случае необходимо существование отправителей и получателей заметок (твитов, в случае *Twitter*) — в отсутствие одного из этих видов другой вид вымирает (нет отправителей — нет получателей, нет получателей — нет смысла в отправителях).

Целью работы является исследование сети как динамической системы с учетом информационного насыщения получателей и ресурсно-информационного взаимодействия отправителей и получателей.

Для достижения данной цели решали следующие задачи:

- определение ненулевого равновесного числа отправителей и получателей сообщений;
- определение параметрических условий существования ненулевого равновесного числа отправителей и получателей.

Динамику системы "отправители—получатели" моделировали системами автономных обыкновенных дифференциальных уравнений первого порядка

$$\begin{cases} \dot{x} = f(x, y) \\ \dot{y} = g(x, y), \end{cases}$$

где $x(t) \geq 0$ — число отправителей заметок; $y(t) \geq 0$ — число получателей заметок.

Для определения равновесных точек системы и анализа их устойчивости проводили качественный анализ динамических систем в первом приближении. Основу такого анализа составляет теорема Ляпунова об устойчивости по первому приближению [5]: если все собственные значения якобиана автономной системы имеют отрицательные действительные части ($\text{Re}(\lambda_i) < 0$), то соответствующая равновесная точка является асимптотически устойчивой. Равновесное состояние является асимптотически устойчивым, если малые отклонения от него со временем затухают [5].

Далее рассматриваются и анализируются динамические модели с определенными уровнями ограниченности.

1. Подход к исследованию динамики социальных сетей

Поскольку сети отличаются большими масштабами и разнородностью, то в целях разработки наиболее адекватных методик для их исследования, моделирования и оптимизации целесообразно условно разделять эти внушительные произвольно действующие динамические конгломераты на уровни: макро- и микро-, полагая при этом, что выводы и показатели, корректные для макроуровня, будут аналогичными и на микроуровне. Отсюда предлагается следующий — поуровневый — подход, основанный на модели текстурированного пространства состояний [5].

Исследуемое пространство условно разделено на два уровня.

Первый уровень представляет собой уровень визуальных образов. Его автоматический мониторинг позволяет выявить возможные отклонения. И в случае выявления таковых осуществляется переход на микроуровень — уровень конкретного слоя текстуры.

Согласно предлагаемой методике на макроуровне ведется работа с визуальными слоями и, соответственно, принимаются решения, затрагивающие уровень макропоказателей сети, например, "конвейерная обработка" визуализированных представлений сети (т. е. слоев текстуры) в режиме реального времени в целях выявления "отклонений" в состоянии сети.

Скорость изменения состояния сети чрезвычайно высока. В связи с масштабами современных сетей постоянный мониторинг усложняется. Это означает, что нужно автоматически определять такие состоя-

ния и точки, к которым, и в какой момент, необходимо применять управляющее воздействие.

Практически на глобальном уровне предлагается проводить выявление существенных изменений на уровне образов — или геометрических отображений структуры сети. Происходит изучение визуальных образов — "срезов" сети в определенные моменты времени ("слоев" в текстурированном пространстве состояний). Через определенный период времени делается "снимок" сети, в этот момент автоматически должен быть сделан вывод о том, произошли значимые изменения или нет. С использованием визуальных образов автоматически решается задача о сходстве изображений. В случае выявления отличий дается сигнал, что изображения не совпадают друг с другом. Делается вывод о наступлении значимого события.

В случае выявления "отклонений" или существенных изменений на первом уровне следует проводить более глубокие исследования сети в целях предметного анализа происходящего. Делается вывод о предполагаемом формировании нового слоя текстуры и происходит переход к микроуровню — изучению конкретного слоя, на котором выявилось отклонение, а также к детальному изучению факторов, вызвавших это изменение.

Ниже предлагается инструментарий исследования на микроуровне, основанный на применении дифференциальных уравнений в сетях, основанных на принципе "микроблогинга", наиболее известным из которых является *Twitter*.

Итак, в сети выявилось существенное изменение. Тогда, например, для случая *Twitter* можно сделать предположение, что это произошло в результате пересылки твитов.

2. Качественный анализ сети. Динамическая система без ограничений

В отсутствие каких-либо ограничений (мальтузианская система) динамика пары "отправитель—получатель" в простейшем случае описывается динамической системой

$$\begin{cases} \dot{x} = -\alpha_1 x + \beta_1 xy; \\ \dot{y} = -\alpha_2 y + \beta_2 xy. \end{cases} \quad (1)$$

Наличие произведения xy в правых частях динамической системы отражает факт асимптотического стремления к нулю числа отправителей и получателей заметок при "вымирании" одного из этих видов. Члены $-\alpha_1 x$ и $-\alpha_2 y$ соответствуют уменьшению числа отправителей и получателей заметок со скоростями α_1 и α_2 соответственно; члены $+\beta_1 xy$ и $+\beta_2 xy$ — увеличению числа отправителей и получателей заметок за счет их взаимодействия со скоростями β_1 и β_2 (в случае *Twitter* может быть приведен следующий пример: отправление ответов от получателей к отправителям твитов с последую-

шим отправлением и получением дополнительных твитов, отправлением ретвитов подписчикам).

Проведем качественный анализ системы (1) в первом приближении.

Система (1) имеет две равновесные точки — нулевую $O(0,0)$ и ненулевую $E\left(\frac{\alpha_2}{\beta_2}, \frac{\alpha_1}{\beta_1}\right)$. Якобиан системы (1)

$$J = \begin{pmatrix} -\alpha_1 + \beta_1 y & \beta_1 x \\ \beta_2 y & -\alpha_2 + \beta_2 x \end{pmatrix}.$$

Собственные значения якобиана в точке $O(0,0)$ ($\lambda_1 = -\alpha_1, \lambda_2 = -\alpha_2$) — отрицательные числа. Следовательно, по теореме Ляпунова об устойчивости в первом приближении нулевая равновесная точка является асимптотически устойчивой. Если $\alpha_1 \neq \alpha_2$, то данное положение равновесия является асимптотически устойчивым узлом. Если $\alpha_1 = \alpha_2$, то данное положение равновесия является асимптотически устойчивым дикритическим узлом. В обоих случаях

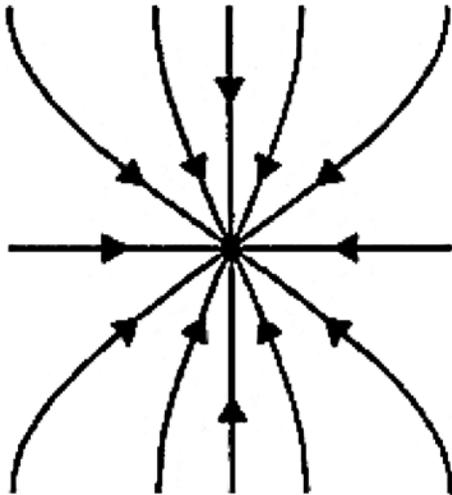


Рис. 1. Асимптотически устойчивый узел

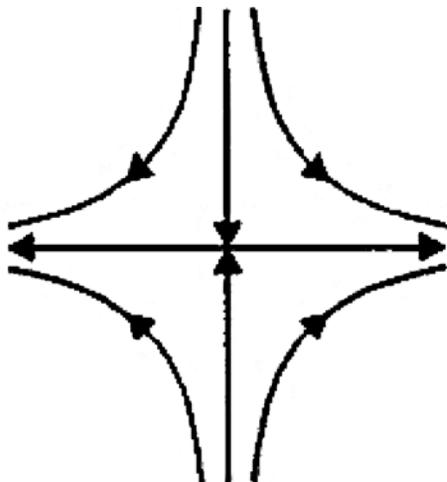


Рис. 2. Неустойчивое седло

число отправителей и получателей заметок асимптотически стремится к нулю, что в пределе соответствовало бы "вымиранию" сети. На рис. 1 представлено схематическое изображение фазовой диаграммы, изображающей асимптотически устойчивый узел (множество траекторий $y = y(x)$ системы при различных начальных условиях). Очевидна сходимость траекторий к равновесной точке, в данном случае к точке O . Асимптотически устойчивый узел является одним из видов аттракторов (траектории притягиваются к равновесной точке) [6].

Собственные значения якобиана в точке $E\left(\frac{\alpha_2}{\beta_2}, \frac{\alpha_1}{\beta_1}\right)$ ($\lambda = \pm \sqrt{\alpha_1 \alpha_2}$) — действительные числа разных знаков. Следовательно, по теореме Ляпунова об устойчивости в первом приближении ненулевая равновесная точка не является устойчивой. Ненулевая равновесная точка является седлом. На рис. 2 представлена фазовая диаграмма, изображающая седловую точку (седло) [7].

Таким образом, не существует положительных значений параметров, при которых система (1) имеет устойчивую ненулевую равновесную точку. Следовательно, модель (1) является неадекватной. Причиной этого является отсутствие ограничений на динамические переменные.

3. Динамическая система с учетом насыщения получателей

Учтем фактор насыщения получателей заметок. Динамическая система (1) при этом принимает следующий вид:

$$\begin{cases} \dot{x} = -\alpha_1 x + \frac{m_x x y}{1 + A_x y}; \\ \dot{y} = -\alpha_2 y + \frac{m_y x y}{1 + A_y x}. \end{cases} \quad (2)$$

В этой системе m_x/A_x и m_y/A_y — максимальные "рационы" отправителей и получателей заметок; $1/A_x$ — число получателей заметок, при котором "рацион" отправителей составляет половину максимального; $1/A_y$ — число отправителей, при котором "рацион" получателей составляет половину максимального. Под максимальным рационом получателей заметок понимается максимальное число полученных заметок до "насыщения" получателей; максимальный рацион отправителей заметок — максимальное число ответов, полученных от получателей.

Система (2) имеет нулевую $O(0, 0)$ и ненулевую $E\left(\frac{\alpha_2}{m_y - \alpha_2 A_y}, \frac{\alpha_1}{m_x - \alpha_1 A_x}\right)$ равновесные точки. Учитывая, что $x(t)$ и $y(t)$ — число отправителей и число получателей коротких сообщений соответственно,

координаты ненулевой точки ограничены следующей системой неравенств:

$$\begin{cases} m_y - \alpha_2 A_y > 0; \\ m_x - \alpha_1 A_x > 0. \end{cases} \quad (3)$$

Якобиан системы (2)

$$J = \begin{pmatrix} -\alpha_1 + \frac{m_x y}{1 + A_x y} & \frac{m_x x}{(1 + A_x y)^2} \\ \frac{m_y y}{(1 + A_y x)^2} & -\alpha_2 + \frac{m_y x}{1 + A_y x} \end{pmatrix}.$$

Собственные значения якобиана в точке $O(0, 0)$ ($\lambda_1 = -\alpha_1, \lambda_2 = -\alpha_2$) — отрицательные числа. В первом приближении нулевая равновесная точка является асимптотически устойчивой. Если $\alpha_1 \neq \alpha_2$, то данное положение равновесия является асимптотически устойчивым узлом (см. рис. 1). Если $\alpha_1 = \alpha_2$, то данное положение равновесия является асимптотически устойчивым дикритическим узлом. Характер нулевой точки системы (2) аналогичен системе (1): в обоих случаях число отправителей и получателей асимптотически стремится к нулю ("вымирание" сети).

Наиболее важный вопрос: существует ли асимптотически устойчивая ненулевая равновесная точка системы (2)? Собственные значения якобиана в ненулевой равновесной точке

$$\lambda = \pm \frac{m_x m_y \alpha_1 \alpha_2}{(1 + A_y x)^2 (1 + A_x y)^2 (m_y - \alpha_2 A_y)(m_x - \alpha_1 A_x)}.$$

Учитывая (3), данные собственные значения — действительные числа разных знаков. Следовательно, ненулевая равновесная точка — седло (см. рис. 2).

Таким образом, учет насыщения отправителей и получателей не меняет характера устойчивости системы. В обеих моделях число отправителей и получателей либо асимптотически стремится к нулю при любых начальных условиях, либо неограниченно возрастает.

4. Динамическая система с учетом насыщения и взаимодействия отправителей и получателей

Учтем фактор взаимодействия между отправителями ($-\gamma_1 x^2$) и получателями ($-\gamma_2 y^2$) твитов. В этом случае динамическая система (2) примет следующий вид:

$$\begin{cases} \dot{x} = -\alpha_1 x - \gamma_1 x^2 + \frac{m_x xy}{1 + A_x y}; \\ \dot{y} = -\alpha_2 y - \gamma_2 y^2 + \frac{m_y xy}{1 + A_y x}. \end{cases} \quad (4)$$

Для упрощения анализа системы (4) введем следующие обозначения:

$$t = \tau/\alpha_1, \quad x = \frac{\alpha_1}{\gamma_1} x_1, \quad y = \frac{\alpha_2}{\gamma_2} x_2.$$

С учетом этих обозначений систему (3) можно представить в следующем виде:

$$\begin{cases} \frac{dx_1}{d\tau} = -x_1 - x_1^2 + \frac{\rho_1 x_1 x_2}{1 + \mu_1 x_2}; \\ \frac{dx_2}{d\tau} = -\varepsilon x_2 - \varepsilon x_2^2 + \frac{\rho_2 x_1 x_2}{1 + \mu_2 x_1}, \end{cases} \quad (5)$$

где $\rho_1 = \frac{m_x \alpha_2}{\alpha_1 \gamma_2}$; $\rho_2 = \frac{m_y \alpha_1}{\alpha_2 \gamma_1}$; $\mu_1 = \frac{A_x \alpha_2}{\gamma_2}$; $\mu_2 = \frac{A_y \alpha_1}{\gamma_1}$; $\varepsilon = \frac{\alpha_2}{\alpha_1}$.

Система (5) имеет нулевую равновесную точку $O(0, 0)$. Якобиан системы (5)

$$J = \begin{pmatrix} 1 - 2x_1 + \frac{\rho_1 x_2}{1 + \mu_1 x_2} & \frac{\rho_1 x_1}{(1 + \mu_1 x_2)^2} \\ \frac{\rho_2 x_2}{(1 + \mu_2 x_1)^2} & -\varepsilon - 2\varepsilon x_2 + \frac{\rho_2 x_1}{1 + \mu_2 x_1} \end{pmatrix}.$$

Собственные значения якобиана в точке $O(0, 0)$ ($\lambda_1 = 1, \lambda_2 = -\varepsilon$) — действительные числа разного знака. Следовательно, точка $O(0, 0)$ является неустойчивым седлом (см. рис. 2). Таким образом, учет взаимодействий между отправителями и взаимодействиями между получателями, по крайней мере, исключает возможность уничтожения сети.

Осталось выяснить факт существования и устойчивость ненулевой равновесной точки системы (4). Ненулевая равновесная точка является решением однородного алгебраического уравнения

$$\begin{cases} 1 + x_1 + x_2(\mu_1 - \rho_1) + \mu_1 x_1 x_2 = 0; \\ \varepsilon(1 + x_2) + x_1(\varepsilon \mu_2 - \rho_2) + \varepsilon \mu_2 x_1 x_2 = 0. \end{cases} \quad (6)$$

Имеем две ненулевые равновесные точки: точку E_1 с координатами

$$x_1^* = \frac{-\sqrt{\delta} - (\mu_1 - \rho_1)(\varepsilon \mu_2 - \rho_2) - \varepsilon \mu_1 + \varepsilon + \mu_2}{2(\varepsilon \mu_1 \mu_2 - \mu_1 \rho_2 - \mu_2)};$$

$$x_2^* = \frac{\sqrt{\delta} + (\mu_1 - \rho_1)(\varepsilon \mu_2 - \rho_2) - \varepsilon \mu_1 - \varepsilon + \mu_2}{2(\varepsilon \mu_1 - \mu_1 \mu_2 + \mu_2 \rho_1)}$$

и точка E_1 с координатами

$$x_1^* = \frac{\sqrt{\delta} - (\mu_1 - \rho_1)(\varepsilon \mu_2 - \rho_2) - \varepsilon \mu_1 + \varepsilon + \mu_2}{2(\varepsilon \mu_1 \mu_2 - \mu_1 \rho_2 - \mu_2)};$$

$$x_2^* = \frac{\sqrt{\delta} + (\mu_1 - \rho_1)(\varepsilon \mu_2 - \rho_2) - \varepsilon \mu_1 - \varepsilon + \mu_2}{2(\varepsilon \mu_1 - \mu_1 \mu_2 + \mu_2 \rho_1)},$$

где $\delta = ((\mu_1 - \rho_1)(\varepsilon\mu_2 - \rho_2) + \varepsilon\mu_1 - \varepsilon - \mu_2)^2 - 4\varepsilon(\mu_1 - \rho_1 - 1)(\varepsilon\mu_1\mu_2 - \mu_1\rho_2 - \mu_2)$, $x_1^* > 0$ и $x_2^* > 0$.

Параметр δ является бифуркационным параметром вилообразной бифуркации системы [7]. Действительно, при $\delta < 0$ существует только нулевая равновесная точка O , при $\delta > 0$ — нулевая и две ненулевые равновесные точки E_1 и E_2 . Бифуркационное значение параметра $\delta_0 = 0$ (скачкообразно появляются ненулевые равновесные количества отправителей и получателей твитов при смене знака δ с отрицательного на положительный).

Вид системы (6) определяет необходимое условие существования ненулевых равновесных точек с положительными координатами:

$$\begin{cases} \mu_1 < \rho_1; \\ \varepsilon\mu_2 < \rho_2. \end{cases} \quad (7)$$

В качестве примера приведем результаты вычислительного эксперимента по определению ненулевых положительных равновесных точек при $\varepsilon = 1$, $\mu_1 = 1$, $\mu_2 = 2$, $\mu_1 - \rho_1 = \varepsilon\mu_2 - \rho_2 = -25$. Данные параметры удовлетворяют системе (6). В этом случае система имеет две ненулевые равновесные точки с положительными координатами $E_1(0.04, 0.04)$ и $E_2(23.03, 12.21)$. Точка E_1 — неустойчивая седловая точка ($\lambda_1 = 2, 29$, $\lambda_2 = -0,45$) (см. рис. 1), E_2 — асимптотически устойчивый узел ($\lambda_1 = -21,23$, $\lambda_2 = -11,98$) (см. рис. 2).

Необходимым и достаточным условием устойчивости равновесных точек является выполнение системы неравенств:

$$\sigma > 0, \Delta > 0.$$

Это следует из теоремы Ляпунова об устойчивости по первому приближению.

Для системы (5)

$$\begin{aligned} \sigma &= -1 + 2x_1^* - \frac{\rho_1 x_2^*}{1 + \mu_1 x_2^*} + \varepsilon + 2\varepsilon x_2^* - \frac{\rho_2 x_1^*}{1 + \mu_2 x_1^*}; \\ \Delta &= \left(2x_1^* - \frac{\rho_1 x_2^*}{1 + \mu_1 x_2^*}\right) \left(\varepsilon + 2\varepsilon x_2^* - \frac{\rho_2 x_1^*}{1 + \mu_2 x_1^*}\right) - \\ &\quad - \frac{\rho_1 x_1^* \rho_2 x_2^*}{(1 + \mu_1 x_2^*)^2 (1 + \mu_2 x_1^*)^2}. \end{aligned}$$

Для системы (5) необходимым и достаточным условием существования устойчивых ненулевых равновесных точек является справедливость неравенства для параметров системы:

$$\begin{cases} k < 0; \\ m < 0; \\ 0 < \rho_1 \rho_2 x_1^* x_2^* < km(1 + \mu_1 x_2^*)^2 (1 + \mu_2 x_1^*)^2, \end{cases} \quad (8)$$

где $k = (1 - 2x_1^*)(1 + \mu_1 x_2^*) + \rho_1 x_2^*$, $m = -(\varepsilon + 2\varepsilon x_2^*) \times (1 + \mu_2 x_1^*) + \rho_2 x_1^*$.

Таким образом, система (2) дает наиболее адекватное описание нелинейной динамики отправителей и получателей коротких сообщений. В сети возможно установление асимптотически устойчивого сосуществования отправителей и получателей, что определяется условиями (7), (8) и $\delta > 0$.

Заключение

В работе сделана попытка решения прикладной задачи актуализации сетевого взаимодействия как диссипативной (представленной в виде нелинейных обыкновенных дифференциальных уравнений) динамической системы с учетом информационного насыщения получателей и ресурсно-информационного взаимодействия отправителей и получателей.

Установлено, что адекватной является динамическая система с учетом насыщения и взаимодействий между отправителями и получателями актуальной сетевой информации. Получены параметрические условия асимптотической устойчивости и вилообразной бифуркации для условий сетевой динамики.

Полученные соотношения, связывающие контролируемые параметры системы (относительные скорости уменьшения и увеличения числа отправителей и получателей заметок, максимальные "рационалы" отправителей и получателей заметок и др.), могут быть использованы для определения качественных условий существования ненулевого конечного числа получателей и отправителей заметок, а также качественных условий устойчивости сети к изменению контролируемых параметров. В этих случаях принципиально важными являются не абсолютные значения контролируемых параметров, которые в некоторых случаях не могут быть оценены, а соотношение между ними в виде неравенств.

На наш взгляд, наиболее перспективным усовершенствованием модели является использование моделей гидродинамического типа, основой которых являются соотношения Л. Онзагера [7]. Эти соотношения связывают потоки и порождающие их движущие силы. Так, если говорить о случае наиболее популярного микроблогингового сервиса *Twitter*, в качестве потоков могут быть использованы потоки твитов, ретвитов и ответов; в качестве движущих сил — различного рода информация. Детальное описание указанных процессов даст возможность сформулировать и провести анализ неравновесной динамической модели *Twitter*. Очевидно, что данная система будет диссипативной. В диссипативных системах при определенных значениях контролируемых параметров возможно появление динамического хаоса (траектории динамической системы являются хаотическими) и хаотических аттракторов [8]. Актуальной станет задача управления хаосом в динамической системе *Twitter*.

Список литературы

1. **Губанов Д. А., Новиков Д. А., Чхартишвили А. Г.** Социальные сети: модели информационного влияния, управления и противоборства. М.: Издательство физико-математической литературы, 2010. 228 с.
2. **Wang F., Wang H., Xu K., Wu J., Jia X.** Characterizing Information Diffusion in Online Social Networks with Linear Diffusive Model // Proceedings of Distributed Computing Systems, IEEE 33rd International Conference, IEEE, 2013. P. 307–316.
3. **Малинецкий Г. Г.** Математические основы синергетики: хаос, структуры, вычислительный эксперимент. М.: Издательство ЛКИ. 2007. 312 с.
4. **Базыкин А. Д.** Нелинейная динамика взаимодействующих популяций. Москва—Ижевск: Изд. Института компьютерных исследований, 2003. 368 с.
5. **Цуканова О. А., Мальцева С. В.** Эволюционный процесс условного текстурирования сетевого пространства состояний в формате единого информационного ресурса сети // Автоматизация и современные технологии. 2013. № 11. С. 26–29.
6. **Teshl G.** Ordinary differential equations and dynamical systems // Graduate Student in Mathematics. Vol. 140. American mathematical society, Providence, RI, 2012. 356 p.
7. **Де Гроот С.** Термодинамика необратимых процессов. М.: ГИТТЛ, 1956. 280 с.
8. **Лоскутов А. Ю.** Очарование хаоса // Успехи физических наук. 2010. Т. 180, № 12. С. 1305–1329.

A. V. Dmitriev, Prof. Dr., **S. V. Maltseva**, Professor, **O. A. Tsukanova**, Graduate Student,
National Research University "Higher School of Economics
otsukanova@hse.ru, a.dmitriev@hse.ru, smaltseva@hse.ru

Modeling and Qualitative Analysis of a Social Microblogging Network as a Dynamical System

An approach to the investigation into the dynamics of social networks is described. The results of a qualitative analysis of the social network based on a microblogging principle are represented. The network interaction is approximated as a dynamic system characterizing the number of senders and recipients of small content elements.

The aim of the present paper is to make an investigation into the social network as a dynamical system, considering informational saturation and informational interaction between the senders and recipients. Systems without and with restrictions are investigated. It was determined that a social network, represented by a dynamic system considering saturation and interaction between the senders and recipients, is adequate. The parametric conditions of an asymptotic stability and forked bifurcation for the network dynamics are developed.

Keywords: mathematical modeling, social networks, microblogging, dynamical systems, non-linear dynamics, asymptotic stability, forked bifurcation

Reference

1. **Gubanov D. A., Novikov D. A., Chhartishvili A. G.** *Social'nye seti: modeli informacionnogo vlijaniya, upravleniya i protivoborstva* (Social Networks: Models of Informational Influence, Governance and Conflicts), Moscow, Izdatel'stvo fiziko-matematicheskoy literatury, 2010, 228 p.
2. **Wang F., Wang H., Xu K., Wu J., Jia X.** Characterizing Information Diffusion in Online Social Networks with Linear Diffusive Model, *Proceedings of Distributed Computing Systems, IEEE 33rd International Conference, IEEE*, 2013, pp. 307–316.
3. **Malineckij G. G.** *Matematicheskie osnovy sinergetiki: kaos, struktury vychislitel'nyj jeksperiment* (Mathematical Basics of Syner-gism: Chaos, Structures and Calculating Experiment), Moscow, Izdatel'stvo LKI, 2007, 312 p.
4. **Bazykin A. D.** *Nelinejnaja dinamika vzaimodejstvujushhih populjacij*. Moskva—Izhevsk: Institut komp'juternyh issledovanij (Non-linear Dynamics of Interacting Populations). — 2003. — 368 p.
5. **Tsukanova O. A., Maltseva S. V.** Evoljucionnyj process uslovnogo teksturovanija setevogo prostranstva sostojanij v formate edinogo informacionnogo resursa seti (Evolutionary Process of the Conditional Texturing of Network State Space as a Single Information Resource), *Avtomatizacija i sovremennye tehnologii* ("Automation and Modern Technologies"), 2013, no. 11, pp. 26–29.
6. **Teshl G.** *Ordinary Differential Equations and Dynamical Systems*, Graduate Students in Mathematics, vol. 140, American mathematical societv providence, Providence, 2012, RI, 356 p.
7. **De Groot S.** *Termodinamika neobratimyhprocessov* (Thermodynamics of Irreversible Processes), Moscow, GITTL, 1956, 280 p.
8. **Loskutov A. Ju.** Ocharovanie haosa (Charm of Chaos), *Uspеhi fizicheskikh nauk*, 2010, vol. 180, no. 12, pp. 1305–1329.

В. И. Дягилев, канд. техн. наук, доц., dvi-39@mail.ru,

В. А. Коковин, канд. техн. наук, доц., kokovin@uni-protvino.ru,

Государственный университет "Дубна", филиал "Протвино", г. Протвино, Московская область,

С. У. Увайсов, д-р техн. наук, проф., зам. директора по научной работе, s.uvaysov@hse.ru,

С. С. Увайсова, аспирант, uvay@yandex.ru,

Московский институт электроники и математики НИУ ВШЭ,

Национальный исследовательский университет Высшая Школа Экономики, г. Москва

Компьютерное моделирование работы силового преобразователя с выходным синусоидальным напряжением

Предложена математическая модель схемы силового преобразователя (СП) с выходным высокочастотным синусоидальным напряжением для питания пьезоэлектрических излучателей. Эти излучатели могут быть использованы в различных технологических установках.

На основе модели выполнены исследования, которые позволили сформулировать рекомендации для выбора параметров схемы преобразователя с индуктивно-емкостным фильтром.

Ключевые слова: компьютерное моделирование, силовой преобразователь, математическая модель, методика расчета

Введение

Известно [1], что для многих технологических операций, таких как ультразвуковая мойка, очистка, сварка, резание и т. п., применяют пьезокерамические и магнитострикционные преобразователи электрических колебаний в механические. Используемые для возбуждения таких преобразователей генераторы тока высокой частоты имеют, как правило, на выходе прямоугольное биполярное напряжение, которое с помощью фильтра первой гармоники преобразуется в синусоидальное [2]. Современные генераторы работают по принципу транзисторных инверторов — преобразователей постоянного напряжения в переменное [3], и электрические процессы в них достаточно сложны. В этом случае компьютерное моделирование упрощает задачу выбора компонентов этих генераторов при оптимизации системы генератор—нагрузка.

При разработке математической модели были учтены два условия: во-первых, реальные свойства всех компонентов схемы СП, во-вторых, особенности работы колебательного контура в резонансном режиме СП и нагрузки. При создании модели для выполнения первого условия были выбраны реальные мощные силовые ключи, а именно полевые MOSFET-транзисторы, а не их виртуальные модели. Кроме того, были учтены значения паразитных сопротивлений и индуктивностей источника питания, дросселя и выходного трансформатора. Выполнение второго условия позволяет оценить качество синусоиды выходного напряжения СП, а также получить расчетные формулы для опреде-

ления параметров силовых реактивных элементов — дросселя и конденсатора колебательного контура.

Проведенные на ЭВМ исследования предложенной модели показали, что существуют определенные трудности при расчете даже схемы замещения СП рассматриваемого типа ввиду нелинейности происходящих в нем процессов при переменной нагрузке. В предложенной работе показаны два пути преодоления таких проблем с использованием компьютерного моделирования и приведен пример расчета схемы СП для питания мощного пьезоэлектрического излучателя в режиме резонанса.

Разработка схемы замещения генератора

В общем случае силовой преобразователь состоит из источника переменного напряжения (промышленная сеть) и выпрямителя со сглаживающим фильтром. Для получения переменного напряжения с частотой, отличной от частоты промышленной сети, необходимо использовать инвертор (транзисторный или тиристорный) [4]. Как правило, современные СП средней мощности выполняют на транзисторах. При такой конфигурации СП в нагрузке при перекрестной коммутации двух пар транзисторов протекает переменный и прямоугольный по форме ток, равный $J_{\text{н}} = E/R_{\text{н}}$, где E — напряжение источника питания; $R_{\text{н}}$ — сопротивление нагрузки. Для правильного функционирования схемы необходимо выполнение следующих двух условий:

$$J_{\text{кл}} < J_{\text{доп}}; \quad (1)$$

$$t_{\text{вкл}} < T/2, \quad (2)$$

где $J_{\text{кл}}$ — ток через ключ; $J_{\text{доп}}$ — максимальный допустимый ток через транзистор; $t_{\text{вкл}}$ — включенное

Работа выполнена при финансовой поддержке РФФИ (грант № 14-07-00422).

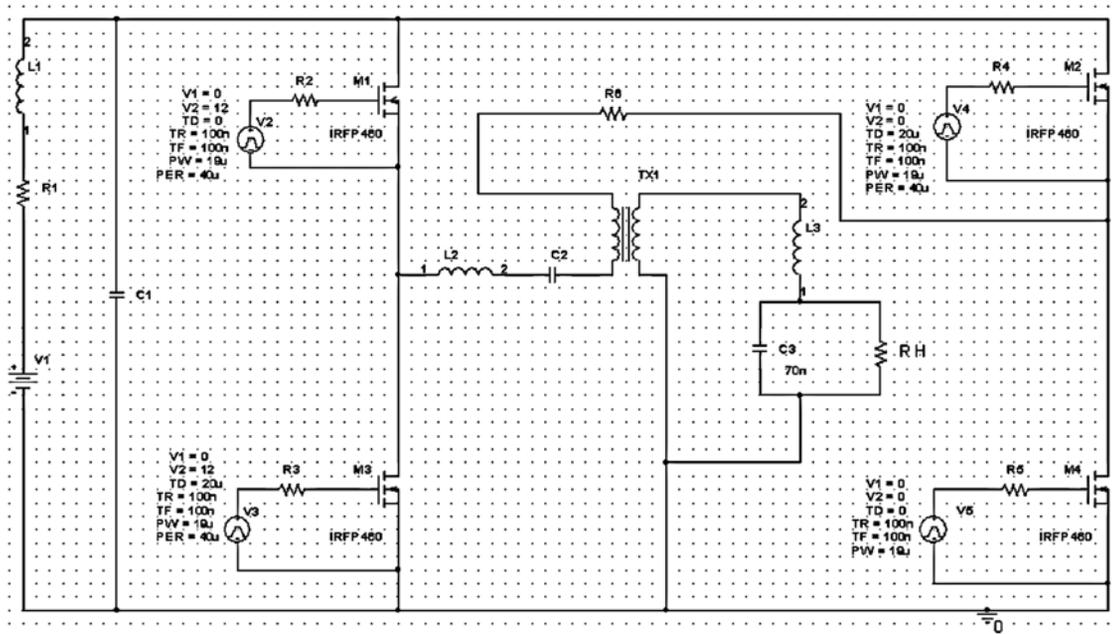


Рис. 1. Схема замещения СП

состояние транзистора; T — период отпираания двух пар транзисторов.

Для получения на выходе СП синусоидального напряжения необходимо дополнительно включить фильтр первой гармоники прямоугольного напряжения. Он представляет собой последовательный (или последовательно-параллельный) колебательный контур LC , настроенный в резонанс с первой гармоникой выходного напряжения [5]. Для выбора значений индуктивности дросселя L и емкости конденсатора C можно воспользоваться известными соотношениями [6], приведенными ниже.

Условие резонанса в цепи с колебательным контуром имеет следующий вид:

$$\omega_0^2 LC = 1, \quad (3)$$

где L и C — параметры колебательного контура; $\omega_0 = 2\pi f_0$ — круговая частота выходного напряжения; f_0 — частота требуемых колебаний рабочего органа технологической установки, $f_0 = 1/T_0 = 1/2t_{и}$, здесь $t_{и}$ — длительность импульса тока через транзистор; T_0 — период выходного синусоидального напряжения СП.

В этом случае можно записать

$$t_{и} < 1/\pi\sqrt{LC}. \quad (4)$$

Из этого выражения можно определить один из двух параметров колебательного контура (L или C). Для определения другого параметра воспользуемся выражением для волнового сопротивления контура

$$Z_B = \sqrt{L/C}. \quad (5)$$

Для нахождения значений L и C примем допущения [7], что

$$R_H = 1/\omega_0 C = \omega_0 L = Z_B. \quad (6)$$

Проверим правильность такого выбора с помощью исследования схемы замещения транзисторного генератора (рис. 1), входящего в состав СП, методом моделирования на ЭВМ.

Схема содержит: источник питания $V1$ с внутренними параметрами (сопротивлением $R1$ и индуктивностью $L1$); конденсатор фильтра $C1$; четыре мощных силовых транзистора $M1$ – $M4$ (MOSFET типа IRFP460) с цепями управления $V2$ – $V5$ и резисторами $R2$ – $R5$; цепь нагрузки, состоящую из колебательного контура (дроссель $L2$, резистор R_H и конденсатор $C3$) и выходного согласующего трансформатора $TX1$. Первичная обмотка трансформатора подключена через токоограничивающий последовательный конденсатор $C2$, индуктивность рассеивания $L2$ и активное сопротивление $R6$ в диагональ переменного тока транзисторного моста $M1$ – $M4$. Нагрузка R_H зашунтирована резонансным конденсатором $C3$. Этот контур подключен ко вторичной обмотке трансформатора $TX1$, с тем чтобы получить необходимое значение выходного напряжения. Такая схема позволяет работать СП в условиях большого диапазона изменения сопротивления нагрузки [8].

Моделирование схемы замещения генератора

Выберем для моделирования схемы значение $R_H = 100$ Ом, $f_0 = 25$ кГц, исходя из этого, с учетом допущения (6) рассчитаем $L3$ и $C2$: $L3 = R_H/(2\pi f_0) = 400$ мкГн. Соответственно, $C2 = 1/(R_H 2\pi f_0) = 40$ нФ.

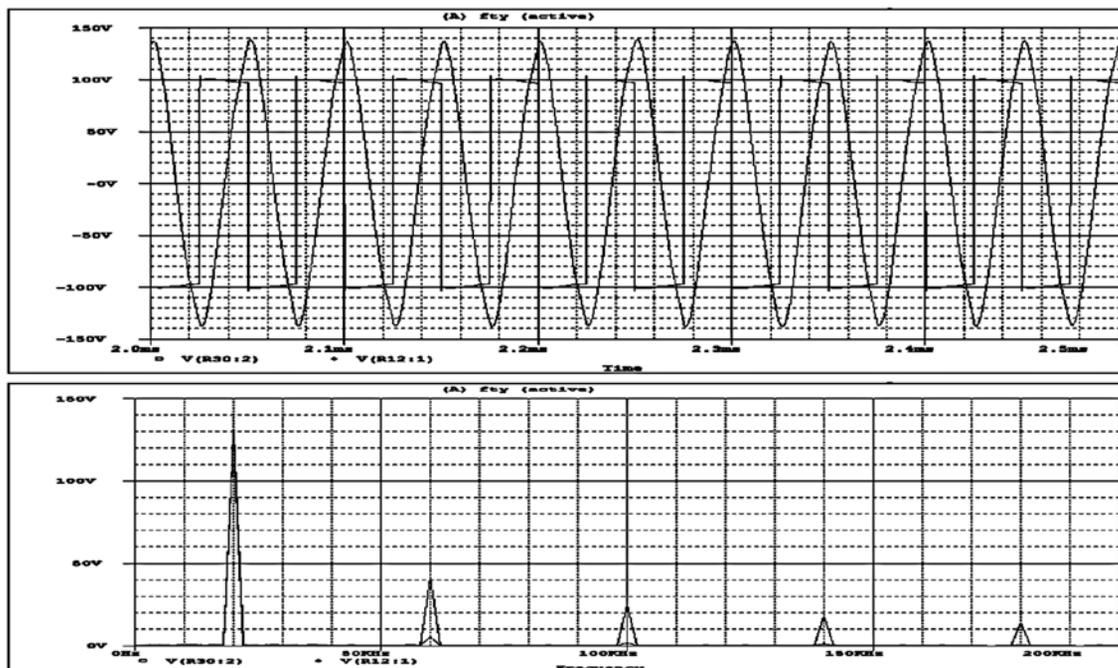


Рис. 2. Временные диаграммы на входе и выходе колебательного контура ($U_k(t)$ и $U_n(t)$ соответственно) — верхний рисунок; результат разложения прямоугольного напряжения в ряд Фурье. Нижняя кривая: ось ординат — напряжение в вольтах, ось абсцисс — частота в килогерцах. Верхняя кривая: ось ординат — напряжение в вольтах, ось абсцисс — время в микросекундах

На рис. 2 приведены временные зависимости $U_k(t)$ (U_k — входное напряжение колебательного контура $C3$ и $L3$). Первая кривая — это прямоугольное напряжение частотой 25 кГц с амплитудой, равной напряжению, питающему транзисторный мост ($M1-M4$). В нашем случае это значение равно 100 В. Вторая кривая — синусоидальное напряжение на выходе колебательного контура $U_n(t)$, и его амплитуда равна 135 В. Для получения первой (основной) гармоники используем разложение прямоугольного напряжения в ряд Фурье. Ее амплитуда равна 130 В (рис. 2). Примерное равенство этих амплитуд (не равны вследствие неидеальности дросселя $L3$) позволяет сделать заключение, что допущение (6) весьма удобно для расчетов параметров схемы генератора тока высокой частоты (ГТВЧ) в составе СП.

Решая (4), (5) и (6), найдем значение L и C :

$$L = t_n Z_B / \pi; C = t_n / \pi Z_B. \quad (7)$$

Теперь остается определить значение Z_B . Для этого необходимо найти коэффициент нагрузки K_H , равный соотношению значения сопротивления нагрузки R_H со значением волнового сопротивления Z_B :

$$K_H = R_H / Z_B. \quad (8)$$

Сопротивление нагрузки определяется заданными значениями (из технического задания) выходного напряжения U_H и мощностью в нагрузке P_H :

$$R_H = U_H^2 / P_H. \quad (9)$$

Чтобы определить K_H , необходимо знать зависимость $U_H(R_H)$. Эта зависимость не имеет аналитического выражения, так как определяется многими факторами: активным сопротивлением дросселя, внутренними параметрами источника питания, параметрами силовых транзисторов и т. д. Методом моделирования на ЭВМ [9] схемы замещения (см. рис. 1) можно получить зависимости $U_H(R_H)$ при различных значениях частот и параметров колебательного контура. Далее рассматривается ход таких исследований.

Для удобства проведения расчетов и построения универсальных характеристик воспользуемся способом расчета параметров СП, при котором значения величин (сопротивлений, напряжений, тока, т. д.) выражаются как множители определенной базовой величины, принятой за единицу. По такой методике принимается $U_H^* = U_H / E$. Здесь U_H^* — относительное значение напряжения на нагрузке (безразмерная величина).

Исследование на математической модели проводили таким образом, чтобы обеспечить инвариантность результатов. То есть необходимо доказать, что в относительных единицах кривая зависимостей $U_H(R_H)$ представляется как зависимость $U_H^*(K_H)$. Для определения параметров колебательного контура L и C необходимо построить зависимость $U_H^*(K_H)$, которая должна быть одинаковой для различных реальных значений сопротивлений нагрузки, выходных частот и питающих напряжений в относительных единицах. Расчет параметров схемы в относительных единицах позволяет получить ко-

Таблица 1

$E, В$	$R_H, Ом$	$Z_B, Ом$	$f, кГц$
100	1...1600	200	25
	1...800	100	
	1...400	50	
	1...200	25	
	1...80	10	

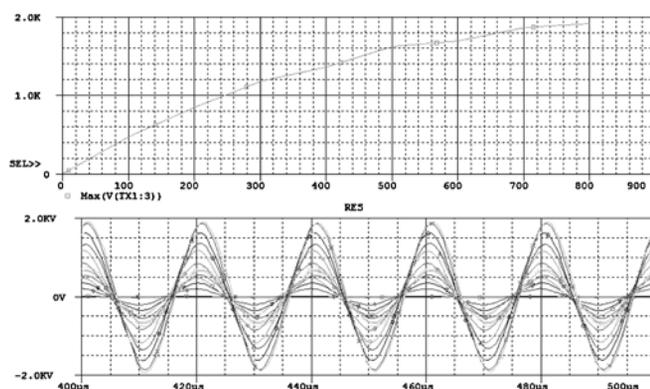


Рис. 3. Временные $U_H(t)$ (нижние) и параметрическая $U_{HM}(R_H)$ зависимости. Нижняя кривая: ось ординат — напряжение в вольтах, ось абсцисс — время в микросекундах. Верхняя кривая: ось ординат — напряжение в киловольтах, ось абсцисс — сопротивление в омах

Таблица 2

$Z_B, Ом$	100	100	100	100	100	100	100	100	100
$R_H, Ом$	1	100	200	300	400	500	600	700	800
K_H	0,01	1	2	3	4	5	6	7	8

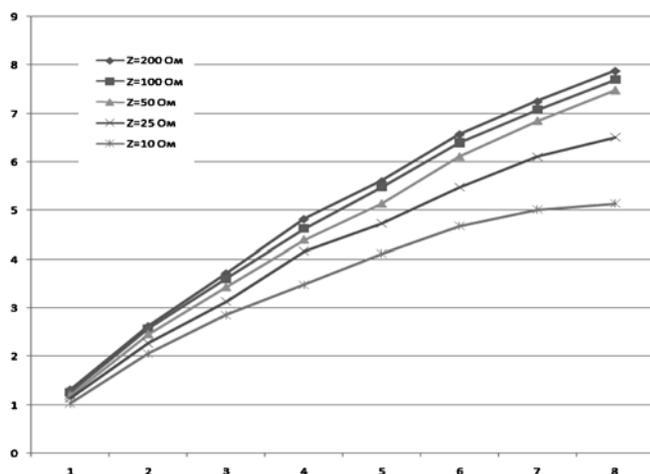


Рис. 4. Семейство параметрических зависимостей $U_{HM}^*(K_H)$ напряжения на нагрузке генератора с выходным колебательным контуром. По оси ординат откладывается напряжение U_{HM}^* в относительных единицах, а по оси абсцисс — значение K_H в относительных единицах

нечный результат для любых заданных параметров. Используя выражения (1–7) и воспользовавшись моделированием схемы замещения (см. рис. 1), можно получить зависимость $U_H^*(K_H)$.

Для демонстрации методики определения значений компонентов СП зададим диапазоны численных значений R_H при фиксированных значениях $E = 100 В$ и частоты генерации выходного напряжения $f = 25 кГц$. Зададим различные значения Z_B , которым соответствуют выбранные диапазоны значений R_H . Интервал значений изменяемости R_H определяется желаемой точностью построения кривой. Определенные значения представлены в табл. 1.

В процессе моделирования основное внимание было уделено построению параметрических зависимостей реальных значений $U_{HM}(R_H)$, $J_{BX}(R_H)$, $J_C(R_H)$ (где U_{HM} — амплитуда напряжения на нагрузке; J_{BX} — ток от источника питания, J_C — ток через конденсатор колебательного контура), при этом временные зависимости (рис. 3) использовали только для идентификации гармонических процессов.

В табл. 2 приведены данные одного из многих опытов при моделировании. Кроме того, в этой таблице рассчитан K_H для построения семейства кривых зависимостей $U_{HM}^*(K_H)$. Методика проведения опытов такова:

- задаем значение $Z_B = 100 Ом$;
- меняем нагрузочное сопротивление от 1 до 800 Ом;
- получаем временные и параметрические зависимости токов, напряжений и мощностей (см. рис. 3).

Ввиду большого объема экспериментального материала приведены только конечные результаты (рис. 4). На этом рисунке приведено семейство параметрических зависимостей $U_{HM}^*(K_H)$ напряжения на нагрузке генератора с выходным колебательным контуром. Кривые построены для различных значений волновых сопротивлений этого контура ($Z_B = 10...200 Ом$).

Анализ результатов исследования

Анализируя семейство кривых на рис. 4, можно сделать следующие замечания: кривые не совпадают и критерием различия является значение волнового сопротивления. Объяснить это явление можно следующими выводами:

- напряжение на резонансном конденсаторе $C3$ снижается при уменьшении шунтирующего сопротивления нагрузки;
- для различных значений Z_B значения $L3$ и $C3$ также различны;
- чем больше емкость $C3$, тем больше ток через нее, он создает дополнительные потери в элементах схемы генератора, особенно в транзисторах моста.

Для подтверждения этих выводов и в целях решения поставленной задачи (построить парамет-

Таблица 3

№	E , В	f , кГц	Z_B , Ом	J_{BX} , А	J_T , А	J_{C3} , А	P_{BX} , Вт	P_H , Вт
1	100	5	25	11,5	—	28,0	1100	1080
2	100	5	50	8,2	—	10,6	700	650
3	100	5	100	4,0	—	5,6	340	360
4	100	5	200	3,1	—	4,0	160	190
5	50	50	25	—	11,7	—	—	—
6	100	50	25	—	23,0	—	—	—
7	200	50	25	—	46,0	—	—	—

рическую зависимость $U_{HM}^*(K_H)$, одинаковую для различных реальных значений сопротивлений нагрузки, выходных частот и питающих напряжений (в относительных единицах) проведем расширенный эксперимент на математической модели. При его проведении фиксировали значения входного тока генератора (ток источника питания $J_{BX} = P_{BX}/E$, где P_{BX} — мощность источника питания) и амплитуды тока конденсатора колебательного контура J_{C3} , а также значения мощностей — входной P_{BX} и в нагрузке P_H при формировании выходного сигнала. Для выходного напряжения были заданы две частоты — 5 и 50 кГц. С уменьшением значения волнового сопротивления Z_B ток J_{C3} резко возрастает. Результаты измерений подтверждают сказанное и приведены в табл. 3. Данные приведены для случая, когда $K_H = 7$ (в табл. 3 J_T — ток силовых транзисторов).

Методика расчета генератора

С учетом разобшенности нагрузочных характеристик $U_H^*(K_H)$ предлагается следующая методика расчета СП. Пусть задано E , P_{HM} , U_{HM} , f (дополнительный индекс "м" у переменных P_H , U_H означает их амплитудное значение).

1. Для выполнения расчетов выберем кривую из семейства $U_H^*(K_H)$, построенную для контура с волновым сопротивлением, такую, где значение нагрузки (по заданию) и волнового сопротивления наиболее близки, например $Z_B = 100$ Ом (см. рис. 3).

2. Установим на ней для заданного значения E точку, соответствующую по ординате значению U_H^* .

3. Соответствующая ей абсцисса даст значение K_H .

4. Определим значение $R_H = K_H Z_B$.

5. Если найденное значение R_H не совпадает с заданным значением, т. е. $R_H \neq (U_{HM}^*/P_{HM})$, то для повышения или понижения напряжения на нагрузке необходимо использовать согласующий трансформатор, коэффициент трансформации которого $k = w_1/w_2$, w_1 , w_2 — число витков первичной и вторичной обмоток соответственно.

6. Известно [2], что приведенное к первичной обмотке сопротивление нагрузки $R'_H = k^2 \cdot R_H$. Согласно принятому нами выше допущению (6) R'_H должно быть равно Z_B . Тогда $k = \sqrt{R'_H/R_H}$. Далее по известной методике расчета трансформатора [2] находим значения w_1 и w_2 .

7. После определения числа витков трансформатора, зная первичное напряжение (из характеристики рис. 4), определяем токи в обмотках. Это позволяет выполнить расчет трансформатора.

Приведенная методика дает возможность сделать выбор компонентов СП.

Рассмотрим расчет генератора СП с числовыми значениями параметров. Пусть задано: $E = 100$ В, $P_{HM} = 500$ Вт, $U_{HM} = 500$ В, $f = 20$ кГц, $w_1 = 100$ витков. Необходимо определить токи и напряжения на элементах мостовой схемы СП и значения индуктивности дросселя L и емкости конденсатора C выходного фильтра. Далее выполним пошаговый расчет СП.

1. Построим кривую $U_H^*(K_H)$ для $Z_B = 100$ Ом.

2. Определим $U_{HM}^* = U_{HM}/E = 500/100 = 5$.

3. Используя построенную кривую зависимости $U_H^*(K_H)$, определяем $K_H = 4,25$.

4. Определим $R_H = K_H Z_B = 425$ Ом.

5. Определим коэффициент трансформации k согласующего трансформатора, используя формулу

$$k = \sqrt{R'_H/R_H} = 0,485.$$

6. По известным значениям w_1 и k получаем $w_2 = w_1/k = 100/0,485 = 235$ витков.

7. Для определения токов и напряжений можно в схему замещения (модель) схемы СП подставить полученные значения и смоделировать на ЭВМ.

Проверить правильность расчета можно с помощью эксперимента на физической модели или на ЭВМ.

Следует заметить, что имеется другой путь расчета параметров ГТВЧ для СП. Если сопротивление его нагрузки не равно волновому сопротивлению колебательного контура (принятому нами в примере $Z_B = 100$ Ом, см. рис. 4), то можно сделать расчет следующим образом. Выбираем $Z_B = R_H$ и делаем расчет на ЭВМ по указанной в данной работе методике. В этом случае не потребуется использовать согласующий выходной трансформатор для подключения нагрузки генератора.

Заключение

Разработана методика компьютерного моделирования генераторов СП, в составе которых имеются сложные резонансные схемы. На основе результатов моделирования предложена методика расчета таких генераторов и даны рекомендации по использованию параметрических зависимостей напряжения на нагрузке генератора с выходным колебательным контуром.

Работа выполнена при финансовой поддержке РФФИ (грант № 14-07-00422).

Список литературы

1. Гершгал Д. А., Фридман В. М. Ультразвуковая технологическая аппаратура. М.: Энергия, 1976. 319 с.
2. Розанов Ю. К. Основы силовой электроники. М.: Энергоатомиздат, 1992. 296 с.
3. Дягилев В. И., Коквин В. А., Увайсов С. У. Исследование схемы силового преобразователя с выходным синусоидальным напряжением для ВИЭ // Материалы Международной науч.-практ. конф. "Инновационные информационные технологии". Т. 2., М.: Изд-во НИУ ВШЭ, 2013. С. 99—106.
4. Беркович Е. И., Ивенский Г. В., Иоффе Ю. С. и др. Тиристорные преобразователи повышенной частоты для электро-технологических установок. Л.: Энергоатомиздат, 1983. 208 с.
5. Erickson R. W. Fundamentals of Power Electronics. 2nd Ed., Kluwer Academic Publishers, 2001. 912 p.
6. Петрухин Г. Д., Волощенко Ю. И., Никитина И. Н. Основы радиоэлектроники. М.: Изд-во МАИ, 1993. 415 с.
7. Дягилев В. И. Транзисторные генераторы тока высокой частоты для электротехнологий. Дубна: Лика, 2011. 67 с.
8. Diagilev V. I., Kokovin V. A., Uvaysov S. U. Designing a power converter with an adaptive control system for ultrasonic processing units // Lecture Notes in Control and Information Sciences. T. 22. Springer, 2015. P. 39—47.
9. Verdieri F., Bacha S., Gerbaud L. Automatic modelling of static converter averaged models. Toulouse. EPE, 2003. P. 1—9.

V. I. Diagilev, Associate Professor, dvi-39@mail.ru,

V. A. Kokovin, Associate Professor, kokovin@uni-protvino.ru

The Public Institution of Higher Education of Moscow Region the Dubna State University,
branch "Protvino", Protvino, Russia,

S. U. Uvaysov, Professor, National Research University

The Higher School of Economics, s.uvaysov@hse.ru, Moscow, Russia,

S. S. Uvaysova, Graduate Student, National Research University

The Higher School of Economics, uvay@yandex.ru, Moscow, Russia

Computer Simulation of the Power Converter with Harmonic wave Output

This paper presents mathematical model of power converter (PC) circuit with output high-frequency sinusoidal voltage for in-feeding the piezoelectric transducers. These transducers can be used in various technological installations. To develop the model we took into account two aspects: the real schema attributes of power converter and peculiar properties of the oscillation circuit in a resonant mode and at load.

The real powerful power switches were chosen, named field MOSFET — transistors, rather than virtual model. Moreover, we also took into account the values of parasitic resistances and inductances of the power source, the output transformer and inductor. The implementation of second aspect gives an opportunity to assess the sine wave quality of PC output voltage, as well as get formulas for calculating the parameters of reactive power elements — inductor and a capacitor oscillation circuit.

The research of the proposed model showed that there are certain difficulties in calculating the PC equivalent circuit. The difficulties are because of the nonlinearity of the processes occurring in PC at variable load. The paper shows two approaches to solve these problems with the use of computer modeling. The example of PC scheme calculating for in-feeding the powerful piezoelectric transducers in resonant mode is also shown.

The research was based on proposed model. That allowed formulate recommendations for selection the converter circuit parameters with LC filters.

Keywords: computer modelling, the power converter, mathematical model, calculation procedure

References

1. Gershal D. A., Fridman V. M. Ul'trazvukovaya tehnologicheskaya apparatura, Moscow, Jenergija, 1976, 319 p.
2. Rozanov Ju. K. Osnovy silovoj jelektroniki, Moscow, Jenergoatomizdat, 1992, 296 p.
3. Djagilev V. I., Kokovin V. A., Uvajsov S. U. Issledovanie shemy silovogo preobrazovatelja s vyhodnym sinusoidal'nym naprjazheniem dlja VIJe, Materialy mezhdunarodnoj nauch.-prakt. konf., Innovacionnye informacionnye tehnologii, T. 2, Moscow, NIU CShJe, 2013, pp. 99—106.
4. Berkovich E. I., Ivenskij G. V., Ioffe Ju. S. i dr. Tiristomye preobrazovateli povyshennoi chastoty dlja jelektrotehnologicheskikh ustanovok, Leningrad, Jenergoatomizdat, 1983, 208 p.
5. Erickson R. W. Fundamentals of Power Electronics, 2nd Ed., Kluwer Academic Publishers, 2001, 912 p.
6. Petruhin G. D., Voloshhenko Ju. I., Nikitina I. N. Osnovy radiojelektroniki, Moscow, MAI, 1993, 415 p.
7. Dyagilev V. I. Tranzistornye generatory toka vysokoj chastoty dlja elektrotehnologij (Transistor high frequency power generators for electric technologies.), Dubna, Lika, 2011, 67.
8. Diagilev V. I., Kokovin V. A., Uvaysov S. U. Designing a power converter with an adaptive control system for ultrasonic processing units, Lecture Notes in Control and Information Sciences. Springer, 2015, T. 22, pp. 39—47.
9. Verdieri F., Bacha S., Gerbaud L. Automatic modelling of static converter averaged models, Toulouse, EPE, 2003, pp. 1—9.

СИСТЕМЫ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ CAD-SYSTEMS

УДК 04.004

П. П. Олейник, канд. техн. наук, системный архитектор программного обеспечения, ОАО "Астон", доцент, Шахтинский институт (филиал) Южно-Российского государственного политехнического университета им. М. И. Платова, Ростов-на-Дону, xsl@list.ru,

В. И. Гурьянов, канд. техн. наук, доц.,

Филиал Санкт-Петербургского государственного экономического университета в г. Чебоксары, Чебоксары, vg2007sns@rambler.ru

UML-профиль для метамодельно-ориентированного проектирования программных приложений баз данных

В настоящее время часть вновь разрабатываемых приложений являются программными приложениями баз данных, для реализации функциональных возможностей которых чаще всего используют объектно-ориентированные (ОО) языки программирования, позволяющие создавать повторно-используемые фрагменты кода, что достигается применением инкапсуляции, наследования и полиморфизма. Дальнейшим развитием ОО-подхода является распространение его принципов на разработку всех уровней приложения от структуры БД и до графического интерфейса пользователя. В итоге разработчику предоставляется единая среда разработки, позволяющая создать конечное приложение. При этом на разработчика возлагается задача корректного проектирования и создания модели предметной области в понятиях выбранного инструмента.

В данной статье представлен UML-профиль, который многократно использован авторами при разработке приложений баз данных в собственной среде разработки. Этот профиль позволяет выполнить метамодельно-ориентированное проектирование приложения и описывает модель предметной области в понятиях развитой объектной метамодели. В конце статьи представлены примеры применения профиля и выдвинуты идеи о путях развития работы.

Ключевые слова: UML, пользовательский профиль UML, объектно-ориентированные базы данных, MDA, программные приложения баз данных, объектно-ориентированное проектирование, предметно-ориентированное проектирование, проектирование баз данных

Введение

В настоящее время часть вновь разрабатываемых приложений являются программными приложениями баз данных, для реализации функциональных возможностей которых чаще всего используют объектно-ориентированные (ОО) языки программирования, позволяющие создавать повторно-используемые фрагменты кода, что достигается применением инкапсуляции, наследования и полиморфизма. Дальнейшим развитием ОО-подхода является распространение его принципов на разработку всех уровней приложения от структуры БД и до графического интерфейса пользователя. В итоге разработчику предоставляется единая среда разработки, позволяющая создать конечное приложение. При этом на разработчика возлагается задача корректного проектирования и создания модели предметной области (ПрО) в понятиях выбранного инструмента.

Однако моделирование ПрО в понятиях среды разработки требует кардинального изменения мышления разработчика, что заставляет искать подходы к преодолению этого препятствия. Таковым под-

ходом может стать технология Model-driven architecture (MDA), смысл которой сводится к тому, что разработчик работает с UML-моделью, а код для языка программирования или платформы генерируется автоматически. Технически это можно реализовать посредством UML-профиля; элементы профиля поставляют данные, необходимые для генерации корректного кода. Профиль UML — это совокупность стереотипов, ограничений и помеченных значений, предназначенных для адаптации языка UML под специфические задачи проектирования, свойственные конкретным ПрО. Профиль создается посредством механизмов расширения UML, определенных в самом языке UML.

Ряд современных UML-редакторов поддерживают режим создания и работы с пользовательскими UML-профилями.

В данной работе представлен UML-профиль, который многократно использован авторами при разработке приложений баз данных в собственной среде разработки. Этот профиль позволяет выполнить метамодельно-ориентированное проектирование приложения и описывает модель предметной

области в понятиях развитой объектной метамодели. В конце статьи представлены примеры применения профиля и выдвинуты идеи о путях развития работы.

1. Обзор имеющихся работ

Благодаря своей гибкости и наличию множества спецификаций UML позволяет расширять существующую нотацию и добавлять новую семантику к существующим элементам (графическим объектам) диаграмм и тем самым использовать полученную нотацию для упрощения моделирования приложений различного назначения и для разных областей знаний [1, 2]. Этот подход активно применяют многие авторы, которые используют собственные UML-профили для своих потребностей (для адаптации UML под специфические задачи проектирования). Так, в работе [3] предложен подход концептуального проектирования реляционных баз данных с помощью построения диаграмм использования прецедентов (Use Case).

Весь процесс концептуального проектирования БД авторы разделяют на четыре этапа, после каждого из которых выполняется построение определенной UML-диаграммы. Результатом выполнения первого этапа является отображение диаграммы прецедентов для всех бизнес-процессов, которые необходимо отобразить сущностями в БД. После выполнения второго этапа авторы получают подробную диаграмму деятельности. На третьем этапе выполняется построение общей диаграммы классов, содержащей только информацию о классах, наследование и некоторые важные отношения ассоциаций. Завершающий четвертый этап позволяет построить подробную диаграмму классов, содержащую все выделенные классы и пригодную после использования правил отображений, описанных авторами, для реализации в РСУБД в виде набора реляционных отношений. Авторы подробно описали разработанный подход на примере проектирования приложений визового центра.

В работе [4] авторы предлагают UML-профиль, позволяющий упростить процесс разработки программных продуктов для промышленных объектов, которые управляют процессом выпуска продукции. Специфика данного ПО заключается в том, что оно работает в режиме реального времени под управлением соответствующей операционной системы с поддержкой параллельности на уровне ядра. При этом само предложение обеспечивает определенный уровень доступности услуг. Все это авторы учли в своем профиле и выделили пользовательские стереотипы для описания входных и выходных портов, контрольных устройств и таймеров, позволяющих их опрашивать.

Решение аналогичной задачи описано в работе [5], где рассматривается процесс обработки информации как набор выделенных авторами фаз и описы-

вается принцип передачи информации от одной фазы на другую. Сделано это в собственной среде разработки, построенной на базе Eclipse, которая управляется UML-диаграммами, описываемыми на языке SysML. Авторы предлагают использовать модельно-управляемую архитектуру и реализуют данный подход в виде набора плагинов. Затем разработчик создает с помощью предложенного профиля программно-независимую модель (PIM) и определяет правила трансформации в программно-зависимую модель (PDM). Сам процесс разработки представляет собой цепочку последовательных детализаций требования к системе, от высокоуровневого описания до построения диаграмм активности.

В работе [6] описан профиль, позволяющий моделировать высокоуровневую архитектуру любого программного продукта. Авторы представили набор введенных ими пользовательских стереотипов и краткие примеры их использования. В заключение дан комплексный пример и описаны основные атрибуты каждой представленной сущности на диаграмме классов.

В работе [7] описан подход, применяемый авторами при проектировании баз данных программных приложений реального времени, и представлен UML-профиль, используемый для решения задачи. Структура базы данных была логически разделена на методы, объекты и атрибуты реального времени. Затем в разработанном UML-профиле были выделены базовые типы данных и пользовательские стереотипы, а также описаны правила установления ассоциаций между объектами. В конце работы представлены правила трансформаций из UML-профиля в пользовательский тип данных, объявленный с помощью объектных расширений языка SQL.

В работе [8] авторы представляют пользовательский профиль, используемый для проектирования приложений, выполняющий загрузку информации в хранилища данных, а также их подготовку, т. е. для ETL-систем (Extract/Transform/Load). При этом авторы выделили набор базовых операций, которые выполняют системы, и на их основе определили пользовательские стереотипы. В конце работы представлен пример использования профиля и графический интерфейс реализованного приложения.

В работе [9] представлен профиль, позволяющий описать паттерны объектно-реляционного отображения (ORM). Это сделано для применения модельно-управляемой архитектуры при разработке приложений баз данных. Авторы ввели множество пользовательских типов данных и описали набор правил трансформации на РСУБД. При этом потребовалось описать такие объекты баз данных, как таблицы, первичные ключи, ограничения и правила соединения таблиц, а также некоторые синтаксические конструкции языка SQL. В заключение представлен пример отображения иерархии классов на таблицы БД.

В работе [10] описан профиль, применяемый авторами при проектировании геоинформационных систем и реализованный в UML-редакторе Enterprise Architect. Этот профиль предназначен для модельно-ориентированной архитектуры и задает набор правил трансформации моделей. В статье представлена тестовая предметная область и продемонстрирована поэтапная ее реализация вплоть до генерации SQL-скриптов для создания БД.

В работе [11] представлен профиль для проектирования сложных веб-ориентированных приложений. Профиль предполагает выделение трех этапов проектирования. На первом этапе выполняется концептуальное проектирование приложения, на втором этапе — навигационное проектирование, позволяющее определить правила перехода между страницами в приложении. А на заключительном этапе происходит проектирование графического интерфейса пользователя. Для реализации этого подхода авторами был создан собственный фреймворк, состав и структура модулей которого описаны в статье.

В работе [12] представлен модельно-ориентированный подход к проектированию мобильных приложений независимо от целевой платформы исполнения и без привязки к предметной области. Авторы предложили собственный UML-профиль, который применяется для построения диаграмм прецедентов, диаграмм классов и диаграмм конечных автоматов. Затем происходит генерация программного кода по созданной модели предметной области.

Консорциум OMG рекомендует использовать UML-профили как для поддержки технологии MDA для различных платформ и языков программирования, так и для хорошо определенных предметных областей [2]. Выше дан обзор разработки и использования профилей для различных платформ. Кратко рассмотрим применение профилей для моделирования предметных областей. Хорошо известными профилями моделирования являются: язык SysML/UML, предложенный OMG в качестве инструмента инженерного проектирования, и многочисленные профили моделирования бизнес-процессов, такие как Ericsson-Penker Profile и RUP Business Modeling. Моделирование предметных областей возможно потому, что UML — это формальный язык и элементам языка можно назначать разную семантику, а не только вычислительную. Из последних работ в этой области можно выделить работы по разработке профилей для моделирования онтологий, таких как OUP (Ontology UML Profile). Один из авторов данной статьи предложил UML-профиль для объектно-ориентированного имитационного моделирования [13, 14], который также можно отнести к этому направлению. Этот UML-профиль позволяет отображать концептуальные модели систем непосредственно на диаграммы классов, что дает возможность создавать объект-

ные имитационные модели. С одной стороны, этот профиль разрешает использовать *унифицированный процесс* для разработки имитационных моделей, с другой стороны, позволяет читать диаграммы классов как фреймовые семантические сети, что делает возможным выполнять валидацию имитационных моделей с самых первых шагов их разработки.

2. Метамоделно-ориентированный подход к разработке приложений баз данных

В настоящее время при проектировании новых программных продуктов на объектно-ориентированном языке программирования используется предметно-ориентированный подход (Domain-Driven Design, DDD) [15]. В результате применения предлагаемых наборов принципов и схем разработчикам удается создавать программные абстракции, которые называются моделями предметных областей и представляются в виде диаграмм классов унифицированного графического языка моделирования UML. Подход DDD полезен в ситуациях, когда разработчик программного продукта не является экспертом в прикладной предметной области разрабатываемого продукта и сталкивается с ней впервые. Он не может знать все области, но с помощью правильного представления структуры, посредством предметно-ориентированного подхода может быть спроектировано работоспособное приложение.

Полученная в ходе DDD диаграмма классов может быть реализована с помощью любого объектно-ориентированного языка программирования в выбранной среде разработки. Современные среды разработки используют модельно-управляемую архитектуру (Model-Driven Architecture, MDA). То есть создается определенная модель (представляющая модификацию модели предметной области), которая реализуется в выбранном инструменте разработки с использованием предоставленных средств.

В этом разделе представлено дальнейшее развитие DDD, которое названо авторами метамоделно-ориентированным подходом, основанным на создании экземпляров метаклассов объектной метамоделли. Решение авторы многократно применяли при реализации различных проектов. По сути разработчику предоставляется несколько видов сущностей, которые используются для построения модели предметной области. Виды создаваемых сущностей могут быть следующих типов.

1. **Сохраняемые.** Их используют для представления информации, которая физически хранится в базе данных. Описание подобных сущностей выполняется путем создания экземпляров метакласса DomainClass.

2. **Вспомогательные.** Экземпляры этих сущностей не сохраняются в БД, но их представление в системе необходимо для упрощения как в реализации бизнес-логики, так и для отображения вспомогательной информации в интерфейсе пользователя.

Описание подобных сущностей выполняется путем создания экземпляров метакласса HelperClass.

3. Классы-параметры методов. Основная бизнес-логика приложения реализуется в виде методов классов. Этот подход соответствует объектно-ориентированной парадигме. В конечном итоге методы представляются пользователю в виде элементов графического интерфейса, например в виде кнопок. При этом методы могут принимать набор параметров. Описание подобных сущностей выполняется с помощью создания экземпляров метакласса MethodParameterClass.

4. Классы-перечисления/множества. Эти классы используют в том случае, когда атрибут может принимать определенное значение из описанного ранее набора и этот набор не может расширяться пользователями. Описание подобных сущностей выполняется с помощью создания экземпляров метакласса Enum.

На рис. 1 представлен фрагмент метамодели, содержащий иерархию метаклассов, которые применяются для описания сущностей предметной области. Эта иерархия была разработана в унифицированной среде быстрой разработки корпоративных приложений SharpArchitect RAD Studio [16],

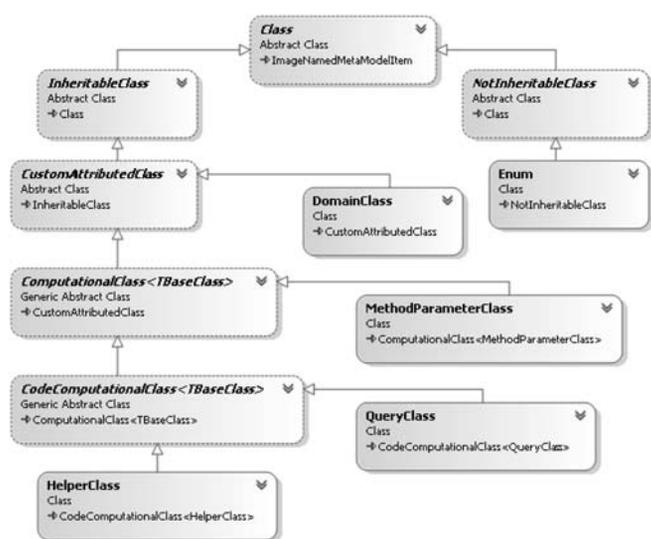


Рис. 1. Фрагмент метамодели, содержащий иерархию метаклассов описания сущностей предметной области

которая была протестирована при реализации различных приложений [17–20].

На рис. 2 представлен фрагмент метамодели, содержащий ассоциации между метаклассами.

Из рисунков видно, что представленная метамодель ортогональна по отношению к различным

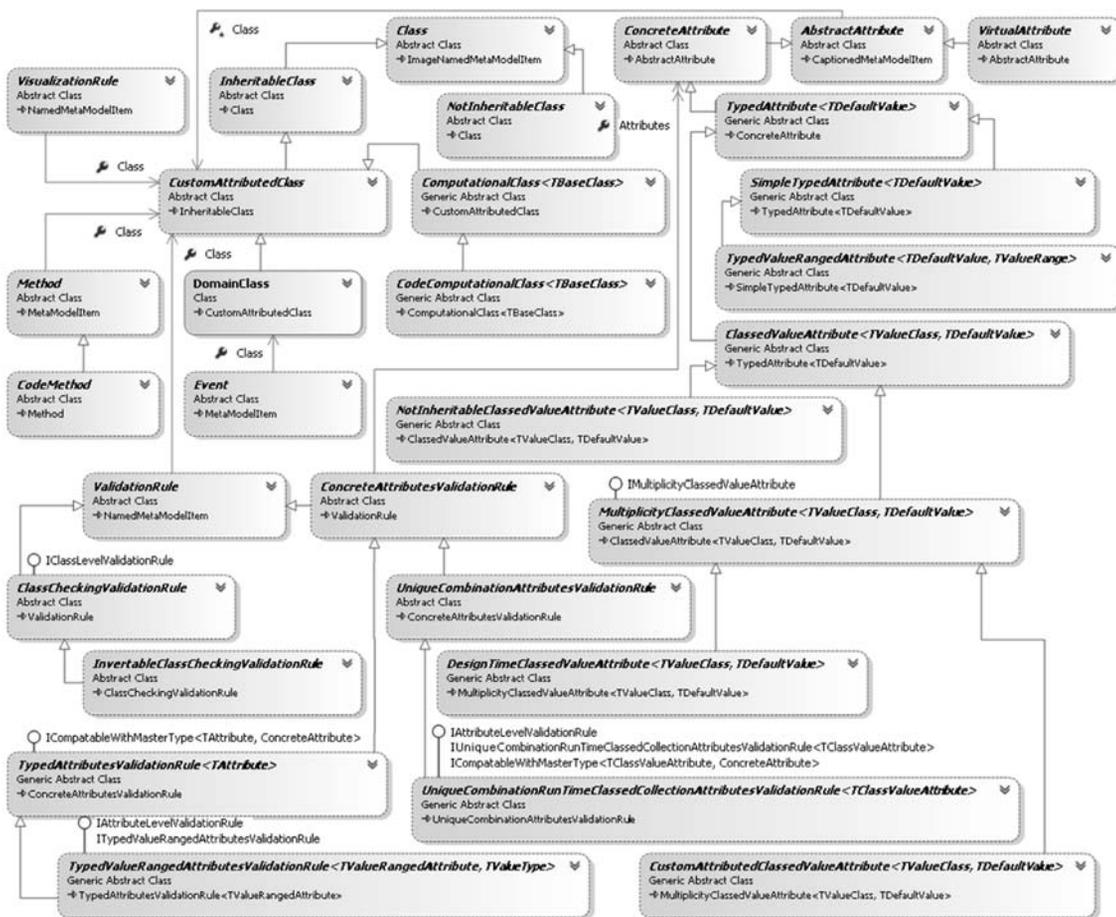


Рис. 2. Фрагмент метамодели, содержащий ассоциации между метаклассами

предметным областям, поэтому ее удалось применить во многих крупных приложениях. При этом данная метамодель используется на всех уровнях приложения, начиная от генерации структуры БД и заканчивая формированием графического интерфейса пользователя. В результате предлагаемый подход метамодельно-ориентированного проектирования имеет следующие преимущества:

- автоматическое создание интерфейса пользователя на основе модели приложения с возможностью настройки конечным пользователем;
- редактор модели приложения позволяет вносить изменения, а именно — описывать классы, связывать их между собой в соответствии с UML-диаграммами и задавать сигнатуру методов в момент выполнения приложения; при этом для вступления изменений достаточно перезапустить приложение одного конкретного пользователя (остальные могут продолжить работу);
- возможность декларативного описания валидационных правил, позволяющих проверять корректность и непротиворечивость данных в момент сохранения их в базе данных;
- возможность декларативного описания визуализационных правил, позволяющих выделить с помощью изменения цвета различные графические элементы на форме, а также управлять видимостью и доступностью элементов;
- множество системных (встроенных) классов, которое позволяет упростить реализацию поведения часто используемых видов сущностей (например, в системе имеется внутренняя поддержка древовидных структур).

3. Необходимость разработки собственного UML-профиля

Разработка собственного профиля — кропотливый и трудоемкий процесс. При проектировании приложений баз данных в рассмотренной среде разработки SharpArchitect RAD Studio один из авторов ранее использовал иной подход, основанный на создании диаграмм объектов, содержащих экземпляры метаклассов, описывающих сущности предметной области (см. рис. 1) [21—22]. На рис. 3 представлен используемый подход.

Несмотря на то что описанное решение было протестировано многократно, за время эксплуатации обнаружены следующие проблемы.

1. Полученная диаграмма объектов не похожа на первоначальную диаграмму классов. Это становится серьезной

проблемой при доработке исходной диаграммы. Так как нет прямого соответствия, то не ясно, куда именно вносить изменения в диаграмме объектов.

2. Полученная диаграмма более громоздкая по сравнению с исходной. Эта проблема становится очень серьезной при проектировании программного приложения, в котором имеются сотни различных сущностей.

3. Для моделирования повсеместно используются экземпляры классов, что стирает грань между атрибутами и классами предметной области. Диаграмма классов оперирует такими понятиями, как класс, атрибут, ассоциация. С концептуальной точки зрения это совершенно различные сущности, которые нельзя смешивать. В предложенном подходе они смешаны.

4. Представленный подход требует от проектировщика глубокого знания не только диаграмм классов, но и диаграмм объектов. Это особенно сложно для людей, начинающих изучать UML, например для студентов вузов.

5. Наличие на диаграмме экземпляров вспомогательных классов, которые отсутствуют в пред-

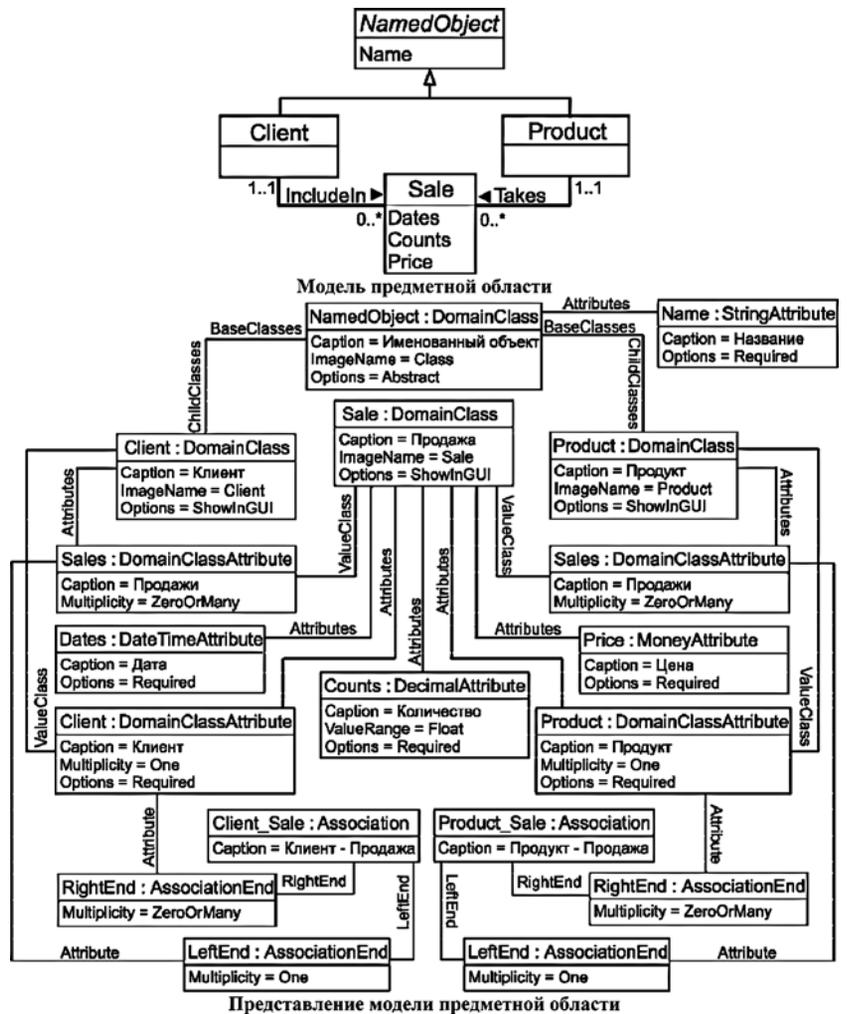


Рис. 3. Проектирование приложения в понятиях метамодели объектной системы с применением диаграммы объектов

метной области. Например, класс AssociationEnd необходим для представления края ассоциации и не присутствует в предметной области. Однако при описании ассоциации экземпляры этого класса загромождают диаграмму объектов.

Анализ выделенных недостатков свидетельствует, что достоинств гораздо меньше. В результате возникла необходимость поиска альтернативного решения. В нашем случае оптимальным решением является разработка собственного профиля.

4. Состав и структура разработанного UML-профиля

UML активно используется при проектировании различных элементов информационных систем, в том числе приложений баз данных. Известны также профили UML, предназначенные для проектирования структуры реляционных БД (РБД). Далее предлагается профиль SharpArchitect UML Profile (SAUP), который предназначен для проектирования ООБД в понятиях метамодели объектной системы [23].

В настоящее время объектно-ориентированный подход используется повсеместно, в том числе и при реализации уровня хранения данных. Использование ОО-парадигмы на уровне БД позволяет применять такие механизмы, как наследование, что отсутствует в других моделях данных, например в РБД. Так, мы можем объявить базовый класс и реализовать в нем наиболее общие атрибуты, а затем унаследовать их в производных классах. При использовании РСУБД разработчику необходимо в каждом отношении, реализуемом таблицей, повторно объявлять одинаковые атрибуты. Также ОО-подход позволяет реализовать древовидные (иерархические) структуры с реализацией рекурсивных запросов, а это довольно трудоемкая задача для реляционной модели данных.

Профиль ориентирован на среду разработки SharpArchitect RAD Studio. Профиль SAUP разработан на основе спецификации UML 2.4.1 [2]. Процесс разработки профиля предполагает выполнение следующих шагов: 1) определение метамодели платформы; 2) с использованием элементов метамодели

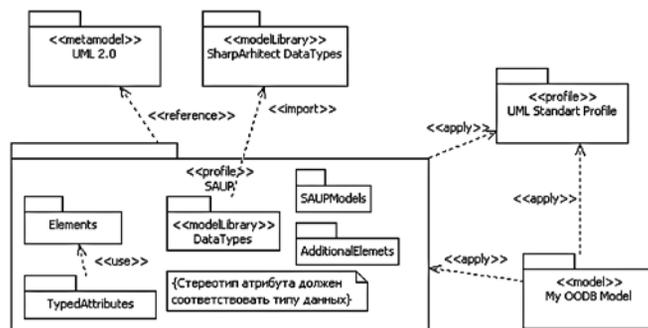


Рис. 4. Архитектура UML-профиля SharpArchitect UML Profile (SAUP)

платформы определение множества стереотипов на базе существующих элементов метамодели UML; 3) для каждого стереотипа определение помеченных значений (*tagged values*) и ограничений (*constraints*). Множество стереотипов помещается в пакет со стереотипом "profile", и выполняется распределение стереотипов по семантически однородным группам. В качестве метамодели платформы используется метамодель, приведенная на рис. 1. Общая структура профиля показана на рис. 4.

Пакет Elements определяет элементы профиля, моделирующие сущности предметной области. Пакет TypedAttributes определяет набор стереотипов для атрибутов с типами данных, определенных в пакете DataTypes, который, в свою очередь, импортирует типы данных из среды разработки SharpArchitect RAD Studio (см. рис. 4). В профиле задано ограничение — стереотипы атрибутов должны соответствовать типам данных. Это ограничение задается специальной таблицей, где каждому типу ставится в соответствие стереотип. Пакет AdditionalElements определяет стереотипы, предназначенные для моделирования вспомогательных классов. Пакет SAUPModels дает определение моделям SAUP. Для построения метамодели профиля использованы элементы стандартного профиля UML (стереотипы и типы данных).

Структура пакета Elements показана на рис. 5. Каждый стереотип определяет некоторое множество помеченных значений, которые поставляют данные классам среды проектирования SharpArchitect RAD Studio (см. рис. 1, 2). Стереотип **DomainClass** определяет ряд помеченных значений, которые поставляют данные классу предметной области DomainClass. Стереотип **DomainAssociation** поставляют данные классам Association и AssociationEnd. Стереотип **EnumClass** в качестве метакласса использует элемент **Class**, а не элемент **Enumeration**. Стереотип **TypedAttribute** определяет общие помеченные значения для классов типов данных (рис. 6). Поскольку все эти стереотипы расширяют один и тот

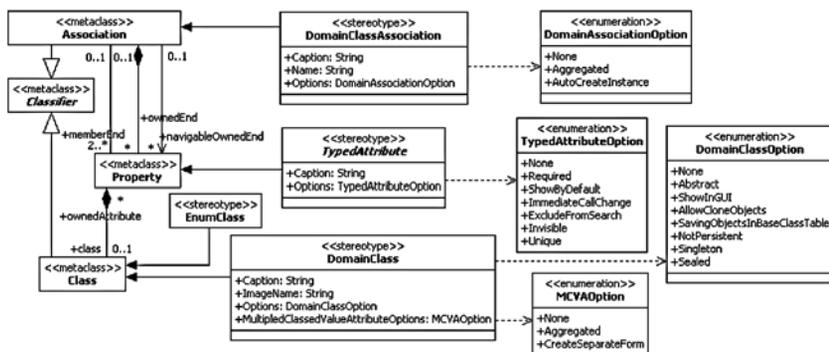


Рис. 5. Структура пакета Elements

же UML-элемент **Property**, все они сгруппированы в пакете `TypedAttributes`. Стереотипы атрибутов, определяемые в пакете `TypedAttributes`, находятся в отношении обобщения со стереотипом `TypedAttribute`, что отражено в зависимости пакета `TypedAttributes` от пакета `Elements`.

Девять стереотипов (**DecimalAttribute**, **EnumAttribute**, **FileDataAttribute**, **HelperClassAttribute**, **IntAttribute**, **MetaModelClassAttribute**, **TextAttribute**, **TypeAttribute**, **StringAttribute**) имеют дополнительные помеченные значения. Остальные стереотипы атрибутов просто наследуют помеченные значения от стереотипа `TypedAttribute`.

Структура пакета `AdditionalElements` представлена на рис. 7.

Пакет `AdditionalElements` определяет ряд дополнительных стереотипов для определения сущностей, отличающихся от сущностей предметной области. Стереотип **HelperClass** помечает вспомогательные классы, которые не хранятся в ООБД. Стереотип **SystemClass** помечает системные классы, например `BaseRunTimeTreeNodeDomainClass`. Эти классы предназначены для реализации типовых решений, таких как паттерны проектирования. Два стереотипа **Parameter** и **MethodParameterClass** предназначены для отображения класса-параметра `MethodParameterClass`. Кроме того, в этом пакете определен стереотип **N-AryAssociation**, который обозначает *n*-арные ассоциации. В качестве метаклассов использованы UML-элементы **Class** и **Association**. В отличие от других стереотипов, этот стереотип не имеет аналогов в метамодели и предназначен для концептуального проектирования.

Пакет `SAUPModels` содержит определение всего одного стереотипа **SAUPModel** на основе метакласса **Model** (из пакета `Models`). Тем самым модель `SAUP` определяется как пакет для размещения как элементов профиля, так и элементов UML.

Как видно из краткого описания `SAUP`, разработанный профиль позволяет проектировать объ-

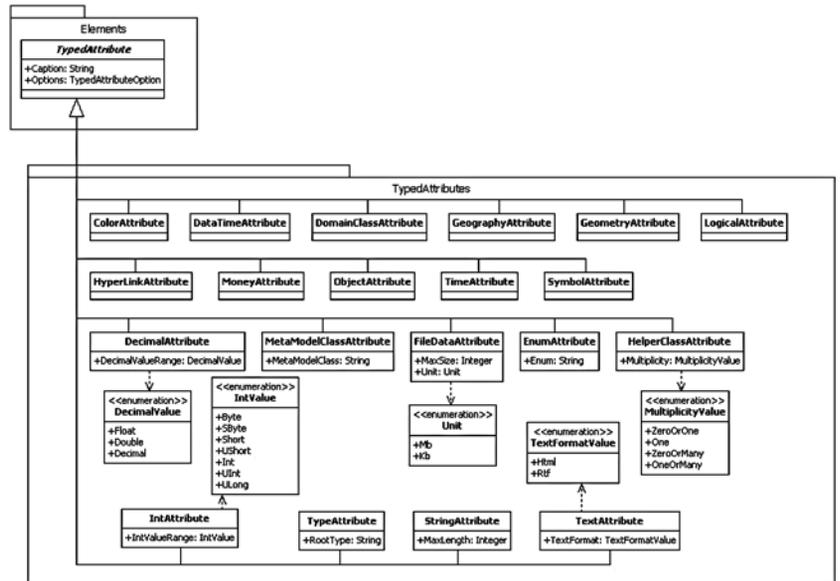


Рис. 6. Структура пакета `TypedAttributes`

ектно-ориентированные приложения баз данных для различных прикладных предметных областей. При этом проектирование с применением UML профиля по сути представляет собой построение диаграммы классов, что вполне логично и естественно по сравнению с построением диаграмм объектов метаклассов объектной системы, предложенным в работах [21, 22]. Авторы считают это решение оптимальным и намерены использовать в будущем только его.

5. Примеры использования разработанного UML-профиля

Рассмотрим несколько примеров применения профиля. Для демонстрации возможностей профиля необходимо рассмотреть относительно большую модель, содержащую множество классов и отношений между ними. Некоторое время назад для демонстрации собственной среды разработки `SharpArchitect RAD Studio` и реализованных в ней функциональных возможностей потребовалось спроектировать тестовую предметную область. Для модели были выделены критерии оптимальности и получена соответствующая реализация. В результате подход был обобщен, и появилась унифицированная модель тестирования инструментов разработки объектно-ориентированных приложений баз данных [23, 24]. Именно эта модель изображена в нотации `SAUP` на рис. 8.

Стереотипом "N-AryAssociation" помечена *n*-арная ассоциация **Position**. На диаграмме также присутствуют стандартные UML-элементы, например класс-ассоциация. Стереотипы отчетливо показывают, какие элементы модели должны быть доопределены. На диаграмме в качестве примера для классов со стереотипом `DomainClass` явно показаны

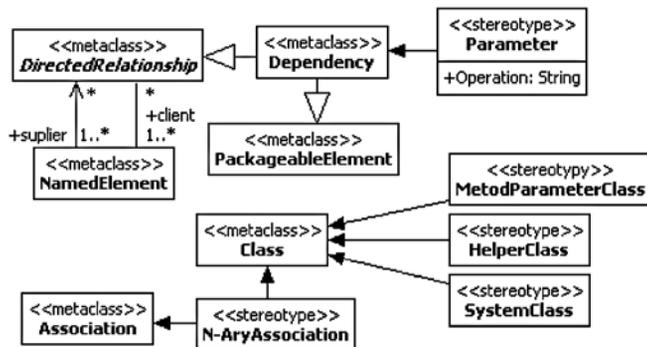


Рис. 7. Структура пакета `AdditionalElements`

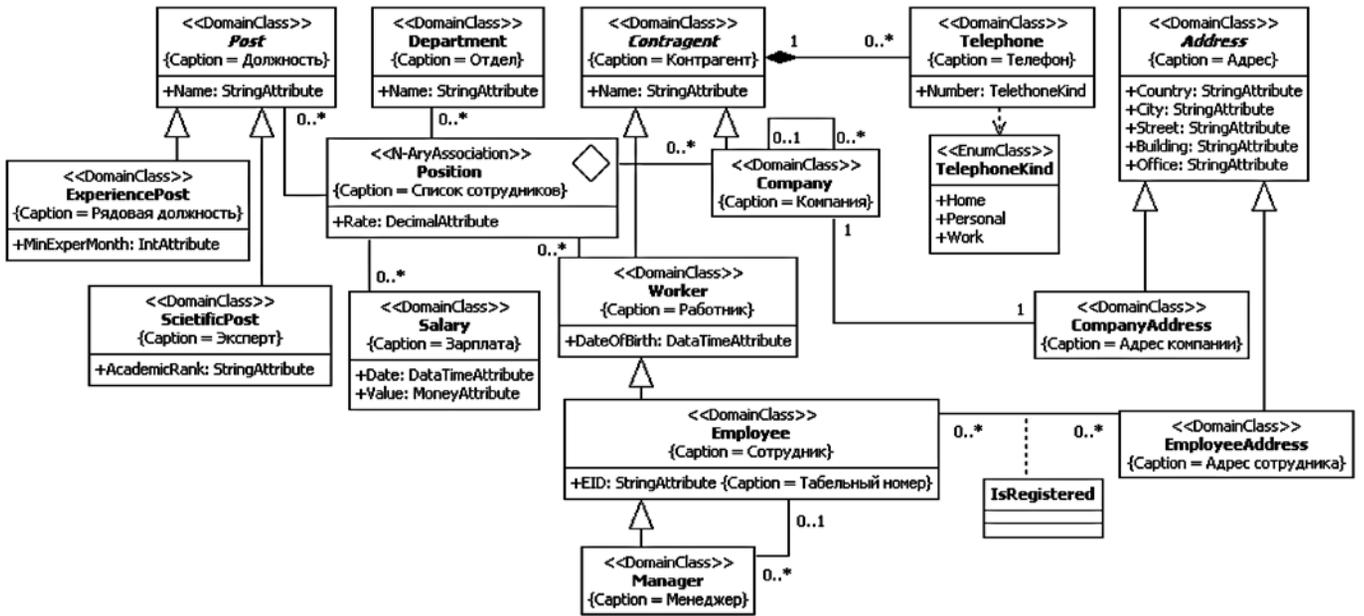


Рис. 8. UML-диаграмма классов унифицированной модели тестирования инструментов разработки объектно-ориентированных приложений, построенная с применением профиля SAUP

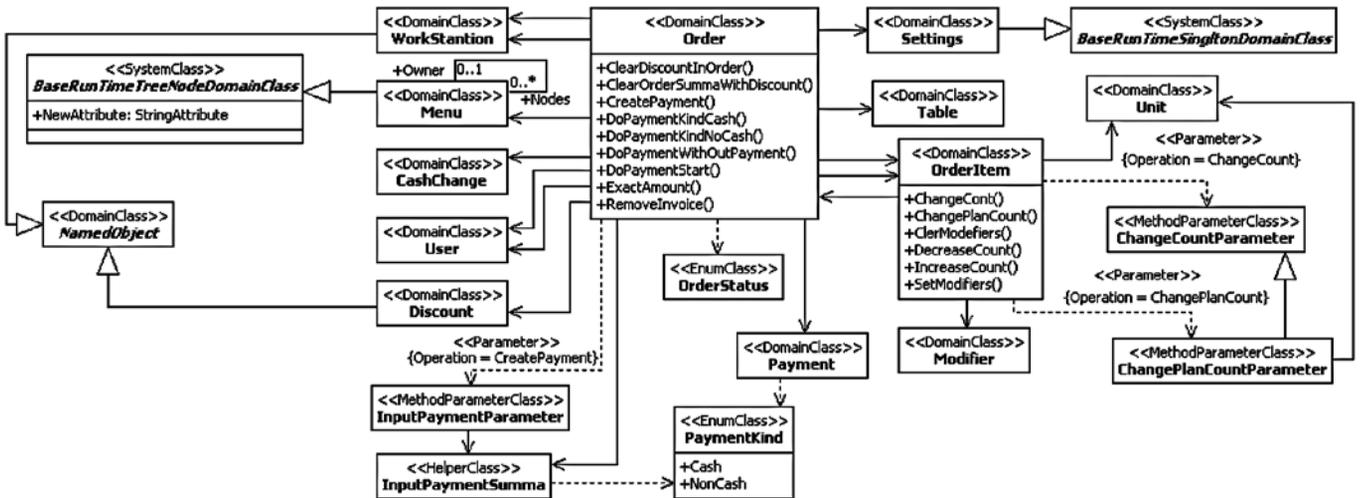


Рис. 9. Фрагмент UML-диаграммы классов информационной системы для ресторанов быстрого питания

помеченные значения Caption, а для класса **Employee** также одно из помеченных значений стереотипа атрибута ED. Хотя все атрибуты проектной модели должны быть помечены стереотипами, их можно не отображать на диаграммах. Точно так же можно скрывать помеченные значения стереотипов. Это позволяет проводить концептуальное и логическое проектирование, не детализируя спецификацию UML-элементов.

Как видно из рис. 8, на нем представлены только классы предметной области, позволяющие описать сохраняемые типы сущностей. В работе [26] при прототипировании графической формы заказа в приложении, разработанном для сенсорного экрана, потребовалось объявить множество дополнительных классов (HelperClass и др., см. рис. 1). У авторов возник ряд проблем при отображении диаграммы

классов. Ниже представлено решение с помощью профиля SAUP (рис. 9).

Данная диаграмма иллюстрирует использование стереотипов пакета AdditionalElements профиля SAUP. Системный класс **BaseRunTimeTreeNodeDomainClass** реализует паттерн *Composite* и помечается стереотипом "SystemClass". Иногда возникает необходимость добавить дополнительные атрибуты в этот и другие системные классы. Это показывается просто как дополнительный атрибут этого класса с типом из библиотеки SharpArhitect Data Types. Класс-параметр **InputPaymentParameter** помечен стереотипом "MethodParameterClass" и находится в отношении зависимости с классом **Order**. Отношение зависимости помечается стереотипом "Parameter" и имеет помеченное значение Operation, которое называет операцию, для которой соз-

дается параметр. Вспомогательный класс **InputPaymentSumma** используется при оплате заказа как наличным, так и безналичным расчетом и представлен для демонстрации использования параметров методов.

Приведенные примеры наглядно демонстрируют, что применение профиля SAUP позволяет организовать мышление проектировщика в понятиях метамодели объектной системы, сохраняя возможность проектирования в диаграммах классов UML.

Заключение

Обзор литературы показал, что благодаря своей гибкости UML можно применять для приложений различных назначений, для этого достаточно разработать собственный профиль. В данной работе представлен процесс метамодельно-ориентированного проектирования приложений баз данных, суть которого заключается в создании экземпляров метаклассов объектной системы. При этом авторы разработали собственный UML-профиль, позволяющий упростить представленный подход. Описанные примеры доказали зрелость предложенного подхода. Дальнейшим развитием профиля является представление динамической составляющей, а именно методов классов, валидационных и визуализационных правил.

Список литературы

1. **The UML and Data Modeling**, White Paper: Technical report, Rational Software, 2003.
2. **Object Management Group**. UML 2.0 Infrastructure Specification, OMG document number formal/2011-08-05, 2011.
3. **Brdjanin D., Maric S.** An Example of Use-Case-driven Conceptual Design of Relational Database // EUROCON, 2007. The International Conference on "Computer as a Tool", 9–12 Sept. 2007, Warsaw, pp. 538–545, DOI: 10.1109/EURCON.2007.4400437
4. **Ritala T., Kuikka S.** UML Automation Profile: Enhancing the Efficiency of Software Development in the Automation Industry // 5th IEEE International Conference on Industrial Informatics, 23–27 June 2007, Vienna. P. 885–890. DOI: 10.1109/INDIN.2007.4384890
5. **Vepsalainen T., Hastbacka D., Kuikka S.** Tool Support for the UML Automation Profile — For Domain-Specific Software Development in Manufacturing // The Third International Conference on Software Engineering Advances, ICSEA '08, 26–31 Oct. 2008, Sliema. P. 43–50. DOI: 10.1109/ICSEA.2008.22
6. **Hengye Zhu, Guangyao Li, Liping Zheng.** A UML profile for HLA-based simulation system modeling // 6th IEEE International Conference on Industrial Informatics, INDIN 2008, 13–16 July 2008, Daejeon. P. 1602–1607. DOI: 10.1109/INDIN.2008.4618360
7. **Idoudi N., Duvallet C., Bouaziz R., Sadeg B., Gargouri F.** How to Model a Real-Time Database? // Proc. of IEEE International Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing, ISORC '09, 17–20 March 2009, Tokyo. P. 321–325. DOI: 10.1109/ISORC.2009.17
8. **Xudong Song, Xiaolan Yan, Liguang Yang.** Design ETL Metamodel Based on UML Profile. Second International Symposium on Knowledge Acquisition and Modeling, KAM '09, Nov. 30. Dec. 1, 2009, Wuhan. P. 69–72. DOI: 10.1109/KAM.2009.112
9. **Torres A., Galante R., Pimenta M. S.** Towards a UML Profile for Model-Driven Object-Relational Mapping // Proc. of XXIII Brazilian Symposium on Software Engineering, SBES '09, 5–9 Oct. 2009, Fortaleza, Ceara. P. 94–103. DOI: 10.1109/SBES.2009.22
10. **Ferreira T. B., Stempluc S. M., Lisboa-Filho J.** Geographical data modeling with the UML GeoProfile and MDA transformations on the Enterprise Architect Tool // Proc. of 9th Iberian Conference on Information Systems and Technologies (CISTI), 18–21 June 2014, Barcelona. P. 1–6. DOI: 10.1109/CISTI.2014.6876987
11. **Mubin S. A., Jantan A. H.** A UML 2.0 profile web design framework for modeling complex web application. International Conference on Information Technology and Multimedia (ICIMU), 18–20 Nov. 2014, Putrajaya. P. 324–329. DOI: 10.1109/ICIMU.2014.7066653
12. **Usman M., Iqbal M. Z., Khan M. U.** A Model-Driven Approach to Generate Mobile Applications for Multiple Platforms // Proc of 21st Asia-Pacific Software Engineering Conference (APSEC), 1–4 Dec. 2014, Jeju. P. 111–118. DOI: 10.1109/APSEC.2014.26
13. **Гурьянов В. И.** Моделирование классификаций на визуальном языке имитационного моделирования UML SP // Сборник докладов Шестой всероссийской научно-практической конференции "Имитационное моделирование. Теория и практика" (ИММОД-2013). Т. 1. Казань: Изд-во "ФЭН" АН РТ, 2013. С. 128–132.
14. **Гурьянов В. И.** Имитационное моделирование на UML SP. Чебоксары: Изд-во Филиала СПбГЭУ в г. Чебоксары, 2014. 135 с.
15. **Эванс Э.** Предметно-ориентированное проектирование (DDD). Структуризация сложных программных систем. М.: Вильямс, 2010, 423 с.
16. **Олейник П. П.** Программа для ЭВМ "Унифицированная среда быстрой разработки корпоративных информационных систем SharpArchitect RAD Studio". Свид-во о гос. регистрации № 2013618212 от 04 сентября 2013 г.
17. **Олейник П. П.** Элементы среды разработки программных комплексов на основе организации метамодели объектной системы // Бизнес-информатика. 2013. № 4 (26). С. 69–76. URL: [http://bijournal.hse.ru/data/2014/01/16/1326593606/IBI%204\(26\)%202013.pdf](http://bijournal.hse.ru/data/2014/01/16/1326593606/IBI%204(26)%202013.pdf)
18. **Олейник П. П., Кураков Ю. И.** Концепция создания обслуживающей корпоративной информационной системы экономического производственно-энергетического кластера // Прикладная информатика. 2014. № 6. С. 5–23.
19. **Oleynik P. P., Nikolenko O. I., Yuzefova S. Yu.** Information System for Fast Food Restaurants // Engineering and Technology. 2015. Vol. 2, No. 4. P. 186–191. <http://article.aascit.org/file/pdf/9026895.pdf>
20. **Oleynik P. P.** Metamodel-Driven Design of Database Applications. // Journal of Computer Science Technology Updates. 2015. Vol. 2, No. 1. P. 15–24. <http://www.cosmoscholars.com/images/JCSTU-v1n1/JCSTU-V2-N1/JCSTU-V2N1A3-Oleynik.pdf>
21. **Олейник П. П.** Предметно-ориентированное проектирование структуры базы данных в понятиях метамодели объектной системы // Объектные системы — 2014: материалы VIII Международной научно-практической конференции (Ростов-на-Дону, 10–12 мая 2014 г.) / Под общ. ред. П. П. Олейника. Ростов-на-Дону: ШИ (ф) ЮРГПУ (НПИ) им. М. И. Платова, 2014. С. 41–46. URL: http://objectsystems.ru/files/2014/Object_Systems_2014_Proceedings.pdf
22. **Oleynik P. P.** Using metamodel of object system for domain-driven design the database structure // Proc. of 12th IEEE East-West Design & Test & Symposium (EWDTS'2014), Kiev, Ukraine, September 26–29, 2014. DOI: 10.1109/EWDTS.2014.7027052
23. **Гурьянов В. И., Олейник П. П.** UML-профиль проектирования структуры объектно-ориентированной базы данных // Объектные системы — 2015: материалы X Международной научно-практической конференции. Ростов-на-Дону, 10–12 мая 2015 г. / Под общ. ред. П. П. Олейника. Ростов-на-Дону: ШИ (ф) ЮРГПУ (НПИ) им. М. И. Платова, 2015. URL: http://objectsystems.ru/files/2015/Object_Systems_2015_Proceedings.pdf
24. **Олейник П. П.** Унифицированная модель тестирования инструментов разработки объектно-ориентированных приложений // Объектные системы — 2014 (Зимняя сессия): материалы IX Международной научно-практической конференции. Ростов-на-Дону, 10–12 декабря 2014 г. / Под общ. ред. П. П. Олейника. Ростов-на-Дону: ШИ (ф) ЮРГПУ (НПИ) им. М. И. Платова, 2014. — С. 25–35. URL: http://objectsystems.ru/files/2014WS/Object_Systems_2014_Winter_session_Proceedings.pdf
25. **Николенко О. И., Олейник П. П., Юзефова С. Ю.** Прототипирование и реализация графической формы заказа для информационной системы ресторанов быстрого питания // Объектные системы — 2015: материалы X Международной научно-практической конференции. Ростов-на-Дону, 10–12 мая 2015 г. / Под общ. ред. П. П. Олейника. Ростов-на-Дону: ШИ (ф) ЮРГПУ (НПИ) им. М. И. Платова, 2015. URL: http://objectsystems.ru/files/2015/Object_Systems_2015_Proceedings.pdf

P. P. Oleynik, PhD, System Architect Software, Aston JSC, Associate Professor, Shakhty Institute (branch) of Platov South Russian State Polytechnic University (NPI), Russia, Rostov-on-Don, xsl@list.ru,

V. I. Gurianov, PhD in Technical Science, Assistant Professor, Saint-Petersburg State Economic University, Branch in Cheboksary, Russia, Cheboksary, vg2007sns@rambler.ru

UML-Profile for Metamodel-Driven Design of Database Applications

Currently, most of the newly developed applications are database applications. At the same time for the functionality of the system most commonly used object-oriented programming languages that allow you to create re-usable pieces of code, which is achieved with the use of encapsulation, inheritance and polymorphism. Further development of the object-oriented approach is the extension of its principles for the development of the entire application. As a result, the developer provides a single development environment that allows you to create a target application. At the same time the developer tasked correct design and construction of domain models in terms of the selected tool.

This article presents a UML-profile that repeatedly used by the authors in the development of database applications in their own development environment. This profile allows you to metamodelno-oriented application design and describes the domain model in terms of the development of object metamodel. At the end of the article provides examples of the use of the profile and draw conclusions on ways of working.

Keywords: UML, UML Profile, Databases, Object-Oriented Design

References

1. **The UML and Data Modeling**, White Paper, Technical report, Rational Software, 2003.
2. **Object Management Group. UML 2.0 Infrastructure Specification**, OMG document number formal/2011-08-05, 2011.
3. **Brdjanin D., Marie S.** An Example of Use-Case-driven Conceptual Design of Relational Database, *EUROCON, 2007. The International Conference on "Computer as a Tool"*, 9–12 Sept. 2007, Warsaw, pp. 538–545. DOI: 10.1109/EURCON.2007.4400437
4. **Ritala T., Kuikka S.** UML Automation Profile: Enhancing the Efficiency of Software Development in the Automation Industry, *Proc. of 5th IEEE International Conference on Industrial Informatics, 23–27 June 2007, Vienna*, pp. 885–890. DOI: 10.1109/INDIN.2007.4584890
5. **Vepsalainen T., Hastbacka D., Kuikka S.** Tool Support for the UML Automation Profile — For Domain-Specific Software Development in Manufacturing, *Proc. of The Third International Conference on Software Engineering Advances, ICSEA '08, 26–31 Oct. 2008, Sliema*, pp. 43–50. DOI: 10.1109/ICSEA.2008.22
6. **Hengye Zhu, Guangyao Li, Liping Aheng.** A UML profile for HLA-based simulation system modelin, *Proc. of 6th IEEE International Conference on Industrial Informatics, INDIN 2008, 13–16 July 2008, Daejeon*, pp. 1602–1607. DOI: 10.1109/INDIN.2008.4618360
7. **Idoudi N., Duvallet C., Bouaziz R., Sadeg B., Gargouri F.** How to Model a Real-Time Database? *Proc. of IEEE International Symposium on Object/Component/Service-Oriented Real/Time Distributed Computing, ISORC '09, 17–20 March 2009, Tokyo*, pp. 321–325, DOI: 10.1109/ISORC.2009.17
8. **Xudong Song, Xiaolan Yan, Liguang Yang.** Design ETL Metamodel Based on UML Profile, *Second International Symposium on Knowledge Acquisition and Modelibg, KAM'09, Nov. 30, 2009-Dec. 1 2009, Wuhan*, pp. 69–72. DOI: 10.1109/KAM.2009.112
9. **Torres A., Galante R., Pimenta M. S.** Towards a UML Profile for Model-Driven Object-Relational Mapping, *XXIII Brazilian Symposium on Software Engineering, SBES '09, 5–9 Oct., Fortaleza, Ceara*, pp. 94–103. DOI: 10.1109/SBES.2009.22
10. **Ferreira T. B., Stempluc S. M., Lisboa-Filho J.** Geographical data modeling with the UMLGeoProfile and MDA transformations on the Enterprise Architect Tool, *9th Iberian Conference on Information Systems and Technologies (CISTI), 18–21 June 2014, Barcelona*, pp. 1–6. DOI: 10.1109/CISTI.2014.6876987
11. **Mubin S. A., Jantan A. H.** A UML 2.0 profile web design framework for modeling complex web application., *International Conference on Information Technology and Multimedia (ICIMU), 18–20 Nov. 2014, Putrajaya*, 324–329 pp., DOI: 10.1109/ICIMU.2014.7066653
12. **Usman M., Iqbal M. Z., Khan M. U.** A Model-Driven Approach to Generate Mobile Applications for Multiple Platforms, *21st Asia-Pacific Software Engineering Conference (APSEC), 1–4 Dec. 2014, Jeju*, pp. 111–118. DOI: 10.1109/APSEC.2014.26
13. **Gur'yanov V. I.** Modelirovanie klassifikatsiy na vizual'nom yazyke imitatsionnogo modelirovaniya UML SP, *Sbornik dokladov Shestoy vsrossiyskoy nauchno-prakticheskoy konferentsii "Imitatsionnoe modelirovanie. Teoriya i praktika" (IMMOD-2013), vol. 1, Kazan*, Izdatel'stvo "FEN" Akademii nauk RT, 2013, pp. 128–132.
14. **Gur'yanov V. I.** *Imitatsionnoe modelirovanie na UML SP*. Cheboksary, Filial SPbGUEU v. g. Cheboksary, 2014. 135 p.
15. **Evans E.** *Predmetno-orientirovannoe proektirovame (DDD). Strukturizatsiya slozhnykh programmnykh sistem*, Moscow, Vil'yams, 2010, 423 p.
16. **Oleynik P. P.** *Programma dlya EVM "Unifitsirovannaya sreda bystroy razrabotki korporativnykh informatsionnykh sistem SharpArchitect RAD Studio"*, svidetel'stvo o gosudarstvennoy registratsii № 2013618212 ot 04 sent. 2013 g.
17. **Oleynik P. P.** Elementy srede razrabotki programmnykh kompleksov na osnove organizatsii metamodeli ob'ektnoy sistemy, *Biznes-informatika*, 2013, no. 4 (26), pp. 69–76. URL: [http://bijournal.hse.ru/data/2014/01/16/1326593606/1B1%204\(26\)%202013.pdf](http://bijournal.hse.ru/data/2014/01/16/1326593606/1B1%204(26)%202013.pdf)
18. **Oleynik P. P., Kurakov Yu. I.** Kontseptsiya sozdaniya obsluzhivayushchey korporativnoy informatsionnoy sistemy ekonomicheskogo proizvodstvenno-energeticheskogo klastera, *Prikladnaya informatika*, 2014, no. 6, pp. 5–23.
19. **Oleynik P. P., Nikolenko O. I., Yuzefova S. Yu.** Information System for Fast Food Restaurants, *Engineering and Technology*, vol. 2015, 2, no. 4, pp. 186–191., <http://article.aascit.org/file/pdf/9020895.pdf>
20. **Oleynik P. P.** Metamodel-Driven Design of Database Applications, *Journal of Computer Science Technology Updates*, 2015, vol. 2, no. 1, pp. 15–24. URL: <http://www.cosmoscholars.com/images/JCSTU-v1n1/JCSTU-V2-N1/JCSTU-V2N1A3-Oleynik.pdf>
21. **Oleynik P. P.** Predmetno-orientirovannoe proektirovanie struktury bazy dannykh v ponyatiyakh metamodeli ob'ektnoy sistemy, *Ob'ektnye sistemy — 2014: materialy VIII Mezhdunarodnoy nauchno-prakticheskoy konferentsii, Rostov-na-Donu, 10–12 maya 2014 g.*, Pod obshch. red. P. P. Oleynika, Rostov-na-Donu: ShI (f) YuRGPU (NPI) im. M. I. Platova, 2014, pp. 41–46, URL: http://objectsystems.ru/files/2014/Object_Systems_2014_Proceedings.pdf
22. **Oleynik P. P.** Using metamodel of object system for domain-driven design the database structure, *Proc. of 12th IEEE East-West Design & Test Symposium (EWDTS'2014), Kiev, Ukraine, September 26–29, 2014*, DOI: 10.1109/EWDTS.2014.7027052
23. **Gur'yanov V. I., Oleynik P. P.** UML-profil' proektirovaniya struktury ob'ektno-orientirovannoy bazy dannykh, *Ob'ektnye sistemy — 2015: materialy X Mezhdunarodnoy nauchno-prakticheskoy konferentsii, Rostov-na-Donu, 10–12 maya 2015*, Pod obshch. red. P. P. Oleynika, Rostov-na-Donu, ShI (f) YuRGPU (NPI) im. M. I. Platova, 2015, http://objectsystems.ru/files/2015/Object_Systems_2015_Proceedings.pdf
24. **Oleynik P. P.** Unifitsirovannaya model' testirovaniya instrumentov razrabotki ob'ektno-orientirovannykh prilozheniy, *Ob'ektnye sistemy — 2014 (Zimnyaya sessiya): materialy IX Mezhdunarodnoy nauchno-prakticheskoy konferentsii, Rostov-na-Donu, 10–12 dekabrya 2014 g.*, Pod obshch. red. P. P. Oleynika, Rostov-na-Donu, ShI (f) YuRGPU (NPI) im. M. I. Platova, 2014, pp. 25–35. URL: http://objectsystems.ru/files/2014WS/Object_Systems_2014_Winter_session_Proceedings.pdf
25. **Nikolenko O. I., Oleynik P. P., Yuzefova S. Yu.** Prototirovanie i realizatsiya graficheskoy formy zakaza dlya informatsionnoy sistemy restorana bystrogo pitaniya, *Ob'ektnye sistemy — 2015: materialy X Mezhdunarodnoy nauchno-prakticheskoy konferentsii, Rostov-na-Donu, 10–12 maya 2015 g.*, Pod obshch. red. P. P. Oleynika, Rostov-na-Donu, ShI (f) YuRGPU (NPI) im. M. I. Platova, 2015, URL: http://objectsystems.ru/files/2015/Object_Systems_2015_Proceedings.pdf

П. Н. Бибилло, д-р техн. наук, проф., зав. лаб., И. П. Логинова, канд. техн. наук, ст. науч. сотр.
Объединенный институт проблем информатики НАН Беларуси

Формирование энергоемких тестов для комбинационных логических схем по результатам оценки их энергопотребления

Предлагается формализация задачи и алгоритмы нахождения тестовых векторов, которые обеспечивают режим максимального энергопотребления комбинационной логической схемы, синтезированной в базе проектирования заказной КМОП СБИС. Эксперименты показали, что выбор одной десятой части "энергоемких" наборов для комбинационных схем позволяет примерно на 50–70 % повысить энергопотребление схем при повторном моделировании на найденных тестах.

Ключевые слова: комбинационные КМОП-схемы, моделирование, псевдослучайные тестовые векторы, оценка энергопотребления

Введение

Сокращение энергопотребления электронных логических схем является комплексной проблемой и требует решения в различных аспектах [1]. Одним из таких аспектов является оценка энергопотребления схемы в режиме повышенного энергопотребления. Получение логических схем выполняется с помощью синтезаторов [2], которые превращают высокоуровневые описания алгоритмов функционирования схем на языке VHDL (либо Verilog) в структурные описания логических схем. Приближенная оценка энергопотребления логических КМОП-схем осуществляется моделированием таких описаний в системах логического моделирования [3, 4]. Более точная оценка энергопотребления проекта логической схемы выполняется в схемотехнических системах аналогового моделирования. И в том и в другом случае для каждой пары (такта) сменяемых значений входных воздействий проектировщики получают значение энергии, потребляемой схемой в данном такте. Таким образом, начальное моделирование позволяет получить для каждой пары тестовых воздействий (тестовых векторов — двоичных входных наборов) некоторое число, определяющее общее (динамическое и статическое) потребление, возникающее в результате переключений транзисторов, входящих в логические КМОП-элементы схемы.

В данной работе описывается формализация задачи нахождения энергоемкого теста для комбинационных логических КМОП-схем, и предлагаются два алгоритма нахождения заданного числа k тестовых векторов, использование которых при повторном моделировании таких схем обеспечивает режим их возможно большего энергопотребления. Выбор тестовых векторов осуществляется по результатам начального моделирования логической схемы. Результаты начального моделирования представляют собой последовательность двоичных на-

боров, причем для каждой пары соседних наборов (такта срабатывания и энергопотребления схемы) указывается число — потребляемая энергия в данном такте.

Первый алгоритм оперирует на множестве всех возможных пар тестовых векторов и предназначен для схем небольшой размерности, второй алгоритм обрабатывает псевдослучайные последовательности тестовых векторов, и его можно применять для схем произвольной размерности. Приводятся результаты экспериментов по нахождению энергоемких тестов для комбинационных логических схем из библиотечных КМОП-элементов [3].

2. Формализация и решение задачи 1 нахождения теста

Пусть задана комбинационная логическая схема R , имеющая n входов x_1, x_2, \dots, x_n . Булево пространство V^X над переменными вектора $\mathbf{x} = (x_1, x_2, \dots, x_n)$ содержит 2^n двоичных наборов \mathbf{x}_i^* . Каждому двоичному набору $\mathbf{x}_i^* \in V^X$ поставим в соответствие число i , равное десятичному эквиваленту этого набора. Если при моделировании схемы R входной набор i сменяется входным набором j , то упорядоченной паре $\langle i, j \rangle$ соответствует некоторое число $S_{i, j}$, выражающее энергопотребление, например значение потребляемого схемой тока. Будем считать, что значения всех переменных входного набора подаются одновременно в очередном такте работы схемы. Сформулируем формальную постановку задачи нахождения теста T заданной длины, обеспечивающего максимальное суммарное энергопотребление, при условии, что при моделировании оценены все возможные упорядоченные пары входных тестовых наборов [5].

Пусть задано множество V чисел: $V = \{0, 1, 2, \dots, 2^n - 1\}$. Рассмотрим множество L всех $2^n (2^n - 1)$ упорядоченных пар $\langle i, j \rangle$, составленных из элементов множества V . Каждой паре $\langle i, j \rangle$ соответствует

неотрицательное целое число $S_{i,j}$ — вес пары $\langle i, j \rangle$. Каждой упорядоченной последовательности

$$P = \langle i_1, i_2, i_3, i_4, \dots, i_{k-2}, i_{k-1}, i_k \rangle \quad (1)$$

элементов (необязательно различных) множества V соответствует множество

$$\langle i_1, i_2 \rangle, \langle i_2, i_3 \rangle, \langle i_3, i_4 \rangle, \dots, \langle i_{k-2}, i_{k-1} \rangle, \langle i_{k-1}, i_k \rangle \quad (2)$$

упорядоченных пар, составленных из соседних элементов последовательности (1). Правильной k -последовательностью назовем такую упорядоченную последовательность (1), что все упорядоченные пары вида (2) являются различными. Заметим, что для $P = \langle i_q, i_{q+1}, i_q \rangle$ получаем две различные упорядоченные пары $\langle i_q, i_{q+1} \rangle, \langle i_{q+1}, i_q \rangle$. Такие последовательности определяют пиковое энергопотребление для двух тактов.

Задача нахождения теста T имеет следующую формальную постановку.

Задача 1. Для заданного числа k требуется составить из элементов множества L правильную k -последовательность P с максимальной суммой весов

$$S = \sum_{q=2}^{k-1} (S_{i_{q-1}, i_q} + S_{i_q, i_{q+1}}). \quad (3)$$

Пример 1. Результаты моделирования энергопотребления логической схемы заданы в табл. 1, где в левой части таблицы заданы наборы из множества L , в правой части — веса $S_{i,j}$, соответствующие упорядоченным парам. На практике в качестве весов обычно используют вещественные числа, получаемые программы схемотехнического моделирования. Далее в примерах для удобства будем использовать в качестве весов целые числа. Например, для первой пары $\langle 111, 101 \rangle = \langle 7, 5 \rangle$ вес $S_{7,5} = 44$, для последней пары $\langle 110, 111 \rangle = \langle 6, 7 \rangle$ вес $S_{6,7} = 14$. Среднее потребление (средний вес) на всех возможных парах наборов составляет $S_{\text{ср}} = 53,21$ (см. табл. 1).

Если каждому элементу множества V поставить в соответствие вершину полного ориентированного графа G , то задача может быть переформулирована в графовой постановке: в полном ориентированном графе G , дуги которого взвешены неотрицательными целыми числами, требуется найти простую цепь M , состоящую из $k - 1$ дуг и имеющую максимальную сумму S весов, входящих в нее (т. е. в цепь) дуг. Данная задача и алгоритмы ее решения хорошо известны в теории графов [6].

Табл. 1 может быть интерпретирована как задание весов дуг полного ориентированного графа G и переписана в виде табл. 2: элемент $\langle i, j \rangle$ (строка i , столбец j) матрицы смежности полного ориентированного графа G задает вес $S_{i,j}$ дуги, исходящей из вершины i и заходящей в вершину j . Заметим, что в данном примере $S_{i,j} = S_{j,i}$, а значения весов получены в результате подсчета числа переключе-

Таблица 1

Окончание табл. 1

Входные наборы (тест)	Потребляемая энергия (по тактам) $S_{i,j}$	Входные наборы (тест)	Потребляемая энергия (по тактам) $S_{i,j}$
111	0	001	56
101	44	101	56
111	44	000	70
100	62	101	70
111	62	100	26
011	72	010	34
111	72	100	34
010	88	001	62
111	88	100	62
001	72	000	44
111	72	100	44
000	82	011	62
111	82	001	32
110	14	011	32
100	52	000	46
110	52	011	46
011	86	010	28
110	86	000	18
010	78	010	18
110	78	001	36
001	82	000	18
110	82	001	18
000	72	010	36
110	72	011	28
101	58	100	62
011	36	101	26
101	36	110	58
010	60	111	14
101	60	$S_{\text{ср}}$	53.21

Таблица 2

i	j							
	0	1	2	3	4	5	6	7
0		18	18	48	44	70	72	82
1	18		36	32	62	56	82	72
2	18	36		28	34	60	78	88
3	46	32	28		62	36	86	72
4	44	62	34	62		26	52	62
5	70	56	60	36	26		58	44
6	72	82	78	86	52	58		14
7	82	72	88	72	62	44	14	

ний транзисторов в каждом такте моделирования энергопотребления комбинационной логической схемы, имеющей три входа.

Алгоритм 1 решения задачи 1 является "жадным".

Для каждой вершины графа G находится одна ориентированная цепь, состоящая из $k - 1$ дуг и имеющая максимальный суммарный вес. Весом цепи является сумма весов входящих в цепь дуг. Формирование цепи тоже осуществляется "жадным" алгоритмом, каждый раз в цепь добавляется дуга, имеющая максимальный вес.

В рассматриваемом примере полного графа для $k = 7$ найдена цепь из шести дуг (упорядоченных пар)

$$\langle 1,6 \rangle \rightarrow \langle 6,3 \rangle \rightarrow \langle 3,6 \rangle \rightarrow \langle 6,2 \rangle \rightarrow \langle 2,7 \rangle \rightarrow \langle 7,2 \rangle,$$

со следующими весами дуг: $S_{1,6} = 82$, $S_{6,3} = 86$, $S_{3,6} = 86$, $S_{6,2} = 78$, $S_{2,7} = 88$, $S_{7,2} = 88$. Вес найденной цепи равен

$$S_{1,6} + S_{6,3} + S_{3,6} + S_{6,2} + S_{2,7} + S_{7,2} = 82 + 86 + 86 + 78 + 88 + 88 = 508.$$

Таким образом, найденный энергоемкий тест, состоящий из семи наборов, имеет следующий вид:

001
110
011
110
010
111
010.

Среднее потребление энергии на полученном энергоемком тесте составляет: $S_{\text{ср}}^{\text{алг 1}} = 508:6 = 84,6$ (условных единиц потребляемой энергии), что значительно больше, чем среднее энергопотребление $S_{\text{ср}} = 53,21$ при начальном моделировании.

Если начальное моделирование проведено не на всех парах входных наборов, то граф G не является полным. Например, для табл. 3 (пример 2), задающей результаты моделирования на тесте из 32 входных наборов, соответствующий граф представлен на рис. 1. В результате применения алгоритма 1 для $k = 7$ найдена цепь (рис. 2) из шести дуг (упорядоченных пар):

$$\langle 5,0 \rangle \rightarrow \langle 0,7 \rangle \rightarrow \langle 7,2 \rangle \rightarrow \langle 2,6 \rangle \rightarrow \langle 6,3 \rangle \rightarrow \langle 3,4 \rangle,$$

со следующими весами дуг: $S_{5,0} = 70$, $S_{0,7} = 82$, $S_{7,2} = 88$, $S_{2,6} = 78$, $S_{6,3} = 86$, $S_{3,4} = 62$.

Вес найденной цепи равен

$$S_{5,0} + S_{0,7} + S_{7,2} + S_{2,6} + S_{6,3} + S_{3,4} = 70 + 82 + 88 + 78 + 86 + 62 = 466.$$

Таким образом, найденный энергоемкий тест, состоящий из семи наборов, имеет следующий вид:

101
000
111
010
110
011
100.

Среднее потребление энергии на полученном энергоемком тесте составляет: $S_{\text{ср}}^{\text{алг 1}} = 466:6 = 77,6$ (условных единиц потребляемой энергии).

Заметим, что алгоритм 1 для данного примера находит в графе G цепь заданной длины, однако при достаточно больших значениях числа входных переменных логической схемы и последовательности псевдослучайных тестовых наборов такая цепь, как правило, алгоритмом 1 не находится. Заметим, что граф G является цепью, если тестовые наборы не повторяются.

Таблица 3

Окончание табл. 3

Входные наборы (тест)	Потребляемая энергия (по тактам) $S_{i,j}$	Входные наборы (тест)	Потребляемая энергия (по тактам) $S_{i,j}$
101	0	101	36
111	44	101	0
100	62	000	70
111	62	100	44
101	44	000	44
000	70	000	0
001	18	111	82
011	32	110	14
000	46	011	86
001	18	010	28
011	32	000	18
100	62	100	44
110	52	101	26
111	14	010	60
010	88	100	34
110	78		
011	86	$S_{\text{ср}}$	43.56

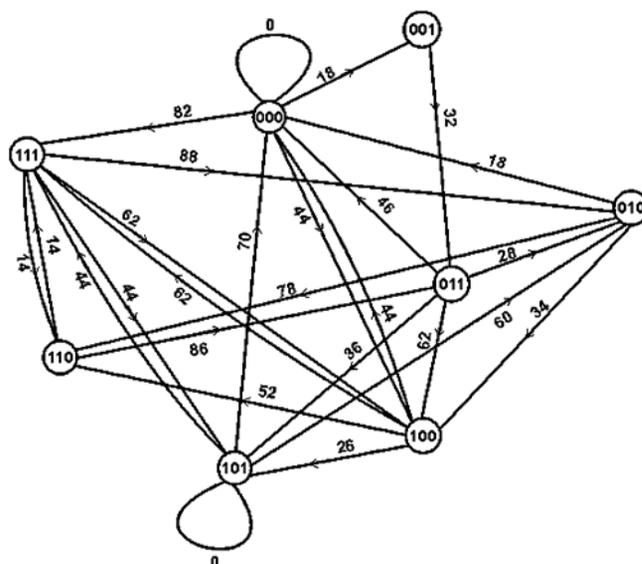


Рис. 1. Ориентированный граф $G(V, L^4)$, соответствующий табл. 3

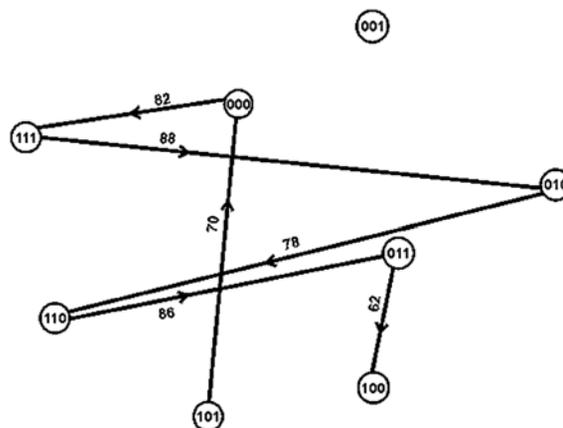


Рис. 2. Ориентированный граф $G(V, L^4)$ с выделенными дугами, соответствующими найденному алгоритмом 2 тесту

3. Формализация и решение задачи 2 нахождения теста

Следующая задача возникает после проведения оценки энергопотребления комбинационной схемы на последовательности псевдослучайных тестовых векторов, в этом случае не для всех упорядоченных пар входных наборов известны их веса, так как некоторые пары входных наборов не были использованы при начальном моделировании.

Задача 2. Задана упорядоченная последовательность из m случайно выбранных наборов из множества V , для каждой пары $\langle i_{q-1}, i_q \rangle$ наборов указан ее вес $S_{q-1, q}$. Требуется найти правильную k -последовательность ($k < m$) P с максимальной суммой весов.

Пусть псевдослучайная последовательность задана в табл. 3, для упорядоченной пары $\langle 010, 111 \rangle$ нет значения веса, поэтому если эта пара войдет в результирующую последовательность (т. е. в решение задачи 2), то только при повторном моделировании будет определен ее вклад в энергопотребление. Так как для некоторых (а на практике очень многих) пар будут отсутствовать веса, то получаемая в результате решения задачи 2 тестовая последовательность обеспечит больший суммарный вес, чем тот, который определяется найденными парами, так как веса некоторых пар будут определены при повторном моделировании. Табл. 3 соответствует табл. 4, задающая веса дуг графа G .

Алгоритм 2 решения задачи 2 тоже является "жадным".

Все дуги графа G упорядочиваются по убыванию веса. Для каждой вершины графа находится ориентированная цепь, которая обязательно содержит первых $k/2$ дуг упорядоченной последовательности и имеющих максимальный вес. Формирование цепи осуществляется "жадным" алгоритмом, каждый раз в цепь добавляется дуга, либо имеющая максимальный вес из $k/2$ упорядоченной последовательности, либо позволяющая увеличить длину цепи, но не включенная в L^q (множество взвешенных дуг). Весом цепи является сумма весов входящих в цепь дуг, за исключением тех дуг, веса которых не были определены при начальном моделировании.

В результате применения алгоритма 2 для $k = 7$ найдена цепь из шести дуг (упорядоченных пар) следующего вида:

$$\langle 4,7 \rangle \rightarrow \langle 7, 2 \rangle \rightarrow \langle 2, 6 \rangle \rightarrow \langle 6,3 \rangle \rightarrow \langle 3, 0 \rangle \rightarrow \langle 0, 7 \rangle,$$

со следующими весами дуг: $S_{4,7} = 62$, $S_{7,2} = 88$, $S_{2,6} = 78$, $S_{6,3} = 86$, $S_{3,0} = 46$, $S_{0,7} = 82$. Вес найденной цепи равен

$$\begin{aligned} S_{4,7} + S_{7,2} + S_{2,6} + S_{6,3} + S_{3,0} + S_{0,7} = \\ = 62 + 88 + 78 + 86 + 46 + 82 = 442. \end{aligned}$$

Таким образом, найденный алгоритмом 2 энергоемкий тест, состоящий из семи наборов, имеет следующий вид:

100
111
010
110
011
000
111.

Среднее потребление энергии на полученном энергоемком тесте составляет: $S_{\text{ср}}^{\text{алг 2}} = 442:6 = 73,6$ (условных единиц потребляемой энергии). Заметим, что среднее потребление на исходном тесте составляет $S_{\text{ср}} = 43,56$ (см. табл. 3). В рассмотренном примере число $n = 3$, и в графе нашлась цепь, состоящая из шести дуг.

Приведем **пример 3** (табл. 5), когда граф G представляет собой цепь, так как все псевдослучайные тестовые наборы являются различными.

Таблица 4

i	j							
	0	1	2	3	4	5	6	7
0		18	0	0	44	0	0	82
1	0		0	32	0	0	0	0
2	18	0		28	34	0	78	0
3	46	0	28		62	36	0	0
4	44	0	0	0		26	52	62
5	70	0	60	0	0		0	44
6	0	0	0	86	0	0		14
7	0	0	88	0	62	44	14	

Таблица 5

Входные наборы (тест)	Потребляемая энергия (по тактам) $S_{i,j}$
101010101	0
101011011	42
100000010	126
111010100	224
000110110	126
000001010	140
010011100	140
000111101	112
001000110	70
101111101	126
100110010	154
000100011	70
110110000	182
100011111	140
001000111	84
110110010	140
001111111	140
011101101	112
100111001	140
001001110	126
$S_{\text{ср}}$	126

Алгоритм 2 для $k = 7$ нашел следующее решение:

$\langle 101011011, 100000010 \rangle \rightarrow \langle 100000010, 111010100 \rangle \rightarrow$
 $\rightarrow \langle 111010100, 000100011 \rangle \rightarrow \langle 000100011, 110110000 \rangle \rightarrow$
 $\rightarrow \langle 110110000, 101111101 \rangle \rightarrow \langle 101111101, 100110010 \rangle.$

Запишем данное решение в сокращенном виде, используя вместо двоичных наборов их числовые эквиваленты:

$\langle 347, 258 \rangle \rightarrow \langle 258, 468 \rangle \rightarrow \langle 468, 35 \rangle \rightarrow \langle 35, 432 \rangle \rightarrow$
 $\rightarrow \langle 432, 381 \rangle \rightarrow \langle 381, 306 \rangle,$

со следующими весами дуг: $S_{347,258} = 126$, $S_{258,468} = 224$, $S_{468,35} = \text{"-"}$, $S_{35,432} = 182$, $S_{432,381} = \text{"-"}$, $S_{381,306} = 154$. Веса отсутствующих дуг обозначены через "-". Среднее значение потребления до повторного моделирования

$$(S_{347,258} + S_{258,468} + S_{35,432} + S_{381,306}) : 6 = 126 + 224 + 182 + 154 = 686 : 6 = 114,3.$$

В данном примере среднее значение энергопотребления до повторного моделирования меньше, чем 126 — среднее значение на исходном тесте. Однако на практике, как показал эксперимент, часто бывает иначе, а именно: среднее значение энергопотребления до повторного моделирования может быть больше среднего на исходном тесте. При повторном моделировании с найденным энергоемким тестом

101011011
 100000010
 111010100
 000100011
 110110000
 101111101
 100110010

выяснились веса отсутствующих в начальном эксперименте пар наборов: $S_{468,35} = 154$, $S_{432,381} = 154$. Суммарный вес энергоемкого теста равен

$$S_{347,258} + S_{258,468} + S_{468,35} + S_{35,432} + S_{432,381} + S_{381,306} = 126 + 224 + 154 + 182 + 154 + 154 = 994.$$

Среднее значение потребления на энергоемком тесте $S_{\text{ср}}^{\text{алг 2}} = 994 : 6 = 165,7$.

4. Эксперимент

Проведенный эксперимент состоял из выполнения пяти этапов и заключался в схемной реализации систем ДНФ булевых функций F логическими схемами из библиотечных логических элементов, оценке их потактового энергопотребления на исходных и энергоемких тестах. Энергоемкие тесты строили с помощью описанных ранее алгоритмов 1 и 2. Перечислим этапы эксперимента по получению и оценке энергоемких тестов.

Этап 1. Синтез комбинационных схем по VHDL-представлениям систем булевых функций.

Этап 2. Построение начальных тестов длины $K_{\text{исх}}$ (входных наборов) трех типов (T_1, T_2, T_3).

Этап 3. Оценка потактового энергопотребления полученных комбинационных схем с помощью VHDL-моделирования на начальных тестах, получение среднего значения $S_{\text{ср}}$ энергопотребления на начальных тестах.

Этап 4. Выполнение программ, реализующих алгоритмы 1 и 2, и получение энергоемких тестов.

Этап 5. Повторное моделирование схем на энергоемких тестах, получение средних значений $S_{\text{ср}}^{\text{алг 1}}$, $S_{\text{ср}}^{\text{алг 2}}$ энергопотребления на энергоемких тестах.

Исходными данными явились 19 примеров систем ДНФ полностью определенных булевых функций из набора примеров [7]. В качестве промышленной системы синтеза логических схем во всех экспериментах использовали синтезатор Leonardo-Spectrum (версия 2010a.7). В качестве целевой библиотеки синтеза использовали отечественную библиотеку [3] проектирования заказных цифровых КМОП СБИС. В качестве числа k задавалось число, равное примерно одной десятой от общего числа наборов при начальном моделировании. Например, для первой строки табл. 6 (пример b12) $K_{\text{исх}} = 512$, $k = K_1 = K_2 = 54$.

Результаты эксперимента представлены в табл. 6, где S — площадь схемы в условных единицах; n — число входов схемы; T — тип теста:

T_1 — тестовые наборы с равновероятными значениями нулей и единиц, т. е. на любом входе логической схемы для любого входного набора вероятность появления единицы (нуля) равна 0,5 (псевдослучайные наборы);

T_2 — тестовые наборы, состоящие из упорядоченной по возрастанию десятичного эквивалента последовательности из 2^n всех наборов булева пространства размерности n (левая часть таблицы истинности системы полностью определенных булевых функций);

T_3 — специальная последовательность тестовых наборов, генерируемая с помощью алгоритма из работы [8] и задающая все упорядоченные пары входных наборов из булева пространства размерности n ;

$K_{\text{исх}}$ — число наборов исходного теста;

K_1 — число наборов теста, полученного алгоритмом 1;

K_2 — число наборов теста, полученного алгоритмом 2;

$S_{\text{ср}}^{\text{алг 1}}$ — среднее потребление на энергоемком тесте, полученном алгоритмом 1;

$S_{\text{ср}}^{\text{алг 2}}$ — среднее потребление на энергоемком тесте, полученном алгоритмом 2.

Таблица 6

Имя	S	n	T	Исходный тест		Алгоритм 1			Алгоритм 2		
				$K_{исх}$	$S_{ср}$	K_1	$S_{ср}^{алг1}$	%	K_2	$S_{ср}^{алг2}$	%
b12	19826	15	T1	512	9,18e-8	54	9,94e-8	108,30	54	13,0e-8	142,3
C1355	96065	41	T1	4096	2,24e-7				412	2,78e-7	124,11
CHT	47770	47	T1	4096	7,64e-8				412	1,68e-7	219,90
FRG2	299501	143	T1	4096	1,18e-6				412	1,61e-6	136,4
I8	205684	133	T1	4096	4,29e-7				412	5,26e-7	122,61
C1908	92522	33	T1	4096	2,21e-7				412	3,11e-7	140,72
Life	15178	9	T1	256	1,18e-7	28	1,48e-7	125,42	28	1,49e-7	126,27
Life	15178	9	T2	512	4,34e-8	54	4,90e-8	112,90	54	1,29e-7	297,24
C432	43357	36	T1	4096	2,01e-7				412	2,98e-7	148,26
FRG1	40154	28	T1	4096	2,12e-7				412	3,29e-7	155,19
C880	25808	60	T1	4096	8,69e-8				412	1,24e-7	142,69
C499	46917	41	T1	4096	6,19e-11				412	76,3e-011	893,4
I3	30545	132	T1	4096	2,35e-11				412	2,83e-011	120,43
I9	32236	88	T1	512	1,11e-7	54	1,29e-7	116,22	54	2,34e-7	210,81
DALU	124071	75	T1	1024	4,98e-7	104	5,48e-7	110,04	104	7,29e-7	146,39
CM42A	1830	4	T1	64	1,16e-8	8	1,80e-8	155,17	8	1,61e-8	138,79
CM42A	1830	4	T3	241	1,49e-8	26	2,81e-8	188,59	26		
CM82A	3437	5	T1	128	2,22e-8	14	3,99e-8	179,73	14	3,11e-8	140,09
CM82A	3437	5	T3	993	2,18e-8	102	4,21e-8	193,12	102		
CM85A	4994	11	T2	2048	2,09e-8	206	2,88e-8	137,80	206	3,99e-8	190,91
CM138A	1099	6	T1	2048	3,25e-13	206	1,01e-12	310,77	206	5,76e-13	177,23
CM162A	6283	14	T1	2048	3,04e-8	206	3,25e-8	106,91	206	3,74e-8	123,03
INTB	444709	15	T1	1024	1,31e-6	104	1,36e-6	106,82	104	1,59e-6	121,37
Среднее							Алгоритм 1	152,99	Алгоритм 2		191,34
									Алгоритм 2 (без учета C499)		156,24

Обсуждение результатов эксперимента

Анализ представленных в табл. 6 и других полученных экспериментальных результатов позволяет сделать следующие выводы.

1. Использование начальных полных тестов типа $T3$ позволяет выбрать более энергоемкие тесты, чем использование начальных тестов типа $T1$ и псевдослучайных тестов типа $T2$.

2. Алгоритм 1 является достаточно трудоемким алгоритмом и пытается найти среди имеющихся пар входных наборов наиболее энергоемкие тесты, однако он неприменим для тестов из псевдослучайных наборов.

3. Алгоритм 2 является быстродействующим, применимым для задач произвольной размерности и достаточно эффективным, однако, выбирая наиболее энергоемкие пары, он не может предсказать потребление на отсутствующих при начальном моделировании парах наборов, хотя достаточно часто обеспечивает увеличение на 20 % среднего энергопотребления даже без повторного моделирования.

4. Выбор одной десятой части (10 %) энергоемких наборов для комбинационных схем позволяет примерно на 50–70 % повысить энергопотребление схем на найденных коротких энергоемких тестах.

Заключение

Разработанные программы формирования энергоемких тестов включены в систему [9] логического синтеза КМОП-схем и используются для оценки вариантов схем в целях получения проектов КМОП СБИС, характеризующихся пониженным энергопотреблением. Уменьшение числа наборов в энергоемком тесте приводит к увеличению среднего значения потребляемого схемой тока, что, в свою очередь, важно для определения минимальной ширины проводников в сетях питания и заземления в СБИС. Правильно выбранная ширина таких проводников важна для предотвращения эффектов электромиграции [1], приводящих к разрыву проводников и сбоям функционирования СБИС.

Список литературы

1. Рабаи Ж. М., Чандракасан А., Николич Б. Цифровые интегральные схемы, 2-е изд.: Пер. с англ. М.: Вильямс, 2007. 912 с.
2. Библо П. Н. Системы проектирования интегральных схем на основе языка VHDL. StateCAD, ModelSim, LeonardoSpectrum. М.: СОЛОН-Пресс, 2005. 384 с.
3. Библо П. Н., Кириенко Н. А. Оценка энергопотребления логических КМОП-схем по их переключательной активности // Микроэлектроника. 2012. № 1. С. 65–77.

4. Ghosh A., Devadas S., Keutzer K., White J. Estimation of Average Switching Activity in Combinational and Sequential Circuits // Proc. 29th ACM/IEEE Design Automation Conference, June, Anaheim, CA. 1992. P. 253—259.

5. Бибилло П. Н. Нахождение теста для режима максимального энергопотребления комбинационной логической схемы // Управляющие системы и машины. 2010. № 5. С. 39—45.

6. Харари Ф. Теория графов. М.: Мир, 1973. 300 с.

7. <http://www1.cs.columbia.edu/~cs6861/sis/espresso-examples/ex/>

8. Закревский А. Д. Минимизация перебора ориентированных пар. Танаевские чтения // Доклады Четвертой Международной научной конференции (29 марта 2010 г., Минск). Минск: ОИПИ НАН Беларуси, 2010. С. 58—62.

9. Бибилло П. Н., Черемисинова Л. Д., Кардаш С. Н., Кирiyенко Н. А., Романов В. И., Черемисинов Д. И. Автоматизация логического синтеза КМОП схем с пониженным энергопотреблением // Программная инженерия. 2013. № 8. С. 35—41.

P. N. Bibilo, Professor, Head of Lab., I. P. Loginova, Associate Professor, Senior Researcher
United Institute of Informatics Problem, National Academy of Sciences of Belarus, Minsk

The Creation of Energy-Intensive Tests for Combinational Logic Circuits According to the Results of Evaluation of Their Power Consumption

The decrease of energy consumption of logic circuits is a complex problem and it requires the decision in various aspects. One of these aspects is evaluation of power consumption in the mode of heightened consumption of energy. The design of logic circuits is performed using program-synthesizer that converts high level VHDL descriptions of algorithms for functioning circuits into structural description of logic circuits. The approximate evaluations of power consumption in logic CMOS circuits are implemented through modeling of such descriptions in logical simulation systems, for example in CAD ModelSim (Mentor Graphics). The more exact evaluation of power consumption in logic circuit projects are made by analog simulation systems. In both approaches, for every pair of input variable values, the designer gets the value of energy consumed by the circuit in the current time unit. This paper gives formalization of the problem of finding energy-intensive tests for combinational logic CMOS circuits. Also, it offers two algorithms for searching a given number of test vectors, using of which in logical re-simulation of such circuits provides the mode of their heightened consumption. Selection of test vectors is done by results of primary logic circuit simulation. Results of primary logic circuit simulation (both logical and analog) are represented by sequence of binary sets, each pair of neighboring sets related to the time activation being put in correspondence to power consumption of the circuit (i.e. the number equal power consumption in the time of activation is shown). The first algorithm operates on the set of all possible test vector pairs and intended for small-scale circuits. The second algorithm is fast and efficient, deals with pseudorandom sequences of input vectors and can be applied for arbitrary-dimension circuits. At the same time, when choosing the most energy-intensive pairs it cannot predict energy consumption of unused set pairs (in primary simulation), however it gives often enough up to 20—30 % increase of power average consumption even without re-simulation. The results of experiment on building of energy-intensive tests for combinational logic circuits from library CMOS gates are presented. Experiments shows that selection of average 10 % energy-intensive test sets from the whole sets of test vector pairs allows 50—70 % increasing power consumption of circuits in found energy-intensive tests by remodeling.

Keywords: combinational CMOS circuits, simulation, pseudorandom test vectors, estimating energy consumption

References

1. Raba Zh. M., Chandrakasan A., Nikolic B. *Cifrovye integral'nye shemy* (Digital integrated circuits), 2nd ed.: Moscow, Williams, 2007. 912 p. (in Russian).

2. Bibilo P. N. *Cistemy proektirovaniya integral'nyh shem na osnove yazyka VHDL. StateCAD, ModelSim, LeonardoSpectrum* (Systems of design of integrated schemes on the basis of the VHDL. StateCAD, ModelSim, LeonardoSpectrum). Moscow, SOLON Press, 2005. 384 p. (in Russian).

3. Bibilo P. N., Kiriyenko N. A. *Ocenka jenergotrebleniya logicheskikh KMOP-shem po ih pereklyuchatel'noj aktivnosti* (Estimation of energy consumption of logical CMOS schemes on their switching activity), *Mikroelektronika*, 2012, no. 1, pp. 65—77 (in Russian).

4. Ghosh A., Devadas S., Keutzer K., White J. Estimation of Average Switching Activity in Combinational and Sequential Circuits, *Proc. 29th ACM/IEEE Design Automation Conference*, June, Anaheim, CA. 1992, pp. 253—259.

5. Bibilo P. N. *Nahozhdenie testa dlja rezhima maksimal'nogo jenergotrebleniya kombinacionnoj logicheskoy shemy* (Finding of dough for the mode of the maximum energy consumption of the combinational logical scheme), *Upravljajushhie sistemy i mashiny*, 2010, no. 5, pp. 39—45 (in Russian).

6. Harare F. *Teorija grafov* (Theory of graphs). Moscow, Mir, 1973. 300 p. (in Russian).

7. <http://www1.cs.columbia.edu/~cs6861/sis/espresso-examples/ex/>

8. Zakrevsky A. D. *Minimizacija perebora orientirovannyh par* (Minimization of search of the focused couples), *Tanaevskie chtenija. Doklady Chetvertoj Mezhdunarodnoj nauchnoj konferencii*, Minsk. March 29, 2010, Minsk, OIPI NAN of Belarus, 2010. pp. 58—62 (in Russian).

9. Bibilo P. N., Cheremisinova L. D., Kardash S. N., Kiriyenko N. A., Romanov V. I., Cheremisinov D. I. *Avtomatizacija logicheskogosmeza KMOP shem s ponizhennym jenergotrebleni* (Automation of logical synthesis of CMOS of schemes with the lowered energy consumption), *Programmnyaya Ingeneria*. 2013, no. 8, pp. 35—41 (in Russian).

БЕЗОПАСНОСТЬ ИНФОРМАЦИИ CRYPTOSAFETY INFORMATION

УДК 004.7

С. М. Авдошин, канд. техн. наук, проф., руководитель департамента программной инженерии факультета компьютерных наук НИУ ВШЭ, savdoshin@hse.ru,
А. В. Лазаренко, студент, avlazarenko@edu.hse.ru
Национальный исследовательский университет "Высшая школа экономики" (НИУ ВШЭ)

Технология анонимных сетей

Представлен обзор использующихся в настоящее время анонимных сетей, построенных на основе технологии луковой маршрутизации и пиринговых сетей. Описаны ключевые особенности сетей, приведена их сравнительная характеристика.

Ключевые слова: анонимные сети, луковая маршрутизация, невидимый Интернет, оверлейные сети, пиринговые сети, слоистое шифрование

Введение

На сегодняшний день количество контента в сети Интернет растет в объеме и сложности за счет блогов, видео, музыки, персонализированных веб-страничек и программ. Веб 2.0 обеспечил людей такими технологиями, как вики, подкасты, новостные ленты, социальные сети, хостинг-сервисы и поисковые системы. До тех пор пока пользователи создают контент в открытых источниках, любого автора можно отследить, установить личность создателя, среагировать вовремя на передачу планов о совершении вооруженного нападения или минировании аэропорта. Но что если человек хочет скрыть свое авторство, передать незаметно для спецслужб какие-либо цифровые файлы и свести к минимуму свой след в глобальной сети?

В таком случае на арену выходят анонимные сети. Они позволяют анонимизировать Интернет-коммуникации, сделать сложной возможность связать участников взаимодействия (например, пользователя и веб-сервер, который он посетил).

Совершенно понятно, зачем такие сети нужны злоумышленникам, но зачем они нужны обычным людям? Наиболее очевидная причина использования инструментов для анонимизации в сети — предотвращение возможности слежки рекламными компаниями за пользователями в сети, получение доступа к заблокированным сетевым ресурсам. Правительства используют анонимные сети для разведки и слежки, а люди в странах, лишенных свободы слова, используют их для общения друг с другом.

Актуальность исследования анонимных сетей вызвана необходимостью разработки методов деанонимизации и атак на такие сети, поскольку эти сети широко используют террористы и продавцы нелегальных товаров. Правительства различных стран используют как технические механизмы противодействия анонимности сетей, финансируя

программы по кибербезопасности, так и юридические, например, ФСБ РФ готовит закон против анонимности в Интернете [1, 2].

Ниже приведены используемые в работе термины и сокращения:

AES (*Advanced Encryption Standard*) — алгоритм блочного шифрования с симметрическим ключом.

API (*Application Programming Interface*) — программное обеспечение, предоставляющее условия для взаимодействия между другими программами.

BGP (*Border Gateway Protocol*) — динамический, дистанционно-векторный протокол маршрутизации.

Cjdns — сетевой протокол и его реализация, с помощью которого можно создать масштабируемую, безопасную и простую в настройке сеть.

DARPA (*Defense Advanced Research Projects Agency*) — агентство правительства США.

DNS (*Domain Name System*) — система доменных имен, представляющая распределенную компьютерную систему преобразования символического имени в IP-адрес и наоборот.

F2F (*Friend-to-Friend*) — специфическая форма P2P сети, где пользователи могут осуществлять прямые соединения только с друзьями или пользователями, которым они доверяют.

I2P (*Invisible Internet Project*) — проект с открытым исходным кодом, пытающийся создать анонимную сеть для коммуникаций через Интернет.

ID — уникальный идентификатор.

IP (*Internet Protocol*) — коммуникационный протокол цифровых форматов сообщений, использующийся при обмене сообщениями между компьютерами в одной сети или серией соединенных сетей, использующих протокол TCP/IP.

ISP (*Internet Service Provider*) — организация, предоставляющая клиентам доступ к Интернету.

OSPF (*Open Shortest PathFirst*) — основанный на ссылках маршрутизационный протокол, исполь-

зующий алгоритм Дейкстры для вычисления кратчайшего пути между известными устройствами.

P2P (*Peer-to-Peer*) — пиринговая сеть, состоящая из группы равноправных компьютеров, каждый из которых выполняет роль узла для распространения информации в группе.

PFS (*Perfect Forward Secrecy*) — совершенная прямая секретность. Свойство кодирования данных, удостоверяющее, что сессионные ключи не скомпрометированы при компрометации одного из долговременных ключей.

QSPN (*Quantum Shortest Path Netsukuku*) — маршрутизационный алгоритм сети Netsukuku.

SDK (*Software Development Kit*) — набор инструментов, использующийся для разработки приложений, предоставляемый поставщиками железа или программного обеспечения.

SOCKS — сетевой протокол, позволяющий пересылать пакеты от клиента к серверу через прокси-сервер прозрачно (незаметно для них) и таким образом использовать сервисы за межсетевыми экранами.

SSH (*Secure Shell*) — криптографический протокол и интерфейс для запуска сетевых сервисов и безопасных сетевых коммуникаций с удаленного компьютера.

TCP (*Transmission Control Protocol*) — протокол, который предоставляет коммуникационную безопасность между клиент-серверными приложениями, которые взаимодействуют друг с другом через Интернет.

TLS (*Transport Layer Security*) — криптографический протокол, обеспечивающий защищенную передачу данных между узлами в сети Интернет.

Tor (*The Onion Router*) — свободное программное обеспечение с открытым исходным кодом, позволяющее пользователям защитить свою приватность и безопасность в Интернете.

UDP (*User Datagram Protocol*) — часть протокола Интернета, использующаяся программами, запущенными на разных компьютерах в сети.

Беспроводная ячеистая сеть (*wireless mesh network*) — коммуникационная сеть, созданная из узлов, соединенных беспроводным способом, имеющая ячеистую топологию.

ЛМ — луковая маршрутизация.

ЛП — луковый прокси.

Микс — устройство для передачи и хранения, принимающее какое-то количество сообщений фиксированной длины от нескольких источников, совершающее криптографическую трансформацию сообщений и затем передающее сообщение к следующему пункту назначения в случайном порядке.

Оверлейная сеть (*Overlay Network*) — общий случай логической сети, создаваемой поверх другой сети.

Проверка целостности (*Integrity Checking*) — проверка программ на прочность программного продукта на каждой фазе разработки.

Публичная сеть — тип сети, в которой любой пользователь имеет доступ и возможность соединения с другими сетями в Интернете.

Служба каталогов (*Directory Service*) — программные системы, которые хранят, организуют и предоставляют доступ к информационным директориям в порядке объединения ресурсов сети.

ЧМ — чесочная маршрутизация.

Эфемерный ключ — такой ключ, который создан специально для выполнения только одного распределения ключей.

Ячеистая топология (*Mesh Topology*) — сетевая топология компьютерной сети на принципе ячеек, в которой каждая рабочая станция сети соединяется с несколькими другими рабочими станциями этой же сети с возможным принятием на себя функций коммутатора для других рабочих станций.

1. Луковая маршрутизация

Так называемая луковая маршрутизация (далее — ЛМ) была разработана в середине 1990-х годов в *U.S. Naval Research Laboratory* для защиты коммуникаций в сети разведки США [3]. Впоследствии дорабатывалась компанией *Advanced Research Projects Agency*, запатентована флотом в 1998 году [4].

ЛМ является инфраструктурой общего назначения для частных коммуникаций в публичной сети. ЛМ имеет интерфейсы для стороннего программного обеспечения через специализированный прокси, что позволяет без проблем интегрировать ее с существующими системами. Первые прототипы использовали с июля 1997 года.

ЛМ работает через динамическое построение анонимных соединений, с помощью миксов Чаума [5] в реальном времени. Сеть ЛМ из луковых маршрутизаторов является распределенной и контролируемой несколькими административными доменами, так что никакой единичный луковый маршрутизатор не может разрушить всю сеть или скомпрометировать приватность пользователя.

Анонимные соединения ЛМ являются протокол-независимыми и существуют в трех фазах: установка соединения, продвижение данных, закрытие соединения. Установка начинается, когда инициатор создает так называемую луковицу, определяющую путь соединения через сеть. Под луковицей понимается рекурсивная слоистая структура данных, специфицирующая свойства соединения в каждой точке, т. е. она осуществляет криптографический контроль информации. Каждый луковый маршрутизатор на протяжении маршрута использует свой публичный ключ для дешифровки всей луковицы, которую он получает. Данная операция позволяет выявить следующий луковый маршрутизатор и встроенную луковицу. Луковый маршрутизатор подгоняет встроенную луковицу для соответствия фиксированному размеру и посылает ее в следующий луковый маршрутизатор. После установки соединения данные можно посылать в обоих направлениях. Данные от инициатора каждый раз повторно шифруются с использованием алгоритмов и ключей, установленных в луковице. Во время движения данных через анонимное соединение каждый

луковый маршрутизатор убирает один слой шифрования, заданный криптографическим контролем информации в луковице, установившей дорогу, так что данные адресат получает уже простым текстом.

Вся информация (луковицы, данные, сетевой контроль) посылается через сеть порциями одинакового размера. Все ячейки пребывают в луковый маршрутизатор через фиксированные интервалы времени и смешиваются вместе. Луковица и ячейки с данными на разных участках сети имеют различный вид вследствие слоистого шифрования.

Временные и емкостные характеристики развертывания ЛМ достаточно малы. Так, временная сложность установки соединения обычно составляет менее одной секунды. Вычислительно-дорогое шифрование с открытым ключом используется только для передачи симметричного секретного ключа, во время фазы установки соединения. Фаза продвижения данных использует только AES-шифрование с переданным ключом, что намного быстрее по сравнению с шифрованием с открытым ключом. Задержка данных определяется числом луковых маршрутизаторов на протяжении соединения и может различаться в зависимости от длины маршрута.

В работе [6] можно найти описание модели автомата для протокола луковой маршрутизации, в которой анонимность и несвязность являются гарантированными. Ресурс [7] является официальным сайтом луковой маршрутизации. Работа [8] описывает вероятностный анализ луковой маршрутизации в виде черного ящика.

2. Тог — луковая маршрутизация второго поколения

Тог была создана в центре высокопроизводительных вычислительных систем исследовательской лаборатории Военно-морских сил США в рамках проекта Free Haven совместно с DARPA по федеральному заказу. В 2002 г. эта разработка была рассекречена, исходные тексты переданы независимым разработчикам, создавшим клиент-серверное приложение и опубликовавшим его под свободной лицензией.

Проект поддерживает правозащитная организация *Electronic Frontier Foundation*, существенную финансовую помощь оказывают Министерство обороны и Государственный департамент США, Национальный научный фонд.

Система ЛМ второго поколения имеет множество преимуществ по сравнению с оригинальной версией: свойство совершенной прямой секретности; контроль перегрузки; серверы каталогов; проверка целостности; настраиваемые политики выхода и практичный дизайн для сервисов со скрытой локацией. Тог работает в глобальной сети Интернет, не требует специальных привилегий и модификаций ядра, требует совсем небольшой синхронизации между узлами и предлагает разумный компромисс между анонимностью, удобством использования и эффективностью [9].

Сеть Тог представляет собой группу волонтерских серверов. Пользователи Тог используют эту сеть через подключение к серии виртуальных туннелей, что позволяет делиться информацией через публичные сети без компрометирования приватности.

Посмотреть, кто и зачем использует Тог, можно на ресурсе [10]. Пользователи Тог являются той частью, которая делает его столь защищенным. Тог прячет пользователя между другими пользователями в сети. Таким образом, чем больше пользовательская база Тог, тем сильнее защищается их анонимность. На сегодняшний день число людей, ежемесячно использующих Тог, приблизилось к 2 млн, а число волонтерских серверов в сети ежедневно превышает 6000.

Сеть Тог является оверлейной. Каждый луковый маршрутизатор запускается как нормальный процесс на уровне пользователя, без каких-либо специальных привилегий. Каждый пользователь запускает локальное программное обеспечение, называемое луковым прокси (далее — ЛП), для получения директорий, установки цепи и обеспечения цепи соединений. Эти ЛП принимают ТСП-поток и размножают их через цепи. Луковый маршрутизатор на другой стороне цепи соединяется с запрошенной конечной точкой и передает данные.

Трафик проходит через соединения в сообщениях фиксированного размера. Каждое сообщение имеет длину 512 байт и состоит из заголовка и полезной информации. Заголовок содержит идентификатор цепи, специфицирующий принадлежность цепи (много цепей могут быть размноженными в одном TLS-соединении).

Оригинальная луковая маршрутизация строила одну цепь для каждого ТСП-потока, однако в Тог каждая цепь может быть поделена между несколькими ТСП-потоками.

Для создания приватного пути прохождения через сеть с помощью ЛМ пользовательское программное обеспечение или клиент последовательно строят цепь защищенных соединений через ретрансляторы в сети. Каждый ретранслятор на пути знает только то, какой ретранслятор отправил ему данные и какому ретранслятору он должен их передать. Никакой отдельный ретранслятор не знает полного пути, который проделывает пакет данных внутри сети.

Как только цепь устанавливается, пользователь получает возможность анонимно пользоваться Интернетом и просматривать скрытые службы Тог. Тог работает только для ТСП-потока и может быть использован любым приложением с поддержкой сетевого протокола SOCKS.

В работе [11] приведен обзор скрытых служб сети Тог, недоступных в обычном Интернете (такие порталы имеют доменную зону .onion и называются Tor hidden services). Ресурс [12] содержит различные метрики, относящиеся к Тог. Ресурс [13] описывает анализ трафика внутри сети Тог и типичное поведение пользователей. Работа [14] содержит аналитическое исследование скрытых служб Тог в Интернете.

3. Пиринговые (P2P) анонимные сети

Под пиринговой (P2P) сетью понимается одноранговая сеть, состоящая из группы равноправных компьютеров. В такой сети каждый компьютер может выступать клиентом, сервером или узлом для распространения информации в группе. Сети P2P разделяют на централизованные и децентрализованные. В свою очередь, децентрализованные сети делят на структурированные, неструктурированные и гибридные [15]. Пиринговые анонимные сети, которые мы рассматриваем далее в этом разделе, являются децентрализованными и гибридными. Различают три поколения пиринговых сетей [16].

Tarzan — отказоустойчивая, масштабируемая и легкоуправляемая сеть, представляющая собой пиринговый анонимизирующий оверлей [17]. Инициатор сообщения выбирает путь для пакетно-ориентированной маршрутизации через псевдослучайно выбранные узлы ограниченной топологии, таким образом Tarzan предоставляет анонимность как клиенту, так и серверу.

Узлы сети Tarzan коммуницируют через последовательности ретрансляторов, выбранных из пула узлов-волонтеров. Разработчики предоставили технологию для открытия и выбора других узлов в качестве коммуникационных ретрансляторов: все узлы могут быть потенциальными организаторами трафика, т. е. потенциальными ретрансляторами.

Tarzan позволяет приложениям-клиентам взаимодействовать с Интернет-серверами через специальные туннели. Два конца туннеля — узлы Tarzan, на которых запущено клиентское приложение, и узел Tarzan, на котором запущен транслятор сетевого адреса. Tarzan полностью прозрачен для клиентских приложений и серверов. Так что он должен быть установлен и настроен на всех участвующих узлах.

MorphMix обеспечивает подключение к системе простым способом любого человека, имеющего доступ к Интернету, и эффективно оперирует большим числом участвующих узлов, даже несмотря на динамическую среду и присутствие ненадежных узлов [18].

Процесс шифрования организуется следующим образом. До того как участник n_1 посылает сообщение участнику n_2 , он заголовок шифрует в соответствии с правилами шифрования ссылок между n_1 и n_2 [18], используя симметрический ключ k_{11} . Когда n_2 получает сообщение, он убирает шифрование ссылки, используя k_{11} , убирает один слой шифрования, используя k_{n1} , определяет следующее промежуточное соединение в соответствии с идентификатором в заголовке, устанавливает поля в заголовке для следующей ссылки, шифрует его в соответствии с шифрованием ссылок между n_2 и n_3 , используя k_{12} , посылает все это участнику n_3 . Это продолжается до тех пор, пока не достигается финальный узел, ретранслирующий данные на сервер, с которым n_1 хочет осуществить взаимодействие. Сообщения посылаются обратно n_1 точно так же,

но в обратном порядке — каждый узел добавляет слой шифрования.

С технической точки зрения важно понимать, где оперирует сеть: на верхушке IP-слоя или на уровне приложения. В верхушке IP-слоя система прозрачна для транспортного протокола и протокола приложения. Данные поступают от инициатора после прохождения IP-слоя и посылаются через сеть с использованием UDP. На уровне приложения программа доступа, запущенная на компьютере инициатора, устанавливает TCP-соединение.

В настоящее время сети Tarzan и MorphMix, относящиеся к пиринговым сетям первого поколения, не используются. В данной статье мы их рассматриваем, поскольку они оказали значительное влияние на последующие разработки.

Freenet оперирует как сеть идентичных узлов, обеспечивающих место для хранения данных и маршрутизацию запросов к ним. Учитывается желание пользователя физически разместить данные в том или ином домене [19]. Трансляционный поиск или централизованный каталог расположений не используется. Обозначения файлов не позволяют определить их физическое расположение, поэтому невозможно обнаружить первоисточник или пункт назначения файла, продвигающегося через сеть. У каждого узла есть свое собственное хранилище данных, которое он делает доступным для чтения и записи в сети. Freenet дает возможность пользователям делиться своим свободным пространством на диске.

Запросы для ключей пересылаются от узла к узлу, через цепочки прокси-запросов, в которых каждый узел принимает решение о том, куда посылать запрос дальше. Маршрут определяется в зависимости от запрошенного ключа. Алгоритмы маршрутизации для хранения и получения данных были специально разработаны для адаптивной настройки маршрутов в реальном времени.

Каждому запросу дается лимит выхода из промежуточного соединения, аналогичный времени жизни IP-соединения. Время жизни уменьшается в каждом узле для предотвращения возникновения бесконечных цепей. Любому запросу приписывается псевдоуникальный случайный идентификатор. Поэтому узлы могут предотвращать циклы посредством отказа выполнять те запросы, которые они уже обрабатывали. При возникновении такой ситуации узел выбирает другой узел для дальнейшего соединения. Этот процесс продолжается до тех пор, пока запрос не будет выполнен или не исчерпает свое время жизни.

I2P-сеть является полностью распределенной, автономной, масштабируемой, эластичной и безопасной [20]. Все компоненты сети поставляются с открытым исходным кодом. Примечательной особенностью I2P-сети является то, что она может как выступать в роли оверлейной сети, использующейся в качестве надстройки над Интернетом, так и работать автономно, независимо от Интернета.

С технологической точки зрения I2P строит сеть P2P, которая берет преимущества анонимности и безопасности сетей, основанных на миксах Чаума [5]: свободную маршрутизацию и микс-каскады, производительность, масштабируемость и устойчивость распределенных хэш-таблиц, глобальную совместимость с Интернетом.

Сеть не нуждается в информации о пункте назначения сообщения. Аналогично, при получении сообщения, посланного через I2P, никто не знает, откуда оно поступило или кто его послал, однако отправитель может включить эту информацию. Более того, у машин, направляющих письмо из своего компьютера в точку назначения, нет информации об отправителе сообщения и точке назначения.

В сети используется локальная независимость. Это означает, что во время отправления в пункт назначения для сети неважно, где он находится физически. I2P включает не только сетевое программное обеспечение, но и I2P SDK, у которого есть API на нескольких языках, есть реализации маршрутизаторов, которые поддерживают коммуникации только с локальными конечными точками.

Вместо фокусирования на анонимном доступе в публичный Интернет ключевой целью проекта I2P является предоставление анонимного хостинга сервисов (аналог скрытых сервисов Tor).

Каждый узел I2P является маршрутизатором, так что нет четких различий между сервером и клиентом. I2P не использует централизованные серверы каталогов для подключения к узлам, а применяет вместо этого распределенные хэш-таблицы, описанные в работе [21].

Вместо ссылок на другие роутеры и сервисы I2P использует криптографические идентификаторы, при этом отсутствует DNS-подобный сервис.

Криптографический идентификатор маршрутизатора отличается от идентификатора сервиса, поэтому, если сервис будет запущен на каком-то маршрутизаторе, установление связи между этими двумя идентификаторами представляет собой трудную задачу. I2P использует вариант луковой маршрутизации, который называют чесночной маршрутизацией (Garlic Routing) [20].

Многие сервисы, такие, например, как BitTorrent, eDonkey и т. п., могут находиться внутри сети I2P [22]. Основные приложения, доступные в сети I2P: Susimail — почтовый клиент [23], SusiDNS — DNS-клиент [24], I2Psnark — торрент-клиент [25], iMule — свободный анонимный клиент файлообменной сети [26].

В работе [27] исследованы внутренние сервисы I2P с точки зрения сайтов, находящихся внутри сети.

Netsukuku представляет собой ячеистую сеть с P2P-протоколом, генерирующим и поддерживающим себя автономно. Этот протокол разработан для обработки неограниченного числа узлов с минимальной нагрузкой на процессор и память [28].

Сеть устанавливается через компьютеры, соединенные друг с другом физически, таким образом, она не является оверлейной. Netsukuku строит

маршруты, которые соединяют все компьютеры в сети и является самоуправляемой и автономной. При добавлении узла к Netsukuku сеть автоматически переприсваивает топологию, прокладывая самые быстрые и эффективные маршруты для коммуникаций с вновь прибывшими узлами. При увеличении числа узлов в сети она становится более эффективной. В Netsukuku отсутствует разница между приватными и публичными сетями.

Эта сеть является децентрализованной и распределенной. IP-адрес, определяющий компьютер, выбирается случайно, поэтому невозможно ассоциировать его с каким-то конкретным физическим местом. Маршруты, созданные огромным числом узлов, имеют высокую сложность и плотность. Единственный способ контролировать сеть — получить над ней физический контроль, поскольку каждый узел сети является ее частью.

В настоящее время маршрутизаторами Интернета управляют разные протоколы, такие как OSPF, RIP или BGP, основанные на разных классических алгоритмах, способных найти лучший путь для достижения узла в сети.

Эти алгоритмы подходят исключительно для создания небольших и средних сетей, поскольку требуют больших затрат процессорного времени и памяти. Ни один из этих протоколов не может быть использован в такой сети, как Netsukuku, где каждый узел является маршрутизатором, поскольку карта всех маршрутизаторов требует места на каждом компьютере, подсоединенном к сети (около 10 Гбайт).

В сети Netsukuku используется собственный алгоритм, называемый QSPN [29]. В этом алгоритме вся сеть представлена в виде фрактала для вычисления маршрутов, необходимых для подключения узла ко всем остальным.

Благодаря фрактальной структуре нужно всего лишь несколько килобайт, чтобы хранить всю карту Netsukuku.

Помимо сети Netsukuku, существуют еще несколько подобных решений. Так, например, сеть Hyperboria представляет собой автономную, пиринговую беспроводную ячеистую сеть в диапазоне 2,4 ГГц. В такой сети каждый пользователь является провайдером самому себе: с вами нельзя разорвать договор о пользовании Интернетом и подслушать сообщения специальным оборудованием. Сеть является самонастраиваемой, и каждый клиент, подключающийся к сети, увеличивает ее емкость. Современные протоколы для строительства этой сети, такие, например, как сетевой протокол Cjdn, гарантируют шифрование всего трафика, проходящего через сеть [30]. Для государства такая сеть представляет двойное явление: с одной стороны, такой тип сетей позволяет за меньшие деньги подключать к сети удаленные регионы, а с другой стороны, трафик в таких сетях не может быть перехвачен и проанализирован.

Turtle является так называемой сетью F2F, представляющей специфическую форму сети P2P,

в которой пользователи могут осуществлять прямые соединения для обмена информацией только с друзьями или пользователями, которым они доверяют [31]. В сети имеется большой набор узлов и большой набор данных. Предполагается, что у каждого узла сети есть владелец, обладающий персональным набором данных и желающий получить доступ к остальным данным в сети.

Каждый набор данных имеет свой набор свойств, состоящий из пар, содержащих атрибут и значение, которые используются во время обработки запросов пользователей. Запросы состоят из определенного числа пар (атрибут, значение), связанных логическими операторами AND, OR, NOT.

Каждый пользователь устанавливает криптозащищенное соединение между своим узлом и всеми дружественными узлами в наборе.

Во время распространения запроса генерируется дерево передачи запроса, с корнем в узле, изначально направившим запрос. Дерево строится на основе связей доверия между пользователями и используется для доставки ответа на запрос. Для того чтобы сопоставить запросы с ответами, каждый узел хранит таблицу с запросами, которые он ретранслировал, но для которых процесс ответа на запрос еще не завершен.

Ответ на запрос состоит из адреса запрашивающего узла, финального бита, идентификатора запроса, значения счетчика промежуточных соединений, ответа. Финальный бит используется для дифференциации между частичными и окончательными ответами. Узел, получающий положительный ответ от одного из своих детей-узлов в дереве трансляции запроса, немедленно доложит узлу-родителю. Узел указывает, что у него больше нет ответов для продвижения вперед, через отсылку пакета ответа с финальным набором битов.

Запрос завершается после того, как вызвавший узел получает финальный ответ от всех своих друзей. Вызвавший узел собирает все части пакетов ответа воедино. Узел сортирует все частичные ответы, чтобы установить отдельные наборы атрибутов данных. Как только пользователь выбирает результат, в котором он заинтересован, происходит выдача данных.

Cеть RetroShare — новое поколение файлового P2P обмена имеет F2F-архитектуру. Эта сеть позволяет установить шифрованное соединение между аутентифицированными друзьями [32]. Соединение используется для разных коммуникационных сервисов и файлообмена. Оно не зависит от корпоративной системы или центрального сервера, так что все данные посылаются только друзьям и в некоторых случаях ретранслируются через них их друзьям, что делает RetroShare децентрализованной социальной файлообменной сетью.

RetroShare содержит следующие коммуникационные сервисы: приватные чаты с друзьями; приватные или публичные чаты комнаты; письма друзьям; форумы; голос через IP.

Существуют и другие файлообменные сети, например Guntella, которая является полностью децентрализованной сетью второго поколения [33], сети AntsP2P [34], MUTE [35], OneSwarm [36] относятся к сетям третьего поколения и отличаются повышенной безопасностью.

Дополнительную информацию, посвященную анонимным сетям P2P, можно найти в нескольких работах. Работа [37] представляет обзор файлообменных систем P2P, предлагающих некоторую форму анонимности для пользователя. В работе [38] приведен обзор частных сетей P2P, в которых инфраструктура и ресурсы предоставляются пользователями, а новые пользователи могут присоединиться только по персональному приглашению.

Заключение

В работе приведен обзор успешно применяющихся анонимных сетей. На сегодняшний день пользователи имеют богатый выбор решений, позволяющих сохранить свою анонимность в глобальной сети Интернет и даже развернуть собственный анонимный сервис. Сети различаются по своей архитектуре, типу маршрутизации, предназначению и целевой аудитории. Однако при всем многообразии решений нет ни одного, которое могло бы предоставить абсолютную защиту от внешнего наблюдателя. Технологии деанонимизации будет посвящена следующая статья авторов.

Список литературы

1. **National Security Strategy** [Electronic resource] // Whitehouse [Official website]. URL: https://www.whitehouse.gov/sites/default/files/docs/2015_national_security_strategy.pdf (accessed: 8.09.2015).
2. **Зыков В.** ФСБ готовит закон против анонимности в интернете. 16 августа, 2013. [Электронный ресурс] // Известия [Официальный сайт]. URL: <http://izvestia.ru/news/555552> (дата обращения: 26.08.2015 г.).
3. **Goldschlag D., Reed M., Syverson P.** Onion Routing for Anonymous and Private Internet Connections. January 28, 1999 [Electronic resource] // Onion Routing [Official website]. URL: <http://www.omon-router.net/Publications/CACM-1999.pdf> (accessed: 8.09.2015).
4. **Patent US 6266704** — Onion routing network for securely moving data through communication networks [Electronic resource] // Google [Official website]. URL: <http://www.google.com/patents/US6266704> (accessed: 17.09.2015).
5. **Chaum D.** Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms [Electronic resource] // Free Haven [Official website]. URL: <http://www.freehaven.net/anonbib/cache/chaum-mix.pdf> (accessed: 8.09.2015).
6. **Feigenbaum J., Johnson A., Syverson P.** A Model of Onion Routing with Provable Anonymity [Electronic resource] // Yale [Official website]. URL: <http://www.cs.yale.edu/homes/jf/FJS.pdf> (accessed: 8.09.2015).
7. **Onion Routing** [Electronic resource] // Onion Routing [Official website]. URL: <http://www.onion-router.net> (accessed: 8.09.2015).
8. **Feigenbaum J., Johnson A., Syverson P.** Probabilistic Analysis of Onion Routing in a Black-box Model [Electronic resource] // Yale [Official website]. URL: <http://www.cs.yale.edu/homes/jf/WPES07-Aaron.pdf> (accessed: 8.09.2015).
9. **Dingledine R., Mathewson N., Syverson P.** Tor: The Second-Generation Onion Router [Electronic resource] // Tor project [Official website]. URL: <https://svn.torproject.org/svn/projects/design-paper/tor-design.pdf> (accessed: 8.09.2015).
10. **Who uses Tor?** [Electronic resource] // Tor project [Official website]. URL: <https://www.torproject.org/about/torusers.html.en> (accessed: 8.09.2015).
11. **Biryukov A., Pustogarov I., Weinmann R.-P.** Content and popularity analysis of Tor hidden services. July 29, 2013 [Electronic resource].

resource] // Cryptome [Official website]. URL: <https://cryptome.org/2013/09/tor-analysis-hidden-services.pdf> (accessed: 8.09.2015).

12. **Tor Metrics** [Electronic resource] // Tor project [Official website]. URL: <https://metrics.torproject.org> (accessed: 8.09.2015).

13. **Chaabane A., Manils P., Kaafar M. A.** Digging into Anonymous Traffic: a deep analysis of the Tor anonymizing network [Electronic resource] // IEEE [Official website]. URL: http://ieeexplore.ieee.org/xpl/login.jsp?tp=&number=5636000&url-http%3A%2F%2Fieeexplore.ieee.org%2Fxppls%2Fabs_all.jsp%3Farnumber%3D5636000 (accessed: 8.09.2015).

14. **Gupta N., Malhotra H.** Analytical Study of Tor Hidden Services On Internet [Electronic resource] // International Journal of Research and Development in Technology & Management Sciences — Kailash [Official website]. URL: <http://journal.rtmonline.in/vol20iss8/05262.pdf> (accessed: 8.09.2015).

15. **Scalable and Secure P2P Overlay Networks** [Electronic resource] // Wayne State University [Official website]. URL: <http://www.cs.wayne.edu/~weisong/papers/shen04-overlay.pdf> (accessed: 25.09.2015).

16. **Peer-to-Peer Overlay Networks: A Survey** [Electronic resource] // California state university Northridge [Official website]. URL: <http://www.csun.edu/~andrzei/COMP529-S05/papers/TR-P2P.pdf> (accessed: 25.09.2015).

17. **Freedman M. J., Morris R.** Tarzan: A Peer-to-Peer Anonymizing Network Layer [Electronic resource] // MIT, [Official website]. URL: <http://pdos.csail.mit.edu/tarzan/docs/tarzan-ccs02.pdf> (accessed: 8.09.2015).

18. **Rennhard M., Plattner B.** Introducing MorphMix: Peer-to-Peer based Anonymous Internet Usage with Collusion Detection [Electronic resource] // Free haven: (Official website). URL: <http://www.freehaven.net/anonbib/cache/morphmix:wpes2002.pdf> (accessed: 8.09.2015).

19. **Clarke I., Sandberg O., Wiley B., Hong T. W.** Freenet: A Distributed Anonymous Information Storage and Retrieval System [Electronic resource] // Stanford University: [Official website]. URL: <http://snap.stanford.edu/class/cs224w-readings/clarke00freenet.pdf> (accessed: 8.09.2015).

20. **Astolfi F., Kroese J., Oorschot J.** I2P — The Invisible Project [Electronic resource] // Media Technology: [Official website]. URL: http://mediatechnology.leiden.edu/images/uploads/docs/wt2015_i2p.pdf (accessed: 8.09.2015).

21. **Maymounkov P., Mazieres D.** Kademlia: A Peer-to-peer Information System Based on the XOR Metric [Electronic resource] // MIT: [Official website]. URL: <http://pdos.csail.mit.edu/~petar/papers/maymounkov-kademlia-lncs.pdf> (accessed: 8.09.2015).

22. **Supported Applications** [Electronic resource] // I2P: [Official website]. URL: <https://geti2p.net/en/docs/applications/supported#email> (accessed: 8.09.2015).

23. **Susimail** [Electronic resource] // Susimail: [Official i2p website]. URL: <http://127.0.0.1:7657/susimail/susimail> (accessed: 8.09.2015).

24. **Susidns** [Electronic resource] // Susidns: [Official i2p website]. URL: <http://127.0.0.1:7657/susidns/> (accessed: 8.09.2015).

25. **I2psnark** [Electronic resource] // I2psnark: [Official i2p website]. URL: <http://127.0.0.1:7657/i2psnark/> (accessed: 8.09.2015).

26. **IMule** [Electronic resource] // I2P forum: [Official i2p website]. URL: <http://forum.i2p/viewtopic.php?t=2213> (accessed: 8.09.2015).

27. **Crenshaw A.** Darknets and hidden servers: Identifying the true IP/network identity of I2P service hosts [Electronic resource] // Irongeek: [Official website]. URL: <http://www.irongeek.com/i.php?page=security/darknets-i2p-identifying-hidden-servers> (accessed: 8.09.2015).

28. **The Netsukuku** Wired [Electronic resource] // Netsukuku: [Official website]. URL: <http://netsukuku.freaknet.org> (accessed: 8.09.2015).

29. **Quantum Shortest Path Netsukuku** [Electronic resource] // Arxiv: [Official website]. URL: <http://arxiv.org/pdf/0705.0817v1.pdf> (accessed: 17.09.2015).

30. **Hyperboria** — The privacy-friendly network without borders [Electronic resource] // Hyperboria: [Official website]. URL: <https://hyperboria.net> (accessed: 8.09.2015).

31. **Popescu B. C., Crispo B., Tanenbaum A. S.** Safe and Private Data Sharing with Turtle: Friends Team-Up and Beat the System [Electronic resource] // NLnet: [Official website]. URL: <https://nlnet.nl/project/turtle/2004-cspw.pdf> (accessed: 8.09.2015).

32. **Retrosare** — secure communications for everyone [Electronic resource] // Retrosare: [Official website]. URL: <http://retrosare.sourceforge.net> (accessed: 8.09.2015).

33. **Gnutella** site archive [Electronic resources] // Internet archive Wayback machine: [Official website]. URL: <https://web.archive.org/web/20080525005017/http://www.gnutella.com/> (accessed: 8.09.2015).

34. **Ants P2P** [Electronic resource] // Ants P2P. [Official website]. URL: <http://antsp2p.sourceforge.net> (accessed: 8.09.2015).

35. **Simple, Anonymous File Sharing** [Electronic resource] // MUTE: [Official website]. URL: <http://mute-net.sourceforge.net> (accessed: 8.09.2015).

36. **OneSwarm** — Privacy preserving peer-to-peer data sharing [Electronic resource] // OneSwarm: [Official website]. URL: <http://www.oneswarm.org/index.html> (accessed: 8.09.2015).

37. **Chotia T., Chatzikokolakis K.** A Survey of Anonymous Peer-to-Peer File-Sharing [Electronic resource] // Ecole Polytechnique: [Official website]. URL: <http://www.lix.polytechnique.fr/~kostas/papers/AnonyP2PSurvey.pdf> (accessed: 8.09.2015).

38. **Rogers M., Bhatti S.** How to Disappear Completely: A Survey of Private Peer-to-Peer Networks [Electronic resource] // Researchgate: [Official website]. URL: https://www.researchgate.net/publication/228700831_How_to_disappear_completely_A_survey_of_private_peer-to-peer_networks (accessed: 8.09.2015).

S. M. Avdoshin, PhD, professor, Head of Software Engineering School, Faculty of Computer Science, HSE, savdoshin@hse.ru

A. V. Lazarenko, undergraduate student, Software Engineering School, Faculty of Computer Science, HSE, avlazarenko@edu.hse.ru
National Research University Higher School of Economics (HSE)

Technology of Anonymous Networks

This paper is an overview of currently used anonymous networks based on technology of onion routing and peer-to-peer networking. It describes key features of the networks and their comparative characteristics. The main purpose of every anonymous network is to protect information from the adversaries and provide users with great level of anonymity. All networks can be clustered on two classes: onion routing and its modifications and plain-old peer-to-peer networks. In the first class the major participant is Tor, which is based on the second generation of onion routing. On the other hand, P2P networks can be divided on 2 classes: traditional peer-to-peer and friend-to-friend. Friend-to-friend is a type of routing where users connects only to those users, who are considered as friends. The first class of peer-to-peer networks contains: Tarzan, MorphMix, Freenet, I2P, Netsukuku. The second class is represented by such networks as: Turtle, RetroShare. Current paper is focused only on those networks, which were successful on practice, or have strong impact on anonymous systems. Nowadays users have a wide specter of different solutions which can be used for protecting anonymity on the Internet. Anonymous networks differ by architectures, routing types and target audiences. Unfortunately, there is no any solution, which guarantees 100 % defense from adversaries. Every technology has its own weaknesses and vulnerabilities, allowing attacker to somehow deanonymize a particular user.

Keywords: anonymous networks, onion routing, peer-to-peer, layered encryption, invisible internet, overlay networks

References

1. **National Security Strategy** [Electronic resource], *Whitehouse* [Official website]. URL: https://www.whitehouse.gov/sites/default/files/docs/2015_national_security_strategy.pdf (accessed: 8.09.2015).
2. **Zykov V.** FSB gotovit zakon protiv anonimnosti v internete. [Electronic resource], *Izvestiya* [Official website]. URL: <http://izvestia.ru/news/555552> (accessed: 26.08.2015 г.).
3. **Goldschlag D., Reed M., Syverson P.** Onion Routing for Anonymous and Private Internet Connections. January 28, 1999 [Electronic resource], *Onion Routing* [Official website]. URL: <http://www.omon-router.net/Publications/CACM-1999.pdf> (accessed: 8.09.2015).
4. **Patent US 6266704.** Onion routing network for securely moving data through communication networks [Electronic resource], *Google* [Official website]. URL: <http://www.google.com/patents/US6266704> (accessed: 17.09.2015).
5. **Chaum D.** Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms [Electronic resource], *Free Haven* [Official website]. URL: <http://www.freehaven.net/anonbib/cache/chaum-mix.pdf> (accessed: 8.09.2015).
6. **Feigenbaum J., Johnson A., Syverson P.** A Model of Onion Routing with Provable Anonymity [Electronic resource], *Yale* [Official website]. URL: <http://www.cs.yale.edu/homes/jf/FJS.pdf> (accessed: 8.09.2015).
7. **Onion Routing** [Electronic resource], *Onion Routing* [Official website]. URL: <http://www.onion-router.net> (accessed: 8.09.2015).
8. **Feigenbaum J., Johnson A., Syverson P.** Probabilistic Analysis of Onion Routing in a Black-box Model [Electronic resource], *Yale* [Official website]. URL: <http://www.cs.yale.edu/homes/jf/WPES07-Aaron.pdf> (accessed: 8.09.2015).
9. **Dingledine R., Mathewson N., Syverson P.** Tor: The Second-Generation Onion Router [Electronic resource], *Tor project* [Official website]. URL: <https://svn.torproject.org/svn/projects/design-paper/tor-design.pdf> (accessed: 8.09.2015).
10. **Who uses Tor?** [Electronic resource], *Tor project* [Official website]. URL: <https://www.torproject.org/about/torusers.html.en> (accessed: 8.09.2015).
11. **Biryukov A., Pustogarov I., Weinmann R.-P.** Content and popularity analysis of Tor hidden services. July 29, 2013 [Electronic resource], *Cryptome* [Official website]. URL: <https://cryptome.org/2013/09/tor-analysis-hidden-services.pdf> (accessed: 8.09.2015).
12. **Tor Metrics** [Electronic resource], *Tor project* [Official website]. URL: <https://metrics.torproject.org> (accessed: 8.09.2015).
13. **Chaabane A., Manils P., Kaafar M. A.** Digging into Anonymous Traffic: a deep analysis of the Tor anonymizing network [Electronic resource], *IEEE* [Official website]. URL: http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=5636000&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxppls%2Fabs_all.jsp%3Farnumber%3D5636000 (accessed: 8.09.2015).
14. **Gupta N., Malhotra H.** Analytical Study of Tor Hidden Services On Internet [Electronic resource], *International Journal of Research and Development in Technology & Management Sciences — Kailash* [Official website]. URL: <http://journal.rtmonline.in/vol20iss8/05262.pdf> (accessed: 8.09.2015).
15. **Scalable and Secure P2P Overlay Networks** [Electronic resource], *Wayne State University* [Official website]. URL: <http://www.cs.wayne.edu/~weisong/papers/shen04-overlay.pdf> (accessed: 25.09.2015).
16. **Peer-to-Peer Overlay Networks: A Survey** [Electronic resource], *California state university Northridge* [Official website]. <http://www.csun.edu/~andrzei/COMP529-S05/papers/TR-P2P.pdf> (accessed: 25.09.2015).
17. **Freedman M. J., Morris R.** Tarzan: A Peer-to-Peer Anonymizing Network Layer [Electronic resource], *MIT* [Official website]. URL: <http://pdos.csail.mit.edu/tarzan/docs/tarzan-ccs02.pdf> (accessed: 8.09.2015).
18. **Rennhard M., Plattner B.** Introducing MorphMix: Peer-to-Peer based Anonymous Internet Usage with Collusion Detection [Electronic resource], *Free haven* [Official website]. URL: <http://www.freehaven.net/anonbib/cache/morphmix:wpes2002.pdf> (accessed: 8.09.2015).
19. **Clarke I., Sandberg O., Wiley B., Hong T. W.** Freenet: A Distributed Anonymous Information Storage and Retrieval System [Electronic resource], *Stanford University* [Official website]. URL: <http://snap.stanford.edu/class/cs224w-readings/clarke00freenet.pdf> (accessed: 8.09.2015).
20. **Astolfi F., Kroese J., Oorschot J.** I2P — The Invisible Project [Electronic resource], *Media Technology* [Official website]. URL: http://mediatechnology.leiden.edu/images/uploads/docs/wt2015_i2p.pdf (accessed: 8.09.2015).
21. **Maymounkov P., Mazieres D.** Katdemia: A Peer-to-peer Information System Based on the XOR Metric [Electronic resource], *MIT* [Official website]. URL: <http://pdos.csail.mit.edu/~petar/papers/maymounkov-kademia-lncs.pdf> (accessed: 8.09.2015).
22. **Supported Applications** [Electronic resource], *I2P* [Official website]. URL: <https://geti2p.net/en/docs/applications/supported#email> (accessed: 8.09.2015).
23. **Susimail** [Electronic resource], *Susimail* [Official i2p website]. URL: <http://127.0.0.1:7657/susimail/susimail> (accessed: 8.09.2015).
24. **Susidns** [Electronic resource], *Susidns* [Official i2p website]. URL: <http://127.0.0.1:7657/susidns/> (accessed: 8.09.2015).
25. **I2psnark** [Electronic resource], *I2psnark* [Official i2p website]. URL: <http://127.0.0.1:7657/i2psnark/> (accessed: 8.09.2015).
26. **IMule** [Electronic resource], *I2P forum* [Official i2p website]. URL: <http://forum.i2p/viewtopic.php?t=2213> (accessed: 8.09.2015).
27. **Crenshaw A.** Darknets and hidden servers: Identifying the true IP/network identity of I2P service hosts [Electronic resource], *Irongeek* [Official website]. URL: <http://www.irongeek.com/i.php?page=security/darknets-i2p-identifying-hidden-servers> (accessed: 8.09.2015).
28. **The Netsukuku Wired** [Electronic resource], *Netsukuku* [Official website]. URL: <http://netsukuku.freaknet.org> (accessed: 8.09.2015).
29. **Quantum Shortest Path Netsukuku** [Electronic resource], *Arxiv* [Official website]. URL: <http://arxiv.org/pdf/0705.0817v1.pdf> (accessed: 17.09.2015).
30. **Hyperboria** — The privacy-friendly network without borders [Electronic resource], *Hyperboria* [Official website]. URL: <https://hyperboria.net> (accessed: 8.09.2015).
31. **Popescu B. C., Crispo B., Tanenbaum A. S.** Safe and Private Data Sharing with Turtle: Friends Team-Up and Beat the System [Electronic resource], *NLnet* [Official website]. URL: <https://nlnet.nl/project/turtle/2004-cspw.pdf> (accessed: 8.09.2015).
32. **RetrosShare** — secure communications for everyone [Electronic resource], *RetrosShare* [Official website]. URL: <http://retrosShare.sourceforge.net> (accessed: 8.09.2015).
33. **Gnutella** site archive [Electronic resources], *Internet archive Wayback machine* [Official website]. URL: <https://web.archive.org/web/20080525005017/http://www.gnutella.com/> (accessed: 8.09.2015).
34. **Ants P2P** [Electronic resource], *Ants P2P* [Official website]. URL: <http://antsp2p.sourceforge.net> (accessed: 8.09.2015).
35. **Simple**, Anonymous File Sharing [Electronic resource], *MUTE* [Official website]. URL: <http://mute-net.sourceforge.net> (accessed: 8.09.2015).
36. **OneSwarm** — Privacy preserving peer-to-peer data sharing [Electronic resource], *OneSwarm* [Official website]. URL: <http://www.oneswarm.org/index.html> (accessed: 8.09.2015).
37. **Chotia T., Chatzikokolakis K.** A Survey of Anonymous Peer-to-Peer File-Sharing [Electronic resource], *Ecole Polytechnique* [Official website]. URL: <http://www.lix.polytechnique.fr/~kostas/papers/AnonymousP2PSurvey.pdf> (accessed: 8.09.2015).
38. **Rogers M., Bhatti S.** How to Disappear Completely: A Survey of Private Peer-to-Peer Networks [Electronic resource], *Researchgate* [Official website]. URL: https://www.researchgate.net/publication/228700831_How_to_disappear_completely_A_survey_of_private_peer-to-peer_networks (accessed: 8.09.2015).

ЦИФРОВАЯ ОБРАБОТКА СИГНАЛОВ И ИЗОБРАЖЕНИЙ DIGITAL PROCESSING OF SIGNALS AND IMAGES

УДК 519.7

Л. А. Лютикова, канд. физ.-мат. наук, зав. отделом, e-mail: lylarisa@yandex.ru,
Е. В. Шматова, мл. науч. сотр., e-mail: lenavsh@yandex.ru,
Федеральное государственное бюджетное научное учреждение
"Институт прикладной математики и автоматизации" (ИПМА), г. Нальчик

Анализ и синтез алгоритмов распознавания образов с использованием переменного-значной логики

Проводится логический анализ заданной предметной области, представляющей собой объект и описывающие его признаки в терминах переменного-значной логики, анализируется набор алгоритмов, работающих на заданной области. В ходе исследования разработаны логические процедуры построения корректных алгоритмов, анализирующих предметную область, моделирующих базу знаний для заданных объектов, минимизирующих ее и выделяющих уникальный набор признаков для каждого объекта.

Ключевые слова: алгоритмы, обучающая выборка, набор данных, база знаний, предметная область, переменного-значная логика, дизъюнкты, решающее правило

Введение

На первом этапе развития теории и практики распознавания образов для решения практических задач возникло большое число методов и алгоритмов, применявшихся без какого-либо обоснования. Такие методы проверялись экспериментально. Решение задач медицинской и технической диагностики, компьютерного прогноза месторождений, а также построение экспертных систем ввело в обиход большое число некорректных (эвристических) алгоритмов. В результате возникла необходимость в развитии теории корректирующих операций, синтеза корректных алгоритмов минимальной сложности и решения вопросов об их устойчивости с помощью математических методов.

Логический подход может служить основой для построения теории синтеза корректных алгоритмов распознавания на базе существующих семейств алгоритмов. Данные методы, несмотря на отсутствие адекватных математических моделей исследуемых зависимостей между образом и его свойствами, неполноту и противоречивость данных, позволяют создавать алгоритмы, реализующие определенные рассуждения эксперта.

Как правило, математическую логику привыкли использовать для оценки истинности высказывания. При решении поставленной задачи используется аппарат математической логики, который представляется уместным исходя из реальных качеств характеризующих объектов. Поскольку харак-

теризуемый объект имеет ряд признаков, разбитых каждый на свое число состояний, то возможность кодировать каждый признак предикатами разной значности представляется удобным. Конечной целью применения предикатов переменной значности является вывод о принадлежности исследуемых данных какому-либо объекту или классу.

В данной работе рассматривается логический подход к теоретическому обоснованию построения корректных алгоритмов, расширяющих область получаемых решений на базе существующих алгоритмов.

Постановка задачи

Описание объекта представляет собой m -мерный вектор $X = \{x_1, x_2, \dots, x_m\}$, где m — число признаков, используемых для характеристики объекта, причем j -я координата этого вектора равна значению j -го признака, $j = 1, \dots, m$. В описании объекта допустимо отсутствие информации о значении того или иного признака. Совокупность некоторого числа объектов и их признаков представляет собой выборку, на которой проработало n алгоритмов. Качество работы каждого алгоритма оценивается булевой функцией $a_j(X_j, y_j)$. Ни один из рассматриваемых алгоритмов не распознает все множество заданных объектов. Предлагается логический метод построения нового алгоритма, являющийся корректным на всем множестве распознаваемых объектов, на основе существующих алгоритмов и решающих правил, составленных для исследуемой области.

Формальная постановка задачи

На предметной области, состоящей из объектов и их признаков, рассматривается ряд алгоритмов A_1, A_2, \dots, A_n решения задачи распознавания.

Пусть $X = \{x_1, x_2, \dots, x_m\}$, значение переменной $x_i \in \{0, 1, \dots, k_r - 1\}$, где $k_r \in [2, \dots, N]$, $N \in \mathbb{Z}$ — множество признаков, рассматриваемых в рамках переменного-значной логической системы; $X_i = \{x_1(y_i), x_2(y_i), \dots, x_m(y_i)\}$, $i = 1, \dots, l$ — вектор признаков, характеризующих объект $y_i \in Y$, $Y = \{y_1, y_2, \dots, y_l\}$ — множество объектов; $A = \{A_1, A_2, \dots, A_n\}$ — множество алгоритмов, $a_j(X_i, y_i) \in \{0, 1\}$; $i = 1, 2, \dots, l$; $j = 1, 2, \dots, n$ — качество работы алгоритма на заданном наборе признаков $X_i = \{x_1(y_i), x_2(y_i), \dots, x_m(y_i)\}$, $i = 1, 2, \dots, l$, определяемое формулой

$$a_j(y_i) = \begin{cases} 1, & A_j(X_i) = y_i, \\ 0, & A_j(X_i) \neq y_i, \end{cases} \quad i = 1, 2, \dots, l, j = 1, 2, \dots, n,$$

т. е. результат работы алгоритма на заданном наборе признаков оценивается в рамках булевой алгебры:

1 — алгоритм A_j распознал объект y_i по заданным признакам X_i ,

0 — алгоритм A_j не распознал объект y_i по заданным признакам X_i .

Совокупность описанных данных можно представить в виде двумерной матрицы следующего вида:

x_1	x_2	...	x_m	Y	A'_1	A'_2	...	A'_n
$x_1(y_1)$	$x_2(y_1)$...	$x_m(y_1)$	y_1	$a_1(y_1)$	$a_2(y_1)$...	$a_n(y_1)$
$x_1(y_2)$	$x_2(y_2)$...	$x_m(y_2)$	y_2	$a_1(y_2)$	$a_2(y_2)$...	$a_n(y_2)$
...
$x_1(y_l)$	$x_2(y_l)$...	$x_m(y_l)$	y_l	$a_1(y_l)$	$a_2(y_l)$...	$a_n(y_l)$

$A'_i = \{a_i(y_1), a_i(y_2), \dots, a_i(y_l)\}$, $i = 1, 2, \dots, n$, — вектор, представленный столбцом значений оценки качества работы алгоритма A_i .

Некоторые из заданных в обучающей выборке объектов не распознаются ни одним из рассматриваемых алгоритмов. Математически это записывается следующим образом:

$$\exists y_i \in Y | A_1(X_i) \neq y_i, A_2(X_i) \neq y_i, \dots, A_n(X_i) \neq y_i, \\ i = 1, 2, \dots, l, j = 1, \dots, n.$$

Необходимо построить алгоритм на основе заданных, который обеспечит распознавание всех объектов, заданных в данной предметной области. Это значит найти $A_{n+1}(X_i) | A_{n+1}(X_i) = y_i$ и $A_{n+1}(X) | A_{n+1}(X) = Y$.

Определение. Будем говорить, что алгоритм корректен на множестве объектов Y , определяемых совокупностью признаков X , если $\forall y_i \in Y: a_j(X_i, y_i) = 1$, $i = 1, 2, \dots, l; j = 1, \dots, n$. Иными словами, алгоритм является корректным на том множестве объектов, которые он правильно распознает.

Для анализа предметной области будем использовать алгебру переменного-значной логики [3, 4], которая дает возможность для выразительного кодирования разнородной информации, так как каждый отдельный признак $x_i \in \{0, 1, \dots, k_r - 1\}$ может быть закодирован предикатом любой значности, удобной именно для данного признака.

Аппарат переменного-значной логики является удобным инструментом для простого и выразительного кодирования и декодирования свойств исследуемых объектов. Он позволяет избежать сложности построения процедур фаззификации и дефаззификации, которые необходимы в случае использования нечеткой логики, а также существенно упрощает построение логических конструкций, отражающих соответствие исследуемых объектов и их свойств. В рамках предлагаемого подхода эти логические конструкции выражаются в виде продукционных правил.

Операции переменного-значной логики

Определение. Высказывания переменного-значной логики являются высказываниями, истинность которых определяется следующими значениями: $\{0, 1, \dots, k_r - 1\}$, $k_r \in [2, \dots, N]$, $N \in \mathbb{Z}$, B — формула высказывания, определенная тремя операциями:

- отрицание или обобщенная инверсия (унарная операция);
- & конъюнкция (бинарная);
- дизъюнкция (бинарная).

Используются также константы:

$$0, 1 \dots k_r - 1, k_r \in [2, \dots, N], N \in \mathbb{Z}.$$

Пусть X_i — независимая многозначная переменная величина, $X_i \in [0, \dots, k_r - 1]$, являющаяся одной из характеристик объекта. Введем еще несколько функций и свойств переменного-значной логики.

Перечислим функции переменного-значной логики, называемые элементарными.

1. Значение переменной:

$$x_i^j = \begin{cases} j, & x_i = j; \\ 0, & x_i \neq j. \end{cases}$$

2. Обобщенная инверсия:

$$\overline{x^j} = x^0 \vee x^1 \vee \dots \vee x^{j-1} \vee x^{j+1} \vee \dots \vee x^{k-1}.$$

Заданная таким образом инверсия обеспечивает включение всех возможных интерпретаций отрицания в различных многозначных логических системах.

3. Пусть переменные $X \in [0, \dots, k_i - 1]$, $Y \in [0, \dots, k_j - 1]$ имеют разную значность, тогда обобщенная дизъюнкция:

$$X \vee Y = \max \left[\frac{X}{k_i - 1}; \frac{Y}{k_j - 1} \right] l,$$

$$\text{где } l = \begin{cases} k_i - 1 & \text{при } \frac{X}{k_i - 1} > \frac{Y}{k_i - 1}; \\ k_j - 1 & \text{иначе.} \end{cases}$$

4. Обобщенная конъюнкция:

$$X \& Y = \min \left[\frac{X}{k_i - 1}; \frac{Y}{k_j - 1} \right] l,$$

$$\text{где } l = \begin{cases} k_i - 1 & \text{при } \frac{X}{k_i - 1} < \frac{Y}{k_j - 1}; \\ k_j - 1 & \text{иначе.} \end{cases}$$

5. Импликацию для переменного-значной логики зададим следующим выражением:

$$X \rightarrow Y = \overline{X} \vee Y.$$

Элементарные функции переменного-значной логики обладают следующими свойствами:

$$0 \& X = 0;$$

$$1 \& X = X;$$

$$(k - 1) \vee X = (k - 1);$$

$$0 \vee X = X;$$

$$x^j \& x^k = \begin{cases} x^j, & j = k; \\ 0, & j \neq k. \end{cases}$$

Решающие правила и функция качества ответов

Определение. Решающим правилом для заданной предметной области назовем следующее высказывание:

$$\&_{s=1}^m x_s(y_i) \rightarrow y_i, \quad i = 1, \dots, l, \quad x_s(y_i) \in \{0, 1, \dots, k_i - 1\}, \\ k_i \in [2, \dots, N], \quad N \in Z.$$

В данном случае решающее правило — это правило продукции, логическая интерпретация которого говорит, что из совокупности определенных признаков (тот и этот и т. д. признаки) следует определенный объект.

Пусть имеется n алгоритмов $\{A_1, A_2, \dots, A_n\}$, частично распознающих заданную область. Для каждого заданного набора признаков X_i строим функции качества работы каждого алгоритма и получаем набор векторов $A'_j = \{a_j(y_1), a_j(y_2), \dots, a_j(y_l)\}$, $j = 1, 2, \dots, n$, представленный в матрице в виде столбца A'_j . Получаем результат работы алгоритма на каждой заданной строке, соответствующей объекту y_i , этому же объекту соответствует продукционное правило

$$\&_{s=1}^m x_s(y_i) \rightarrow y_i, \quad x_s(y_i) \in \{0, 1, \dots, k_r - 1\}, \\ i = 1, \dots, l, \quad s = 1, \dots, m.$$

Полученный столбец можно рассматривать как частично заданную булеву функцию на множестве переменных $\{X, Y\}$.

Построение алгоритма, расширяющего область решений

При обработке данных целесообразным является выбор алгоритма, имеющего следующие свойства: $A_j(X_i) = y_i$, $a_j(X_i, y_i) = 1$. В случае если хотя бы один алгоритм нашел решение вида $A_j(X_i) = y_i$, то $\bigvee_{j=1}^n a_j(y_i) = 1$. Если ни один из рассматриваемых алгоритмов не распознает объект y_i , то $\bigvee_{j=1}^n a_j(y_i) = 0$.

Представим всю обучающую выборку в виде решающих правил:

$$\&_{s=1}^m x_s(y_i) \rightarrow y_i, \quad i = 1, \dots, l, \quad x_s(y_i) \in \{0, 1, \dots, k_r - 1\}, \\ k_r \in [2, \dots, N], \quad N \in Z.$$

Для каждого алгоритма выберем решающие правила, по которым алгоритмы распознают объекты: если $\exists a_j(y_i) = 1$, то $\&_{s=1}^m x_s(y_i) \rightarrow y_i$, $i = 1, \dots, l$, $x_s(y_i) \in \{0, 1, \dots, k_r - 1\}$, $k_r \in [2, \dots, N]$, $N \in Z$.

Составим функцию, являющуюся конъюнкцией таких решающих правил для данного алгоритма, руководствуясь следующими логическими рассуждениями: алгоритм A_j распознает объект y_i и алгоритм A_j распознает объект y_p и все остальные объекты, распознаваемые этим алгоритмом:

$$F_j(X_i) = \&_{a_j(y_i)=1} (\&_{s=1}^m x_s(y_i) \rightarrow y_i) = \\ = \&_{a_j(y_i)=1} (\bigvee_{s=1}^m \overline{x_s(y_i)} \vee y_i).$$

Далее можно применить алгоритм сокращения в адаптированном для многозначных логик варианте:

- если некоторая переменная входит в ДНФ (дизъюнктивно нормальная форма) с одним знаком во всех дизъюнктах, то удаляем все дизъюнкты, содержащие эту переменную (данная переменная неинформативна);
- если в ДНФ имеется какой-то однолитерный дизъюнкт x_i^j , то удаляем все дизъюнкты вида $x_i^j \& \dots$ (правило поглощения).

В результате для заданного алгоритма A_j получаем функцию F_j , соответствующую тем решающим правилам, которые распознал заданный алгоритм. Данная функция обладает рядом свойств [2] и практически строит базу знаний данного алгоритма, разбивая область решения на все возможные для данной области классы.

Теорема. Необходимым и достаточным условием, характеризующим набором признаков $\{X_j\}$ к классу K_r , является равенство $F_j(X_i) = K_r$.

Доказательство.

Пусть $F_1(X) = K_r$. Так как $f(X) = f_1(X) \vee f_2(X)$, то и $f(X) = K_r$. Функция $f(X)$ однозначно характеризует заданную базу данных (БД). Можно утверждать, что конкретный набор признаков $\{X_j\}$, ис-

пользуя данные, представленные в БД, характеризует класс K_r .

Предположим, что набор признаков $\{X_j\}$ характеризует объект класса K_r и это не противоречит исходным данным, тогда функция $f(X)$ примет значение $f(X_i) = f_1(X_i) \vee f_2(X_i)$, $f(X) = K_r$. Так как $f_2(X_j)$ не содержит дизъюнкты, содержащие классы, то можно утверждать, что $f_1(X) = K_r$.

Построив для каждого алгоритма соответствующие функции $F_j, j = 1, 2, \dots, n$, получаем множество функций F_1, \dots, F_n . Продолжая наши рассуждения, построим обобщающую функцию, являющуюся конъюнкцией функций F_1, \dots, F_n : $F = \&_{j=1}^n F_j$. Проведя вычисления и преобразования, получим функцию

$$F(X, Y) = f_1(X) \vee f_2(X, Y),$$

где $f_1(X)$ — функция, содержащая только переменные x_s , и назовем ее функцией настройки, а дизъюнкты этой функции — элементами настройки, которые не имеют значения для идентификации объекта, но имеют значение в случае построения нового алгоритма на ранее нераспознанных объектах; $f_2(X, Y)$ — функция, содержащая конъюнкцию признаков и объектов, являющаяся функцией для определения индивидуальных признаков заданных объектов.

Для построения нового корректного алгоритма на наборах данных, не распознаваемых прежними алгоритмами, достаточно использовать функцию $f_1(X)$. Новый алгоритм является конъюнкцией $f_1(X)$ и решающего правила объекта, не распознанного другими алгоритмами. В результате получаем уникальную характеристику объекта и сочетания его признаков, которые не относятся ни к одному из ранее распознанных объектов:

$$A_{n+1} = f_1(X) \& (\&_{s=1}^m x_s^j \rightarrow y_j) \vee_{j=1}^n A_j.$$

ПРИМЕР 1

Пусть $X = \{x_1, x_2, x_3\}$ — набор признаков, значение каждого признака кодируется в рамках трехзначной логической системы $x_s \in \{0, 1, 2\}$, $s = 1, 2, 3$. Соотношение входных данных (признаков объектов), самих объектов и результаты работы алгоритмов распознавания данных объектов представлены следующей матрицей:

x_1	x_2	x_3	Y	A'_1	A'_2	A'_3
0	1	1	a	1	0	1
1	2	2	b	0	1	0
0	1	2	c	1	1	0
1	0	0	d	0	0	0

На основе приведенных соотношений можно записать:

$$A_1: F_1 = (x_1^0 \& x_2^1 \& x_3^1 \rightarrow a) \& (x_1^0 \& x_2^1 \& x_3^2 \rightarrow c)$$

(алгоритм A_1 распознает объекты a и c);

$$A_2: F_2 = (x_1^1 \& x_2^2 \& x_3^2 \rightarrow b) \& (x_1^0 \& x_2^1 \& x_3^2 \rightarrow c);$$

$$A_3: F_3 = (x_1^0 \& x_2^1 \& x_3^1 \rightarrow a);$$

$$F = F_1 \& F_2 \& F_3 = f_1(X) \vee f_2(X, Y);$$

$$f_1(X) = x_1^2 \vee x_2^0 \vee x_3^0 \vee x_1^1 x_2^1 \vee x_1^1 x_3^1 \vee x_2^2 x_3^1;$$

$$f_2(X, Y) = bx_1^1 \vee bx_2^2 \vee ax_3^1 \vee cx_1^0 x_3^2 \vee cx_2^1 x_3^2 \vee bcx_3^2 \vee ax_1^0 x_3^1 \vee ax_2^1 x_3^1 \vee ab;$$

$$A_4 = f_1(X) \& (x_1^1 \& x_2^0 \& x_3^0 \rightarrow d) =$$

$$= x_1^0 x_2^0 \vee x_2^0 x_3^1 \vee x_2^0 x_3^2 \vee x_1^0 x_3^0 \vee x_2^1 x_3^0 \vee x_2^2 x_3^0 \vee$$

$$\vee x_1^1 x_2^1 \vee x_1^1 x_3^1 \vee x_2^2 x_3^1 \vee dx_2^0 \vee dx_3^0.$$

Алгоритм A_4 выделяет индивидуальные признаки объекта d , а именно значения $x_2 = 0$ и $x_3 = 0$. Алгоритм A_4 в дизъюнкции с ранее заданными алгоритмами дает всю область решения заданной предметной области.

Логический подход к построению корректного алгоритма на заданной предметной области

При добавлении в предыдущую матрицу требования корректности к алгоритму $A_{n+1}(X)$ получим матрицу следующего вида:

x_1	x_2	...	x_m	Y	A'_1	A'_2	...	A'_n	A'_{n+1}
$x_1(y_1)$	$x_2(y_1)$...	$x_m(y_1)$	y_1	$a_1(y_1)$	$a_2(y_1)$...	$a_n(y_1)$	1
$x_1(y_2)$	$x_2(y_2)$...	$x_m(y_2)$	y_2	$a_1(y_2)$	$a_2(y_2)$...	$a_n(y_2)$	1
...
$x_1(y_l)$	$x_2(y_l)$...	$x_m(y_l)$	y_l	$a_1(y_l)$	$a_2(y_l)$...	$a_n(y_l)$	1

То есть для $A_{n+1}(X)$ все значения $a_{n+1}(y_i) = 1$, $i = 1, 2, \dots, l$.

Поскольку $a_j(y_i)$ может быть рассмотрена как булева переменная, то $A'_{n+1}(A'_1, A'_2, \dots, A'_n)$ — булева функция, имеющая значения 1 на всех задан-

ных в предметной области наборах $(A'_1, A'_2, \dots, A'_n)$, и может быть представлена следующим образом:

$$A'_{n+1}(A'_1, A'_2, \dots, A'_n) = \bigvee_{i=1}^l \&_{j=1}^n A_j^{\sigma'}(y_i),$$

$$i = 1, 2, \dots, l, j = 1, 2, \dots, n;$$

$$A_j^{\sigma'}(y_i) = \begin{cases} A'_j, & a_j(y_i) = 1; \\ \overline{A'_j}, & a_j(y_i) = 0. \end{cases}$$

Будем считать, что A'_j — это совокупность решающих правил, распознаваемых алгоритмом, $\overline{A'_j}$ — совокупность решающих правил, не распознаваемых данным алгоритмом:

$$A'_j = \&_{s=1}^l (\&_{s=1}^m x_s(y_i) \rightarrow y_i), \text{ когда } a_j(y_i) = 1;$$

$$\overline{A'_j} = \overline{\&_{s=1}^l (\&_{s=1}^m x_s(y_i) \rightarrow y_i)}, \text{ когда } a_j(y_i) = 0.$$

Выразим импликацию и получим следующие выражения:

$$A'_j = \&_{s=1}^l (\bigvee_{s=1}^m \overline{x_s(y_i)} \vee y_i), \text{ когда } a_j(y_i) = 1;$$

$$\overline{A'_j} = \&_{s=1}^l (\&_{s=1}^m x(y_i) \& \overline{y_i}), \text{ когда } a_j(y_i) = 0.$$

Вся исследуемая предметная область может быть представлена в виде решающих правил вида

$$\&_{s=1}^m x_s(y_i) \rightarrow y_i, \quad i = 1, \dots, l, \quad x_s(y_i) \in \{0, 1, \dots, k_r - 1\},$$

$$k_r \in [2, \dots, N], \quad N \in \mathbb{Z}. \quad (1)$$

Теорема. Пусть задано множество решающих правил вида

$$\&_{j=1}^m x_s(y_i) \rightarrow y_i, \quad i = 1, \dots, l, \quad x_j(y_i) \in \{0, 1, \dots, k_r - 1\},$$

$$k_r \in [2, \dots, N], \quad N \in \mathbb{Z},$$

представляющих собой некоторую исследуемую предметную область, тогда на всей исследуемой области

$$A'_{n+1}(A'_1, A'_2, \dots, A'_n) = \bigvee_{i=1}^l \&_{j=1}^n A_j^{\sigma'}(y_i) = 1,$$

$$i = 1, 2, \dots, l, j = 1, 2, \dots, n.$$

Доказательство.

Каждый алгоритм входит в предлагаемую дизъюнкцию как A'_j в одну или несколько конъюнкций и как $\overline{A'_j}$ — также в одну или несколько конъюнкций, поскольку в противном случае это либо универсальный алгоритм, для которого все $a_j(y_i) = 1$, $i = 1, 2, \dots, l$, либо неработающий алгоритм $a_j(y_i) = 0$, $i = 1, 2, \dots, l$. Так как A'_j — это совокупность решающих правил, которые распознаны алгоритмом A_j , а $\overline{A'_j}$ — совокупность решающих правил, которые этим алгоритмом не распознаны, то дизъюнкция этих правил даст описание полной предметной области для каждого алгоритма.

При построении ДНФ

$$A'_{n+1}(A'_1, A'_2, \dots, A'_n) = \bigvee_{i=1}^l \&_{j=1}^n A_j^{\sigma'}(y_i)$$

может быть сокращена до тупиковой ДНФ известными методами. На стадии, когда на место A'_j будут подставлены решающие правила, можно применить алгоритм сокращения в адаптированном для многозначных логик варианте:

- если некоторая переменная входит в ДНФ с одним знаком во всех дизъюнктах, то удаляем все дизъюнкты, содержащие эту переменную (данная переменная неинформативна);
- если в ДНФ имеется какой-то однолитерный дизъюнкт x'_j , то применяем правило поглощения дизъюнкта.

В результате для каждого дизъюнкта получим минимизированную базу знаний, соответствующую набору правил, описанных этим дизъюнктом. Такие дизъюнкты обладают рядом свойств [2] и разбивают область решения на все возможные для данной области классы. Объединение этих областей приводит к минимизированной базе знаний для всей заданной области.

ПРИМЕР 2

Пусть $X = \{x_1, x_2, x_3\}$, $x_i \in \{0, 1, 2\}$:

x_1	x_2	x_3	Y	A'_1	A'_2	A'_3	A'_4
0	0	1	a	1	0	1	0
0	2	1	b	0	0	1	1
2	1	2	c	0	1	0	1
1	2	0	d	0	0	0	0

Строим дизъюнкцию по строкам

$$F = A'_{n+1}(A'_1, A'_2, \dots, A'_n) = \bigvee_{i=1}^l \&_{j=1}^n A_j^{\sigma'}(y_i);$$

$$F = A_1 \& \overline{A_2} \& A_3 \& \overline{A_4} \vee \overline{A_1} \& \overline{A_2} \& A_3 \& A_4 \vee$$

$$\vee \overline{A_1} \& \overline{A_3} \& A_2 \& A_4 \vee \overline{A_1} \& \overline{A_2} \& \overline{A_3} \& \overline{A_4},$$

и далее, записывая алгоритмы решающими правилами и преобразовывая, получим следующее выражение:

$$A_5 = (x_1^0 \& x_2^0 \& x_3^1 \rightarrow a) \& (x_1^0 \& x_2^2 \& x_3^1 \rightarrow b) \&$$

$$\& (x_1^1 \& x_2^2 \& x_3^0 \rightarrow d) =$$

$$= x_1^2 \vee x_3^2 \vee x_2^1 \vee x_1^1 x_2^0 \vee x_1^1 x_3^1 \vee x_1^0 x_3^0 \vee x_2^0 x_3^0 \vee$$

$$\vee x_3^0 d \vee b x_1^0 x_2^2 \vee b x_2^2 x_3^1 \vee b d x_2^2 \vee a x_1^0 x_2^0 \vee$$

$$\vee a x_2^0 \vee a x_2^0 x_3^1 \vee x_1^1 d.$$

Алгоритм A_5 выделяет индивидуальные признаки объекта d .

Заключение

В результате проведенного логического анализа данной предметной области и решающих правил, описывающих объекты, становится понятным, что сложность полученного алгоритма зависит от качества уже заданных алгоритмов и скрытых закономерностей самой предметной области. Предложенный логический метод синтеза позволяет построить корректный алгоритм на всей области данных, моделирует базу знаний, минимизирует ее и фиксирует уникальный для каждого объекта набор признаков.

Список литературы

1. **Журавлев Ю. И.** Об алгебраическом подходе к решению задач распознавания или классификации // Проблемы кибернетики. 1978. Т. 33. С. 5—68.
2. **Shibzukhov Z. M.** Correct Aggregation Operations with Algorithms // Pattern Recognition and Image Analysis. 2014. Vol. 24, N. 3, pp. 377—382.
3. **Тимофеев А. В., Лютикова Л. А.** Развитие и применение многозначных логик и сетевых потоков в интеллектуальных системах // Труды СПИИ РАН. Вып. 2. 2005. С. 114—126.
4. **Лютикова Л. А.** Моделирование и минимизация баз знаний в терминах многозначной логики предикатов. Препринт. Нальчик: НИИ ПМА КБНЦ РАН, 2006. 33 с.
5. **Воронцов К. В.** Оптимизационные методы линейной и монотонной коррекции в алгебраическом подходе к проблеме распознавания // Журнал вычислительной математики и математической физики. 2000. Т. 40, № 1. С. 166—176.
6. **Журавлев Ю. И., Рудаков К. В.** Об алгебраической коррекции процедур обработки (преобразования) информации // Проблемы прикладной математики и информатики. 1987. С. 187—198.

L. A. Lyutikova, Head of Department, e-mail: lyularisa@yandex.ru, **E. V. Shmatova**, Junior Researcher

Recognition Algorithms Analysis and Synthesis with Varied Values Logic

In this paper we carried out logical analysis of the given specified domain that presents an object the features of which describe it using varied values logic and a set of algorithms running within the given domain is studied. We developed logical procedure for constructing correct algorithms, to research the specified domain, simulate knowledge base for specific objects that minimize it and allocate a unique set of characteristics for each object.

Keywords: algorithm, training set, a set of data, knowledge base, the subject area, logic of varied values, clauses, the decision rule

References

1. **Zhuravljov Ju. I.** Ob algebraicheskom podhode k resheniju zadach raspoznavanija ili klassifikacii, *Problemy kibemetiki*, 1978, vol. 33, p. 5—68.
2. **Shibzukhov Z. M.** Correct Aggregation Operations with Algorithms, *Pattern Recognition and Image Analysis*, 2014, vol. 24, no. 3, pp. 377—382.
3. **Timofeev A. V., Ljutikova L. A.** Razvitie i primenenie mnogoznachnyh logik i setevyh potokov v intellektual'nyh sistemah, *Trudy SPII RAN*. 2005, Iss. 2, pp. 114—126.
4. **Ljutikova L. A.** Modelirovanie i minimizacija baz znaniy v terminah mnogoznachnoj logiki predikatov. Preprint, Nal'chik: NII PMA KBNC RAN, 2006, 33 p.
5. **Voroncov K. V.** Optimizacionnye metody linejnoy i monotonnoj korrekcii v algebraicheskom podhode k probleme raspoznavanija, *Zhurnal vychislitel'noj matematiki i matematicheskoy fiziki*, 2000, vol. 40, no. 1, pp. 166—176.
6. **Zhuravljov Ju. I., Rudakov K. V.** Ob algebraicheskoj korrekcii procedur obrabotki (preobrazovanija) informacii, *Problemy prikladnoj matematiki i informatiki*, 1987, pp. 187—198.

ДИСКУССИОННЫЙ КЛУБ DISCUSSION CLUB

УДК 004.6; 528; 004.8

В. Я. Цветков, д-р техн. наук, проф.,

зам. руководителя центра фундаментальных и перспективных исследований, e-mail: cvj2@mail.ru,
Научно-исследовательский и проектно-конструкторский институт информатизации, автоматизации
и связи на железнодорожном транспорте (ОАО "НННАС")

Полисемия информации

Рассматривается полисемия термина "информация" как объективная ситуация, обусловленная развитием термина и его применением в разных областях. Описана "догма одномерности" как характерная причина ошибочных выводов и заключений. Прослеживается эволюция термина "информация" с работ Аристотеля до наших дней. Описано состояние теории информации и информатики, строительство которых как наук не завершено. Это также способствуют полисемии информации.

Ключевые слова: информация, науки об информации, теория информации, информатика, полисемия, догма одномерности

Введение

В настоящее время осознанно или неосознанно в ряде научных исследований и публикаций применяют догму одномерности. В работе [1] выделены три признака догмы одномерности. Эти признаки проявляются как совместно, так и раздельно. *Первый признак* догмы одномерности состоит в том, что полисемическое понятие пытаются определить одним определением или одной трактовкой. *Второй признак* догмы одномерности состоит в том, что сложное многоаспектное явление рассматривается или трактуется только с одной точки зрения, в одном аспекте. *Третий признак* догмы одномерности состоит в принятии своей трактовки и в искажении или упрощении других определений, других точек зрения, которые наряду с данной трактовкой дают альтернативное объяснение данному явлению, процессу или закономерности. Третий признак характерен для специалистов с узкой специализацией, которые не способны понять другие точки зрения из-за недостаточного количества знаний.

Можно еще расширить число этих признаков, но выделим только три. *Четвертый признак* — это категоричные утверждения типа "все правы" или "все неправы", которые сопровождаются нарушением одного из основных законов логики — закона достаточного основания [2]. Такого рода утверждения правомочны, когда данный человек знает абсолютную истину (эпистеме), может это доказать, а остальные ее не знают.

Пятый признак состоит в применении линейных моделей в нелинейных явлениях и процессах.

Например, по двум точкам, которые лежат на параболе или гиперболе, строят прямую и утверждают о линейности этого явления за пределами этих точек. Описание всех нелинейных явлений линейной моделью — типичная догма одномерности.

Шестой признак догмы одномерности связан с тем, что сложный процесс и явление, которое зависит от множества равнозначных параметров, пытаются отобразить только одним параметром, значимо равным одному из многих. Особенно ярко это проявляется при наличии оппозиционных характеристик или оппозиционных переменных. Оппозиционные переменные [3, 4] — это переменные, отражающие два противоположных качества одного объекта или явления, например "хороший—плохой", "сильный—слабый", "достоинства—недостатки" и т. д. Эти переменные могут дополнять друг друга, например, у каждого объекта можно найти как достоинства, так и недостатки. Однако рассмотрение только достоинств или только недостатков всегда создает искаженную модель восприятия или описания. Все признаки догмы одномерности связаны с упрощением "сложной сущности" и заменой "сложной сущности" на "простую сущность". Такой подход встречается в некоторых работах по анализу понятия "информация".

О развитии термина "информация"

Энциклопедия "Британика" (2007) утверждает, что термин "информация" возник в IX веке и имеет латинские корни. Из латинского языка он перешел в испанский и во французский. Из французского

Сравнение некоторых терминов, связанных с информацией

Русс	Латынь	Итал.	Исп.	Франц.	Англ.
Информация	Notitia, Indicium, Informationem, Informationes	Informazioni, Dati	Información	Information, Renseignements, Informations, L'information	Information
Данные	Data, Notitia	Dati	Datos, Información	Données, Des données	Data
Сведения	Notitia	Dati	Datos	Données	Data
Факты	Facta, Rerum, Rebus	Fatti, Dati	Hechos, Datos, Nformación	Faits	Facts
Информировать	Informare, Certiorem	Informare	Informar	Informar	Inform
Передавать сведения	Transmittere notitia	Trasmettere le informazioni	Transmitir información	Transmettre de l'information	Transmit information
Форма	Forma	Forma	Forma	Forme	Form
Внешняя форма	Forma	Forma esterna	Forma externa	Forme extérieure	External form
Внутренняя форма	Interno forma	Forma interna	Forma interna	Forme interne	The internal form, The inner form
Неформальный	Tacitae, Senatus, Temere	Informale	Informal	Informel	Informal

языка перешел в английский. Согласно энциклопедии "Британик" впервые в официальном документе этот термин был употреблен в указе короля Георга в 1330 г. во фразе "Men so wise should go and inform their kings" — "Мудрые люди должны информировать их короля"¹. Однако исторически термин "информация" можно связать с работой Аристотеля "Категории" [5, 6], в которой, говоря о сущности, он выделяет понятие "форма" как внешнее (кажущееся) проявление сущности и понятие "внутренняя форма" — как основное содержание сущности. Это отодвигает данное понятие еще на 13 веков от IX века в глубь истории человечества.

В таблице приведены некоторые термины, связанные с информацией, на разных языках. Расположение иностранных терминов слева направо отражает путь, который прошел термин "информация" из латинского в английский язык. Обращает на себя внимание совпадение терминов "форма" и "информировать" во всех языках.

Объяснение этому дает работа Аристотеля "Метафизика" [7]: "... формой я называю суть бытия каждой вещи и ее первую сущность", "то, что обозначено как форма или сущность, не возникает, а возникает сочетание, получающее от нее свое наименование, и что во всем возникающем есть материя, так что одно [в нем] есть материя, а другое — форма". Форму Аристотель связывает с материей как внешнее ее проявление. Отсюда следует, что форма в понимании Аристотеля — это сведения о внешних свойствах и проявлениях объектов и явлений. Обращает на себя внимание в приведенной таблице различие термина "внешняя форма" на латинском языке и других языках. На латинском языке отсутствует различие между формой и внешней формой, поскольку форма и есть внешнее проявление, т. е. подразумевается форма как внешняя форма.

Говоря о противоположностях, Аристотель [5, 6] отмечает: "противоположности относятся к тому, что тождественно или по виду, или по роду". Следовательно, говоря о форме (внешней форме), мы должны иметь в виду ее противоположность — внутреннюю форму. В настоящее время для связи противоположностей широко употребляют термин *оппозиционные переменные* [4, 5], которые связывают позицию и оппозицию в единое описание свойств объекта или явления.

Обращает на себя внимание сходство термина "внутренняя форма" во всех языках, приведенных в таблице. Таким образом, если форма, как внешняя форма, несет сведения о внешних свойствах объектов и явлений, то ее противоположность — внутренняя форма — несет дополнительные сведения о внутренних свойствах объектов и явлений. Еще раз отметим фразу Аристотеля [7] "формой я называю суть бытия каждой вещи и ее *первую* (внешнюю *авт.*) сущность". При слове "первая", мы, вспоминая дихотомическое деление, должны подумать о наличии "второй". Следовательно, существует и *вторая* сущность, которую можно обозначить как *внутренняя* сущность. Если посмотреть на словообразование такой сущности из составного термина (внутренняя форма), то можно придти к следующей парадигме:

interno forma → informa.

Отсюда вытекает значение термина *informatio* — описание внутренней (неявной, скрытой) но важнейшей сущности объектов и явлений, которая дополняет внешнюю сущность и в совокупности дает полное (целостное) описание объектов и явлений окружающего мира.

Для обозначения понятия "сведения" в латинском языке использовался термин "современная нотация" (запись). Он был более общим по отношению к терминам "информация" и "форма" [8]. Отсюда

¹ Encyclopaedia Britannica — Encyclopaedia Britannica, Inc. 2007.

следует, что "сведения" (notitia IV до н.э.) включали описание *первой, второй* и прочих "сущностей".

С течением времени может происходить трансформация понятий. С IX века термин "информация" подменил термин "нотация" и стал обозначать сведения. Подобных трансформаций понятий в науке немало. Например, термин "геометрия" означает "измерение Земли" и первоначально использовался с этой целью. Но с течением времени его заменил термин "геодезия" — "деление Земли", а геометрия стала разделом математики. Однако международное общество геодезистов (основанное в 1878 г.) до сего времени использует название FIG (Federation Internationale des Geometres), которое буквально переводится как международное общество геометров.

Информация, информатика и теория информации

Полисемия информации [9, 10] усугубляется тем, что к настоящему времени теория информации не сформировалась, а информатика не является теорией информации и не рассматривает эти вопросы достаточно детально. В теории информации выделяют два направления: "энтропийный", положенное работами К. Э. Шеннона, и "содержательное", положенное работами Н. Винера и развиваемое в настоящее время работами Лучиано Флориди.

Энтропийное направление основано на работе К. Э. Шеннона [11]. Под информацией в этой теории понимают "нечто", что уменьшает неинформированность и неопределенность. Такая информация уменьшает незнание или неопределенность. Винер называет теорию К. Э. Шеннона "статистической теорией количества информации" [12]. Первоначально К. Э. Шеннон не говорил о своей работе как о теории информации. Он и отмечал, что "*семантические аспекты информации не релевантны техническим проблемам связи*" [11]. Однако в 1964 г. появилась совместная работа Шеннона и Уивера [13], в которой точка зрения "теории информации на основе математической теории связи" развивалась как основная.

Количеством информации в теории информации по К. Э. Шеннону называют информационную емкость сообщения или иной информационной конструкции безотносительно к содержанию. Ее измеряют в байтах и битах независимо от того, что в этих байтах содержится или не содержится ничего. Соответственно, информативность по К. Э. Шеннону — это измерение в битах информационной емкости носителя информации или информационной конструкции. По Шеннону информация уменьшает "наше незнание", по Виннеру и Флориди — информация увеличивает "наше знание". Увеличе-

ние знания и уменьшение незнания не эквивалентны, что показано в работах [14, 15].

Л. Флориди разделяет и развивает точку зрения Н. Винера о том, что главное в информации — содержательность и знания. Это он обозначает термином "семантика". Главным критерием наличия семантики в информации Л. Флориди определяет истинность информации с позиций эпистемологии [16].

Информатика в настоящее время включает три направления, два из которых представлены в работе Ю. Ю. Черного [17]. К концу 1940-х гг. в связи с частым употреблением словосочетания "научная информация" ученые ВИНТИ предложили "теорию научной информации". Эта позиция нашла отражение в работах А. И. Михайлова, В. А. Полушкина, А. И. Черного, Р. С. Гиляревского и др. (1962—1965 гг). Впоследствии эта область исследований была обозначена как "Информатика 1" [17]. В англоязычной научной литературе она имеет аналог — *information sciences*.

Термин "информатика" появился в 1960-х гг. во Франции для названия области, занимающейся автоматизированной переработкой информации, как слияние французских слов *information* и *automatique* (F. Dreyfus, 1962). В разных странах информатика (ср. *нем.* Informatik, *англ.* Information technology, *фр.* Informatique, *англ.* computer science — компьютерная наука — в США, *англ.* computing science — вычислительная наука — в Великобритании) интерпретировалась как наука о способах получения, накопления, хранения, преобразования, передачи, защиты и использования *информации*. При этом термин "информация" не связывался с какой-либо предметной областью, а был **обобщением понятия информации** как *объекта обработки* на компьютере.

В СССР и России довольно долго этот термин использовался как синоним термину "программирование". В России курсы по изучению информатики длительное время (до 2000 г.) включали изучение программирования в первую очередь и применение информационных систем — во вторую. Но никакой теории информации или семантики информации в них не было. Эту информатику и называют "Информатикой 2". В англоязычной научной литературе она имеет аналог — *computer sciences*.

Объектом изучения (областью исследований) "Информатики 2" или ее доминантой являются *методы обработки информации*, компьютерные модели, алгоритмы анализа и вычислений — *безотносительно* к области использования этой информации. Следует подчеркнуть, что "Информатика 2" занимается именно обработкой информации.

С этих позиций "Информатика 2" — *посредник* между математикой и логикой, с одной стороны, и прикладными науками — с другой. Она возникла и

развивается как наука об обработке информации, безотносительно к ее смысловому содержанию. Напомним определение А. П. Ершова: "Информатика — фундаментальная естественная наука, изучающая процессы передачи и обработки информации" [18]. То есть налицо тесные отношения с теорией связи и отсутствие таких отношений с теорией информации.

Как необходимость выхода из двойственного состояния информатики явились предложения по созданию "Интегральной информатики" или "Информатики 3" [19]. В работе [19] автор (идеолог интегральной информатики) признает неудачным название термина "информатика" и предлагает другой — "компьютинг". Но за рубежом существует достаточное число журналов с таким названием, на русский они переводятся как "вычисление" или "вычислительные методы".

Заключение

Наличие разных направлений в теории информации и информатике обуславливает объективно разные трактовки понятия информация.

Любой термин создается для обозначения и описания некой сущности (объекта, процесса, свойства, явления). Если в процессе развития науки эта сущность разделяется на разные сущности, то необходимо обновление и уточнение термина. Обозначать одним термином или понятием разные сущности нельзя. Это приводит к необходимости введения общих и частных понятий. Термин "информация" является полисемическим, поскольку с его помощью обозначают разные сущности, которые не являются эквивалентными. Сводить общее понятие информации к информации в какой-то узкой области не представляется корректным.

Информация, которая содержится в газетах, не равнозначна информации, которую передают друг другу насекомые. Информация, которую используют для создания компьютерных программ, не равнозначна речевой информации человеческого общения. Информация на космических снимках не равнозначна генетической информации. Информационная емкость сообщений (ее часто тоже называют информацией) не равнозначна семантике или содержательности информационных сообщений. И так далее.

Необходимо оговаривать, о какой информации идет речь относительно предметной области и технологий применения информации. Если использовать точку зрения Аристотеля и говорить об об-

щем понятии термина "информация", то можно трактовать термин "информация" следующим образом: информация — средство описания и раскрытия внешних и внутренних сущностей объектов и явлений окружающего мира, основанное на применении категорий как средства структурного анализа и интерпретации.

Список литературы

1. **Ожерельева Т. А.** Сложность информационных ресурсов // Современные наукоемкие технологии. 2014. № 4. С. 80—85.
2. **Цветков В. Я.** Логика в науке и методы доказательств. LAP LAMBERT Academic Publishing GmbH & Co. KG, Saarbrücken, Germany. 2012. 84 с.
3. **Ожерельева Т. А.** Оппозиционный анализ информационных моделей // Международный журнал прикладных и фундаментальных исследований. 2014. № 11 (Часть 5). С. 146—149.
4. **Tsvetkov V. Ya.** Opposition Variables as a Tool of Qualitative Analysis // World Applied Sciences Journal. 2014. 30 (11). P. 1703—1706.
5. **Аристотель.** Категории. Сочинения. В 4 т. (Серия "Философское наследие"). Т. 2. М.: Мысль, 1975—1983. 688 с.
6. **Categoriae and de interpretation.** The works of Aristotle. Vol. I / Translated into english under the editorship of W. D. Ross. Oxford at the Clarendon Press. 1928. 668 p.
7. **Аристотель.** Метафизика. Сочинения. В 4 т. (Серия "Философское наследие"). Т. 1. М.: Мысль, 1975—1983. 552 с.
8. **Иванников А. Д., Тихонов А. Н., Соловьев И. В., Цветков В. Я.** Инфосфера и инфология. М.: ТОРУС ПРЕСС, 2013. 176 с.
9. **Чернавский Д. С.** Проблема происхождения жизни и мышления с точки зрения современной физики // УФН. 2000. Т. 170. № 2. С. 157—183.
10. **Иванников А. Д., Тихонов А. Н., Цветков В. Я.** Основы теории информации. М.: МаксПресс, 2007. 356 с.
11. **Shannon C. E.** A Mathematical Theory of Communication // Bell System Technical Journal. 1948. Vol. 27. P. 379—423, 623—656.
12. **Winner N.** Cybernetics or Control and Communication in the Animal and the Mashine. New York — Herman et Cie, Paris: The Technology Press and John Wiley & Soris Inc. 1948. 194 p.
13. **Shannon C. E. and Weaver W.** The Mathematical Theory of Communication. The University of Illinois Press — URBANA. 1964. 132 p.
14. **Tsvetkov V. Ya.** The K. E. Shannon and L. Floridi's amount of information // Life Science Journal, 2014, N. 11 (11). P. 667—671.
15. **Цветков В. Я.** Информационная неопределенность и определенность в науках об информации // Информационные технологии. 2015. № 1. С. 3—7.
16. **Floridi L.** In defence of the veridical nature of semantic information // European Journal of Analytic Philosophy. 2007. Vol. 3, N. 1. P. 31—41.
17. **Черный Ю. Ю.** Полисемия в науке: когда она вредна? (на примере информатики) // Открытое образование. 2010. N. 6. С. 97—106.
18. **Ершов А. П.** Информатика: предмет и понятие // Кибернетика. Становление информатики. М.: Наука, 1986. С. 28—31.
19. **Колин К. К.** Становление информатики как фундаментальной науки и комплексной научной проблемы // Системы и средства информатики. Специальный выпуск. Научно-методологические проблемы информатики / Под ред. К. К. Колина. М.: ИПН РАН, 2006. С. 7—58.

Polysemy Information

This article describes the polysemy of the term information as the objective situation. This situation is due to the development of the term and its various applications in different areas. This article describes the "dogma of the one-dimensionality" as a typical cause of erroneous findings and conclusions. The article describes the evolution of the term "information" from the time of Aristotle to the modern days. The article describes the state of information theory and computer science. These sciences are not fully developed at the moment, this situation also contribute polisemii information.

Keywords: information, information science, information theory, computer science, polysemy, dogma-dimensionality

References

1. **Ozherel'eva T. A.** Slozhnost' informacionnyh resursov. *Sovremennye naukoemkie tekhnologii*, 2014, no. 4, pp. 80–85.
2. **Cvetkov V. Ya.** *Logika v nauke i metody dokazatel'stv.* LAP LAMBERT Academic Publishing GmbH & Co. KG, Saarbrücken, Germany, 2012, 84 p.
3. **Ozherel'eva T. A.** Oppozicionnyj analiz informacionnyh modelej. *Mezhdunarodnyj zhurnal prikladnyh i fundamentarnyh issledovanij*, 2014, no. 11 (chast' 5), pp. 746–749.
4. **Tsvetkov V. Ya.** Opposition Variables as a Tool of Qualitative Analysis. *World Applied Sciences Journal*, 2014, no. 30 (11), pp. 1703–1706.
5. **Aristotel'.** Kategorii. Sochineniya. V 4 t. (Seriya "Filosofskoe nasledie"). M.: Mysl', 1975–1983, vol. 2, 688 p.
6. **Categoriae and de interpretation.** The works of Aristotle. Vol. I. Translated into english under the editorship of W. D. Ross. Oxford at the Clarendon Press, 1928, 668 p.
7. **Aristotel'.** *Metafizika.* Sochineniya. V 4 t. (Seriya "Filosofskoe nasledie"). Moscow: Mysl', 1975–1983, vol. 1, 552 p.
8. **Ivannikov A. D., Tihonov A. N., Solov'ev I. V., Cvetkov V. Ya.** *Infosfera i infologiya.* Moscow: TORUS PRESS, 2013, 176 p.
9. **Chernavskij D. S.** Problema proiskhozhdeniya zhizni i myshleniya s tochki zreniya sovremennoj fiziki. *UFN*, 2000, vol. 170, no. 2, pp. 157–183.
10. **Ivannikov A. D., Tihonov A. N., Cvetkov V. Ya.** *Osnovy teorii informacii.* Moscow: MaksPress, 2007, 356 p.
11. **Shannon C. E.** *A Mathematical Theory of Communication, Bell System Technical Journal*, 1948, vol. 27, pp. 379–423 & 623–656, July & October.
12. **Winner N.** *Cybernetics or Control and Communication in the Animal and the Mashine.* New York—Herman et Cie, Paris: The Technology Press and John Wiley & Soris Inc. 1948, 194 p.
13. **Shannon C. E. and Weaver W.** *The Mathematical Theory of Communication.* The University of Illinois Press — URBANA, 1964, 132 p.
14. **Tsvetkov V. Ya.** The K. E. Shannon and L. Floridi's amount of information. *Life Science Journal*, 2014, no. 11 (11), pp. 667–671.
15. **Cvetkov V. Ya.** Informacionnaya neopredelennost' i opredelennost' v naukah ob informacii. *Informacionnye tekhnologii*, 2015, no. 1, pp. 3–7.
16. **Floridi L.** In defence of the veridical nature of semantic information. *European Journal of Analytic Philosophy*. 2007, vol. 3, no. 1, pp. 31–41.
17. **Chernyj Yu. Yu.** Polisemiya v nauke: kogda ona vredna? (na primere informatiki). *Otkrytoe obrazovanie*, 2010, no. 6, pp. 97–106.
18. **Ershov A. P.** Informatika: predmet i ponyatie. *Kibernetika. Stanovlenie informatiki.* Moscow: Nauka, 1986, pp. 28–31.
19. **Kolin K. K.** Stanovlenie informatiki kak fundamental'noj nauki i kompleksnoj nauchnoj problemy. *Sistemy i sredstva informatiki. Special'nyj vypusk. Nauchno-metodologicheskie problemy informatiki.* Pod red. K. K. Kolina. Moscow: IPI RAN, 2006, pp. 7–58.

ИНФОРМАЦИЯ

XVIII Международная конференция

"Аналитика и управление данными в областях с интенсивным использованием данных" ("Data Analytics and Management in Data Intensive Domains" (DAMDID))

будет проводиться с 11 по 14 октября 2016 г. в доме отдыха Ершово
(Московская область, Одинцовский район).

Конференция планируется как **мультидисциплинарный форум** исследователей и практиков из разнообразных областей науки, содействующий сотрудничеству и обмену идеями в сфере анализа и управления данными в условиях их интенсивного использования.

Подробности на сайте конференции <http://damdid2016.frccsc.ru>

ЖУРНАЛ В ЖУРНАЛЕ



**НЕЙРОСЕТЕВЫЕ
ТЕХНОЛОГИИ**

№ 4

АПРЕЛЬ

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.

Галушкин А. И., Пантюхин Д. В.

СуперЭВМ и мемристоры 304

Задаянчук А. И., Попова М. С., Стрижов В. В.

Выбор оптимальной модели классификации временных рядов 313

А. И. Галушкин, д-р техн. наук, проф., зам. зав. каф.,

Московский физико-технический институт, г. Долгопрудный,

проф. НИУ "Высшая школа экономики", г. Москва, e-mail: neurocomputer@yandex.ru,

Д. В. Пантюхин, ст. преподаватель, НИУ "Высшая школа экономики", г. Москва,

преподаватель, Московский физико-технический институт, г. Долгопрудный, e-mail: dim_beavis@mail.ru

СуперЭВМ и мемристоры

Излагается мнение авторов о перспективах развития области сверхвысокопроизводительной вычислительной техники — нейрокомпьютеров в связи с появлением мемристоров. Такие перспективы определяются сверхвысокой параллельностью, обусловленной переходом на нейросетевой логический базис, резким снижением энергопотребления за счет представления информации в виде последовательности узких импульсов, переходом к аналоговой реализации части алгоритмов с резким повышением быстродействия. Представлены примеры разработки подобных нейрочипов и нейросистем последних лет.

Ключевые слова: нейрокомпьютеры, нейрочипы, мемристоры, суперЭВМ, экзафлопные вычисления

Введение

Основными предпосылками для развития супервычислений в настоящее время являются:

- отход от архитектуры фон Неймана к распределенным (в том числе нейросетевым) архитектурам [1—3];
- развитие теории нейронных сетей, нейроматематики и нейроуправления [4—15];
- развитие технологии микроэлектроники для создания нейрокомпьютеров [16—19], в том числе нейросетевых ускорителей вычислений с мемристорами для:

— повышения производительности обработки информации при решении задач;

— резкого понижения энергопотребления по сравнению с классическими кластерными суперЭВМ и суперЭВМ на базе графических процессоров;

— обеспечения радиационной устойчивости вычислительных элементов;

— использования отечественной технологической базы, устойчиво освоенной на текущий момент времени;

— импортозамещения в части основных обрабатываемых элементов в интегральных вычислительных системах.

Основная цель работ — развитие и применение нового поколения нейрокомпьютеров наряду с созданием отечественных СБИС массового потребления — памяти и ПЛИС с мемристорной коммутационной средой.

Цели могут быть достигнуты за счет реализации следующих подходов:

- переход от классического фоннеймановского базиса решения задач к нейросетевому логическому базису;

- переход к импульсному представлению сигналов (когда информационный сигнал определяется последовательностью узких импульсов) вместо представления сигнала в виде уровней токов и напряжений;

- переход к другой перспективной элементной базе — мемристорам (вычислительным элементам, объединяющим функции вычислений и памяти) и, как следствие, реализация в вычислительной системе принципов аналого-цифровой обработки в целях повышения производительности при контролируемом качестве обработки.

Работы в области применения нейрокомпьютеров в США начались по программе DARPA Neural network Study в 1987—1988 гг. Программа изучения нейронных сетей, финансируемая DARPA/ТТО (Tactical Technology Office of the U.-S. Defense Advanced Research Projects Agency), ставила перед собой следующие задачи:

- определение потенциальных применений нейронных сетей в военных системах;
- выявление текущего состояния нейросетевых технологий;
- выявление тенденций развития нейросетевых технологий;
- разработка плана работ на ближайшие пять лет.

Работу возглавляла Лаборатория Линкольна Массачусетского технологического института (Massachusetts Institute of Technology's Lincoln Laboratory — EMIT/LL). Начиная с 1992 г. в США проводилась финансируемая правительством пятилетняя работа по разработке и применению нейрокомпьютеров в военной технике. Президент Буш назвал 90-е годы "десятилетием мозга" вследствие принципиальной важности теории нейронных сетей для развития современного общества.

Задачи, решаемые нейрокомпьютерами, в значительной степени определяют эффективность

применения систем. Эти задачи условно делятся на формализуемые и неформализуемые. Повышение уровня эффективности решения задач обоих классов связано с применением высокопроизводительной вычислительной техники и нейросетевых технологий. Прикладные задачи для высокопроизводительных систем в значительной части используют нейросетевые технологии. К таким задачам относятся:

- задачи обработки сигналов и изображений, в том числе радиолокационных;
- задачи обеспечения информационной безопасности;
- задачи управления динамическими объектами и др.

Попытки придания системам интеллектуальных свойств наталкиваются на ограничения по вычислительной мощности используемых вычислительных средств и их массогабаритным характеристикам, защищенности и т. п.

Реализация мемристорных систем позволит в значительной степени в будущих гибридных суперЭВМ реализовать на отечественной элементной базе вычислительную оболочку, несущую основную вычислительную нагрузку. Для различных применений это в значительной степени увеличит защищенность вычислительной системы в целом; позволит в значительной степени повысить надежность вычислительных систем и улучшит тактико-технические характеристики техники, в которой они будут применяться.

Тенденции развития нейрокомпьютеров

В монографии [20], опубликованной в 2015 г., представлены современные разработки нейрокомпьютеров и, в частности, отмечаются тенденции роста числа публикаций и числа исследователей в этой области (рис. 1).

Начиная с этапа развития разработок транзисторных систем в современных суперЭВМ реализованы два основных свойства: масштабируемость и двухслойность архитектур.

С нашей точки зрения, для перехода на экзафлопный уровень вычислений необходима реализа-

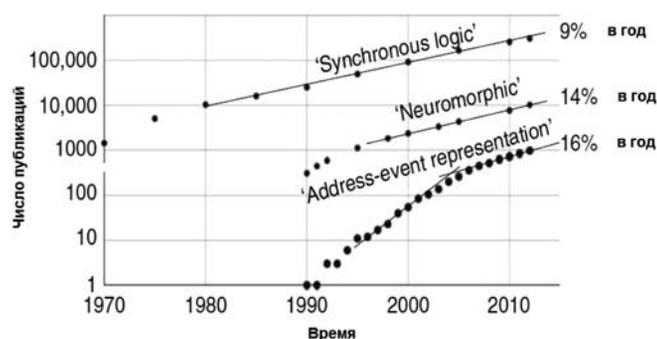


Рис. 1. Рост числа публикаций по теме нейроморфных вычислений ("Neuromorphic") во времени (из Google Scholar) [20]

ция по крайней мере еще одного шага — изменения логического базиса алгоритмов решения задач и, соответственно, элементной базы с переходом от носителя информации в электрических схемах в виде уровней токов и напряжений к носителю информации в виде узких импульсов, подобно тому, как это имеет место в реальной нервной системе.

Ниже перечислены основные проекты по созданию супернейрокомпьютеров за рубежом:

- BICA;
- Big Brain;
- BioRC;
- Blue Brain;
- Brain Initiative;
- BrainscaleS;
- Chist-Era Pneuma;
- Cog Ex-Machina;
- Connection;
- ERS Femmes project;
- ERS Nanobrain;
- Facets;
- FP7 Bion;
- FP7 Nabab;
- Human Brain;
- JFAT;
- MARBLE;
- MHANN;
- MONETA;
- Neu-Neu;
- NeutoDyn;
- Neurogrid;
- NTM 3;
- PEPS project ASME;
- SAND;
- SPAUN;
- SpiNNaker;
- Synapse;
- Проект Irvine Sensors;
- Проект Ливерморской лаборатории;
- Проект Института продвинутых архитектур (Сандиа, Оак Ридж);
- Проект Интел, центр суперкомпьютерных вычислений Сан-Диего, DARPA;
- Проект eLiza самоуправляемой автопомпой компьютерной системы (IBM);
- Проект нейроморфного компьютера с использованием мемристоров (Интел совместно с МТИ).

Один из таких проектов более детально представлен в работе [21]. Точка зрения российской научной школы в этой области представлена в работах [22, 23].

Необходимо отметить революционную роль мемристоров [1, 2] в формировании этапа перехода к экзафлопным вычислениям. Функциональные элементы, подобные мемристорам, были известны еще в 60-е годы прошлого столетия [17, 18], но не выдержали конкуренции с появившимися в начале 70-х годов микропроцессорами.

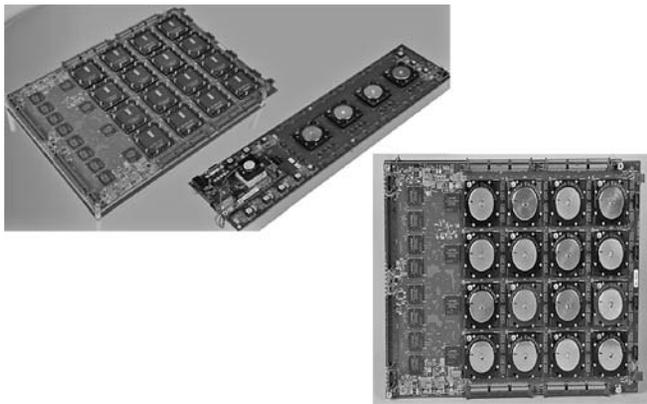


Рис. 2. Разработки фирмы IBM с использованием СБИС TrueNorth [24]

Сейчас мемристоры — детище нанотехнологий — эволюционное, гармоничное развитие вычислительной техники с реализацией следующих принципов:

- переход от классической фоннеймановской архитектуры к распределенной;
- возврат к аналоговой обработке;
- переход к представлению сигналов в виде последовательности узких импульсов;
- совмещение функций памяти и обработки информации;
- переход к нейроматематике — нейросетевым алгоритмам решения задач;
- переход к нейроуправлению — нейросетевым алгоритмам и системам управления сложными системами.

Наиболее передовой разработкой "домемристорной" эпохи, характеризующейся переходом в нейросетевой логический базис и переходом к представлению информации в виде последовательности узких импульсов, но реализованной в виде чисто цифровой СБИС, является разработка СБИС TrueNorth фирмы IBM [24]. Характерные особенности этой СБИС:

- 5,4 млрд транзисторов на чипе;
- 4096 нейросинаптических ядер;
- 1 млн программируемых импульсных нейронов;
- 256 млн настраиваемых синапсов;
- обработка 400×240 -пиксельного видео, 30 кадров в секунду, с потреблением 63 мВт.

На рис. 2 представлен внешний вид разработок фирмы IBM с использованием СБИС TrueNorth. Планируются также изделия, содержащие 256, 1024 и 4096 СБИС TrueNorth с электропотреблением в 256, 1000 и 4000 Вт соответственно (последнее содержит 1 трлн синапсов, что составляет примерно 1 % числа синапсов в мозге человека).

Реализация нейрона с применением мемристоров

В схемной реализации нейрона мемристоры выполняют функцию синапсов — перестраиваемых весовых коэффициентов. Возможно несколько вариантов применения мемристоров для этой цели.

В работе [21] представлен один из вариантов реализации синапса нейрона с применением мемристоров (рис. 3). Вариант принципиальной схемы многовходового нейрона представлен на рис. 4 [25]. На рис. 5, а представлена мостовая схема соединения мемристоров, обеспечивающая реализацию положительных и отрицательных значений весовых коэффициентов в нейронной сети [26], на рис. 5, б — схема синапса с мостом из мемристоров. В работе [27] на основе мостовой схемы приведена схема мемристорного нейрона с несколькими входами.

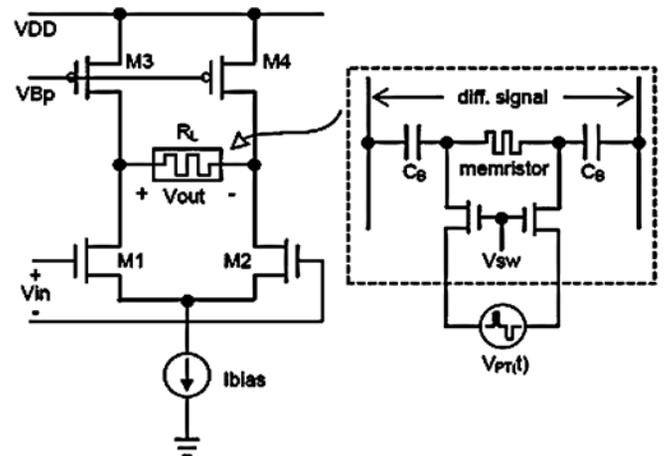


Рис. 3. Реализация синапса нейрона с применением мемристора [21]

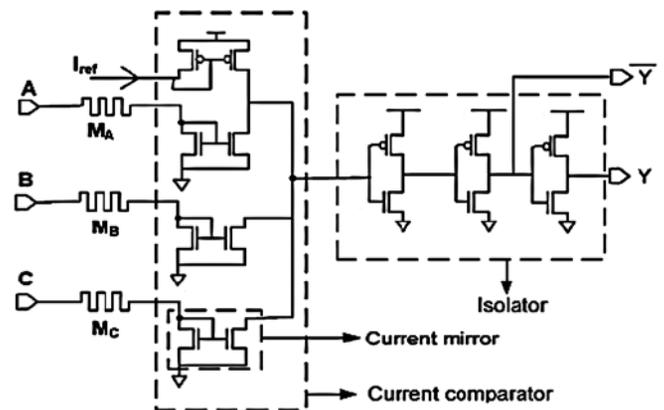


Рис. 4. Схема трехвходового нейрона с применением мемристоров [25]

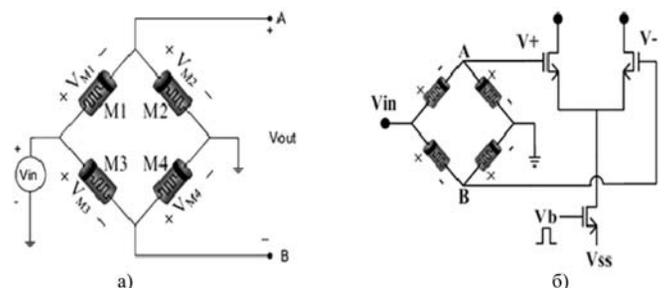


Рис. 5. Схема мемристорного моста (а); схема синапса с мостом из мемристоров (б) [26]

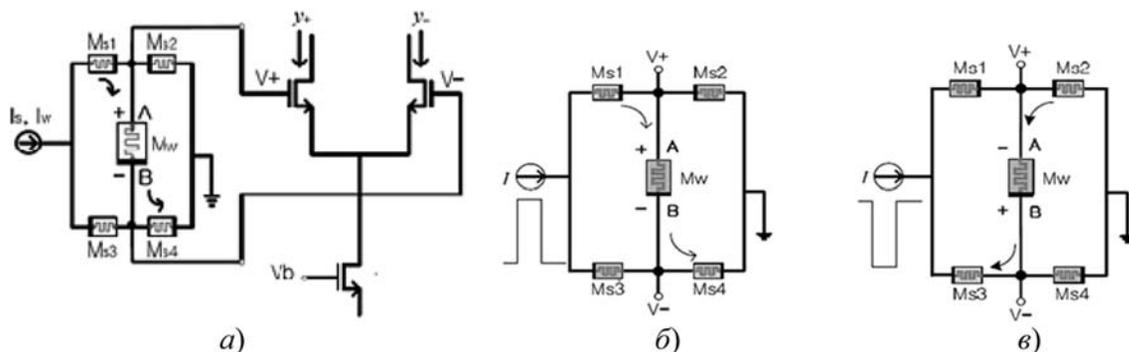


Рис. 6. Вариант реализации в синапсе с применением мемристоров [28] (а), схема изменения знака весового коэффициента в синапсе с применением мемристоров при положительной конфигурации (б) и отрицательная конфигурация (в)

В работе [28] представлен другой вариант соединения мемристоров в схеме синапса (рис. 6, а), а также предлагается схема изменения знака весового коэффициента в синапсе (рис. 6, б, в).

Реализация нейронных сетей с применением мемристоров

Анализ публикаций 2008—2014 гг. показал, что попытки реализации нейронных сетей с применением мемристоров касаются в основном многослойных нейронных сетей следующих видов:

- с полными последовательными связями;
- с обратными связями (рекуррентные нейронные сети);
- клеточные нейронные сети, ориентированные на обработку изображений.

На рис. 7 представлена простейшая двухслойная нейронная сеть с применением мемристоров [26].

В работе [29] предложена (рис. 8) схема реализации весового коэффициента с мемристорами для клеточной нейронной сети. В работе [30] предложена схема применения мемристоров в клеточной нейронной сети CrossNet (рис. 9).

Разработка и реализация принципиальных схем нейронных сетей различных структур с применением мемристоров является предметом дальнейших исследований в ближайшие годы.

Настройка нейронных сетей с применением мемристоров

С нашей точки зрения, работы по реализации алгоритмов настройки нейронных сетей с применением мемристоров имеют самый начальный характер и в основном касаются реализации простейших алгоритмов типа правила Хебба [31]. Ориентация мемристоров на реализацию нейронных сетей с очень большим числом нейронов и настраиваемых коэффициентов делает задачу разработки и реализации

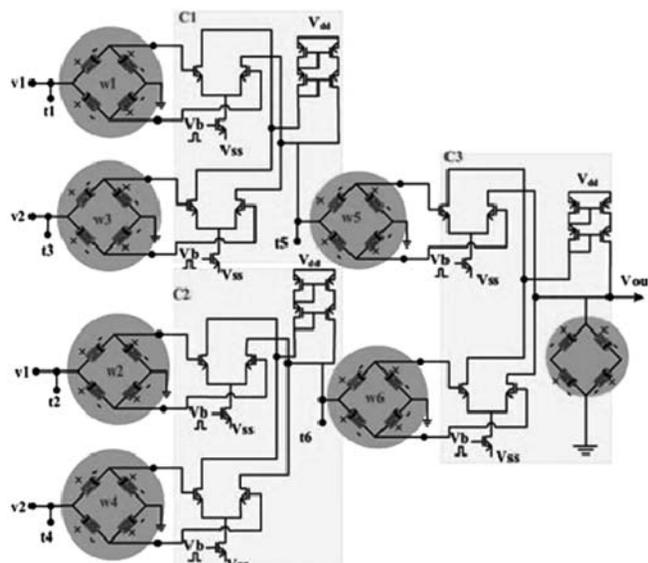


Рис. 7. Схемная реализация двухслойной нейронной сети с мемристорами [26]

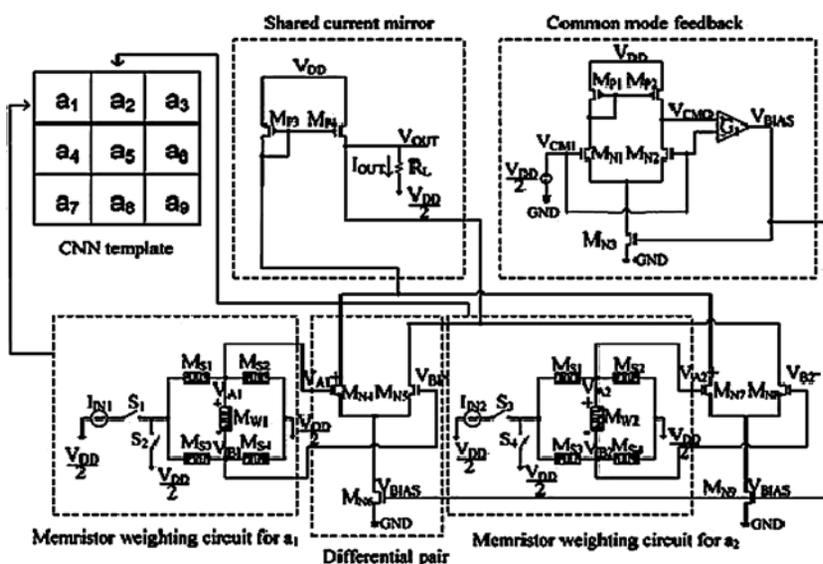


Рис. 8. Схема реализации весового коэффициента с мемристорами для клеточной нейронной сети [29]

алгоритмов настройки важной, сложной, в значительной степени определяемой архитектурой будущих вычислительных систем с применением мемристоров.

В работе [32] представлен аналитический подход к сравнению простейших алгоритмов настройки применительно к мемристормым системам. В [26] представлена схема обучения нейронной сети с при-

менением мемристоров с реализацией блока настройки на хост-ЭВМ (рис. 10, см. четвертую сторону обложки). Такая схема вряд ли жизнеспособна для реализаций с большим числом мемристоров.

Распараллеливание алгоритмов настройки и требование повышения быстродействия должны привести к внутрикристальной реализации алгоритмов обучения на цифровом и аналоговом уровнях.

СБИС-нейрочины с применением мемристоров

Разработка СБИС-нейрочипов с применением мемристоров является одним из важнейших направлений работ в области создания перспективных нейрокомпьютеров. При этом анализ разработок аналоговых и аналого-цифровых нейрочипов должен касаться следующих разделов:

- аналоговые нейрочипы;
- аналого-цифровые нейрочипы;
- клеточные нейрочипы;
- нейрочипы с частотно-импульсным представлением сигналов;
- оптические и оптоэлектронные нейрочипы;
- молекулярные нейрочипы;
- специализированные аналоговые и аналого-цифровые нейрочипы (АЦП, СМАС, обработки изображений, нейроуправления, ассоциативной памяти, обработки речевой информации);
- др.

Типичным примером домемристормой разработки цифровых нейрочипов с частотно-импульсной модуляцией сигналов является схема, представленная на рис. 11 [33]. Необходимо отметить несколько попыток реализации СБИС-нейрочипов с применением мемристоров (цифровых и аналого-цифровых) [34–36] (рис. 12, см. четвертую сторону обложки, 13, 14).

Особое внимание нужно обратить на разработку нейрочипа с применением трехтерминального ферроэлектрического мемристора [37] (рис. 15), в значительной степени идеологически повторяющего разработки мемристоров 60-х годов прошлого века [16, 17].

Прикладные задачи

Любое новое технологическое направление микроэлектроники обречено на гибель, если не будет ориентировано на достаточно массовое производство некоторых элементов. Для мемристоров рынками таких элементов являются

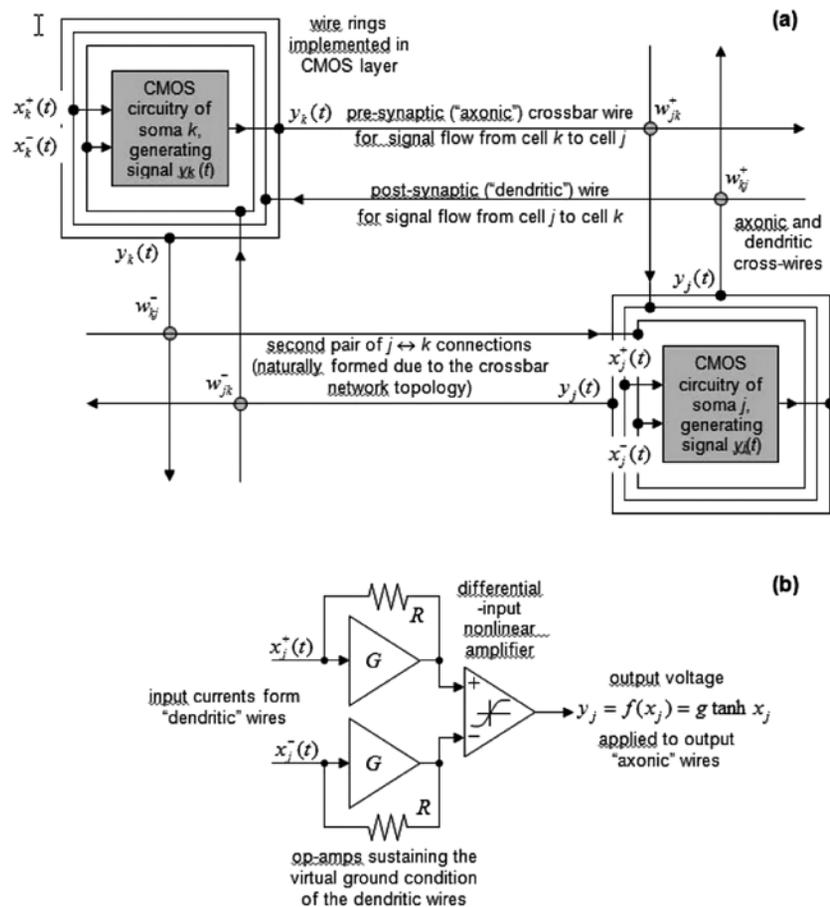


Рис. 9. Связь между одной парой клеток в сети CrossNet [30]

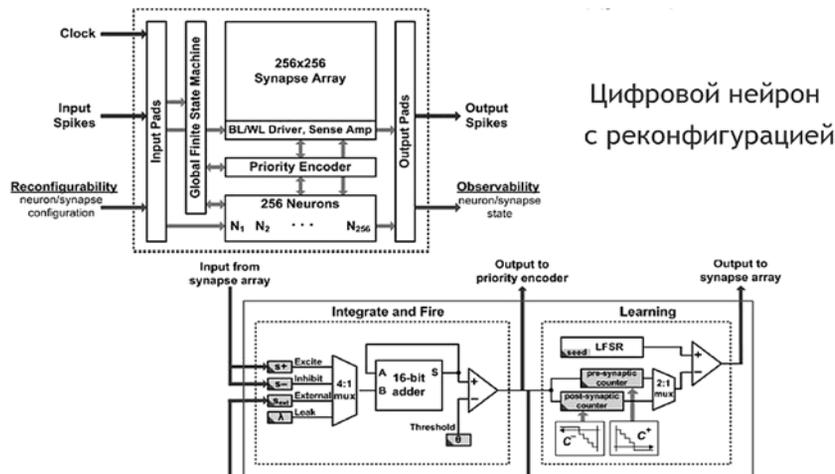


Рис. 11. Пример домемристормых разработок цифровых нейрочипов с частотно-импульсной модуляцией сигналов [33]

перспективные системы памяти и коммутационные системы, в том числе коммутационные системы внутри современных и перспективных ПЛИС.

Большие нейронные сети — главная задача реализации нейрокомпьютеров с применением мемристоров. Как следствие — выбор для решения на мемристорных системах следующих сложных задач:

- обнаружение атак на информационные ресурсы в больших распределенных вычислительных сетях;
- нейрокриптография;
- уравнения математической физики, включая генерацию адаптивных сеток;
- обработка видеоизображений и медиаинформации (проблема big data);
- создание 3D-моделей мозга;
- управление плазмой;
- обработка геномной и протеомной информации;
- любые другие применения нейрокомпьютеров, где необходимо увеличить отношение производительности к стоимости или энергопотреблению.

Заключение

Всю историю развития вычислительной техники сопровождают работы по эмуляции алгоритмов при реализации искусственного интеллекта, принятии решений и нейронных сетей.

Идеология построения нейрокомпьютеров менялась слабо, менялись и развивались технологии реализации.

Российский опыт работ в области нейросетевых технологий и их применений обобщен в монографии [38], где проанализированы российские монографии, кандидатские и докторские диссертации, научно-технические отчеты и научные труды, вышедшие между 1982 и 2010 гг.

Несмотря на то что вычислительная техника за весь большой период существования и развития всегда использовалась для реализации мыслительных функций, только в период 2007—2009 гг. и далее появились работы, в которых показано, что даже самые современные суперЭВМ типа Blue Gene, Fujitsu-K и др. еще крайне недостаточны по производительности для моделирования разделов мозга в реальном времени. Моделирование объемом 50 млрд нейронов (ориентировочно 1/100 реального мозга) с попыткой реализации работы в реальном времени в те-

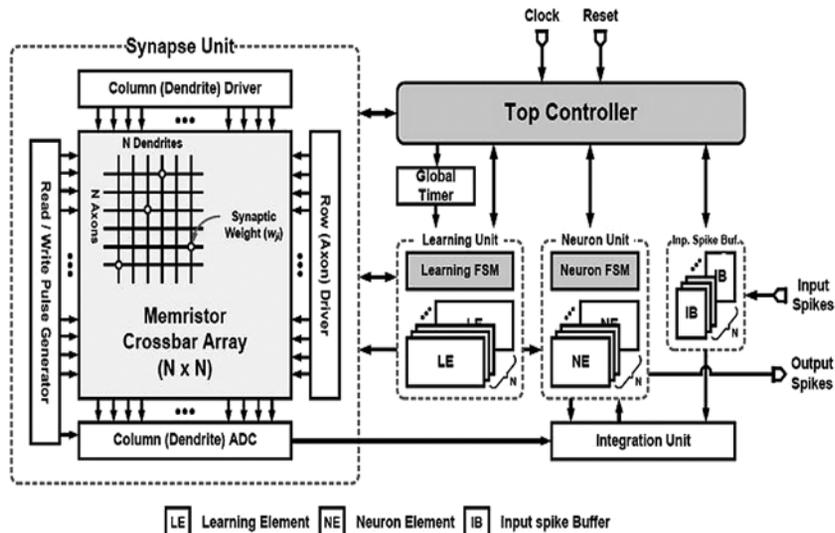


Рис. 13. Вариант архитектуры цифрового нейрочипа с применением мемристоров [35]

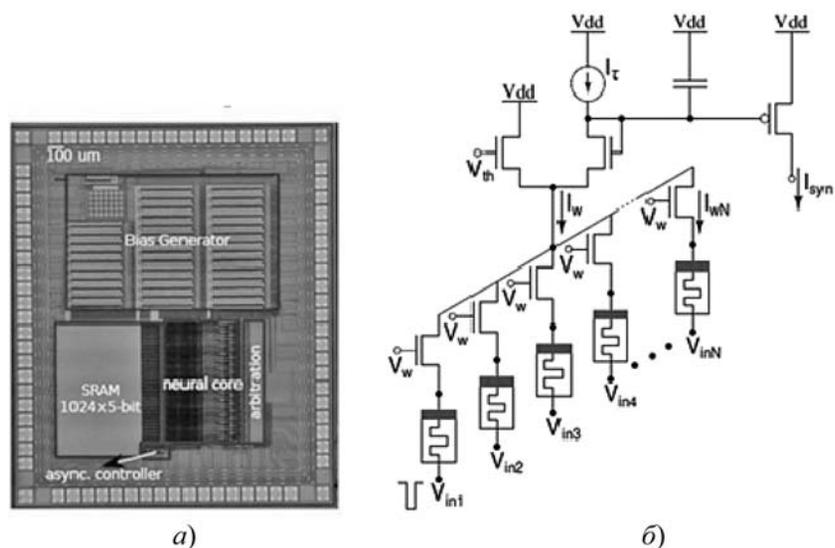


Рис. 14. Аналого-цифровой нейрочип с применением мемристоров [36]: а — микрофотография нейрочипа; б — схема синапса

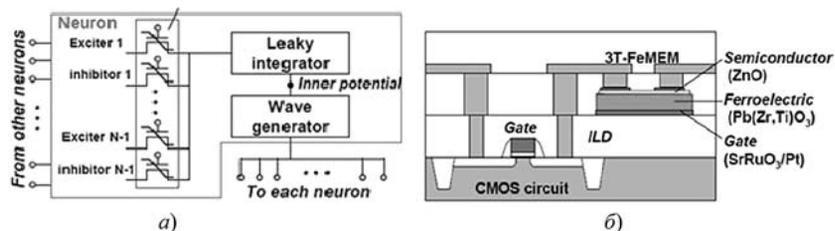


Рис. 15. Нейрочип с применением трехтерминального ферроэлектрического мемристора [37]: а — схема нейрона; б — интеграция 3Т-FeMeМ и CMOS

чение 1 с на суперЭВМ Fujitsu-K заняло около 40 мин при использовании открытого программного обеспечения NEST (NEural Simulation Technology). Эти исследования показывают, как далека современная технология (и думаем, что даже перспективная на ближайшее время) от технологии realiza-

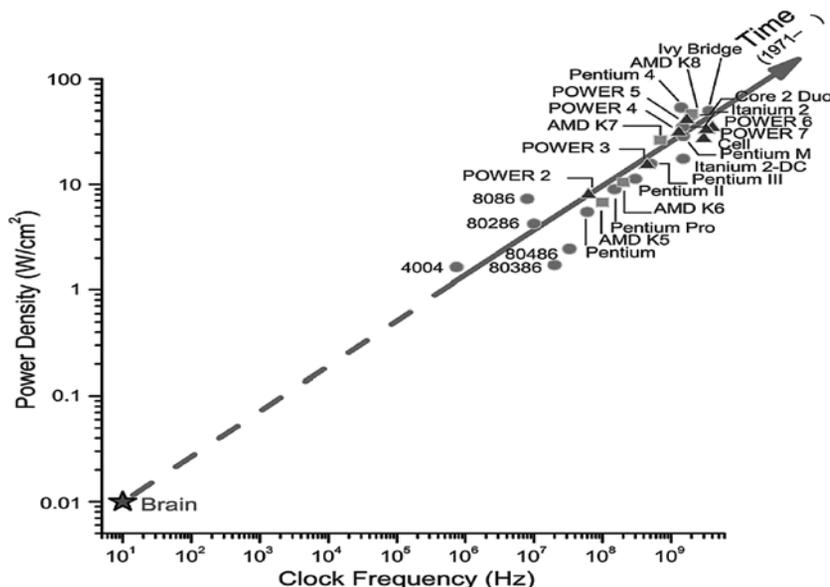


Рис. 16. Тенденции развития современной микроэлектроники классической архитектуры в пространстве [потребляемая мощность — частота работы] [24]

ции аналогов реального живого мозга и как далеки мы от настоящего, глубокого понимания принципов работы живого мозга, которые мы сможем использовать при построении искусственных систем. Более того, на рис. 16 приведена схема [24], показывающая тенденции развития современной микроэлектроники классической архитектуры в пространстве [потребляемая мощность — частота работы].

Мемристоры — это очередной этап эволюционного, гармоничного развития вычислительной техники, в котором за несколько предыдущих десятилетий проведена фундаментальная подготовка к построению высокопроизводительных вычислительных систем, реализующих следующие принципы:

- переход от классической фоннеймановской архитектуры к распределенной;
- возврат в части реализации алгоритмов решения задач к аналоговой обработке в целях увеличения быстродействия при контролируемой точности;
- переход от представления сигналов в виде уровней токов и напряжений к представлению сигналов в виде частоты последовательности узких импульсов;
- переход к архитектурам, в которых функции памяти и обработки хотя бы частично совмещены в отличие от классических ЭВМ с полностью распределенными функциями памяти и обработки;
- переход к нейроматематике — нейросетевым алгоритмам решения задач;
- переход к нейроуправлению — нейросетевым алгоритмам и системам управления сложными динамическими объектами.

Российская научная школа обладает необходимым потенциалом для реализации такой амбици-

озной задачи, как создание нейрокомпьютера на базе мемристоров. Многие коллективы уже ведут практические исследования и организационную работу для решения такой задачи. Нужно также отметить, что вузы (например, МФТИ, ВШЭ и ряд других) уже сейчас предлагают студентам курсовые, дипломные работы в этой области, таким образом готовя будущих ученых и инженеров в этой области.

Список литературы

1. Киви Б. К точке критического перехода // 3D-News (Daily Digital Digest). 29.05.2013.
2. McKenzie A. et al. Toward Exascale Computing through Neuromorphic Approaches // Sandia Report, Sand 2010 — 6312. September 2010.
3. Галушкин А. И. Стратегия развития современных супернейрокомпьютеров на пути к экзафлопным вычислениям // Приложение к журналу "Информационные технологии", 2012. № 2. 32 с.
4. Галушкин А. И. Многослойные системы распознавания образов. М.: Изд. МИЭМ, 1970. 168 с.
5. Галушкин А. И. Синтез многослойных систем распознавания образов. М.: Энергия, 1974. 367 с.
6. Омату С., Халид М., Юсоф Р. Нейроуправление и его приложения. 2-е изд. М.: ИПРЖР, 2000. 272 с.
7. Галушкин А. И., Цыпкин Я. З. Нейронные сети: история развития теории. Кн. 5. Сер. "Нейрокомпьютеры и их применение". М.: Радиотехника, 2001. 840 с.
8. Галушкин А. И. Теория нейронных сетей. Пекин: Изд-во университета Синьхуа (на китайском языке), 2003. 294 с.
9. Galushkin A. I. Neural Network Theory. NY: Springer, 2007. 416 p.
10. Галушкин А. И. Нейронные сети: основы теории. М.: Горячая линия — Телеком, 2010. 496 с.
11. Нейроматематика. Кн. 6. Сер. "Нейрокомпьютеры и их применение". М.: Радиотехника, 2002. 448 с.
12. Галушкин А. И. Нейроматематика (проблемы развития) // Нейрокомпьютер, 2003. № 1. С. 1—40.
13. Галушкин А. И. О методике решения задач в нейросетевом логическом базисе // Приложение к журналу "Информационные технологии". 2006. № 9. 24 с.
14. Терехов В. А., Ефимов Д. В., Тюкин И. Ю. Нейросетевые системы управления. Кн. 8. Сер. "Нейрокомпьютеры и их применение". М.: Радиотехника, 2004. 480 с.
15. Галушкин А. И. Основы нейроуправления // Приложение к журналу "Информационные технологии". 2002. № 10. 24 с.
16. Аналоговые запоминающие и адаптивные элементы / Под ред. Б. С. Сотскова. М.: Энергия, 1973, 260 с.
17. Трейер В. В., Елизаров А. Б. Электрохимические интегрирующие и аналоговые запоминающие элементы. М.: Энергия, 1973. 96 с.
18. Галушкин А. И. Нейрокомпьютеры. Кн. 1. Сер. "Нейрокомпьютеры и их применение". М.: Радиотехника, 2000, 528 с.
19. Аляутдинов М. А., Казанцев П. А., Галушкин А. И., Остапенко Г. П. Нейрокомпьютеры: от программной к аппаратной реализации. М.: Горячая линия — Телеком, 2006, 152 с.
20. Event-Based Neuromorphic Systems (Eds by Shih-Chii Liu, Tobi Delbruck, Giacomo Indiveri, Adrian Whatley, Rodney Douglas). UK: Wiley, 2015. 440 p.
21. Shin S., Kim K., Kang S. M. Memristors and Their Applications for Nanocomputing // IEEE Trans. on Nanotechnology, Mar. 2011.
22. Алюшин А. В., Галушкин А. И., Завалин Ю. В., Занавескин М. Л., Немудров В. Г., Паптюхин Д. В. Суперкомпьютеры с использованием мемристоров // Суперкомпьютеры. 2014. № 3. С. 48—49.

23. **Галушкин А. И.** Мемристоры в развитии высокопроизводительной вычислительной техники // Информационные технологии. 2015. № 2. С. 146—156.

24. **Merolla P. A.** et al. A million spiking-neuron integrated circuit with a scalable communication network and interface // Science, August 2014. N. 345, Iss. 6197. P. 668.

25. **Manem H., Rajendran J., Rose G. S.** Stochastic Gradient Descent Inspired Training Technique for a CMOS/Nano Memristive Trainable Threshold Gate Array // IEEE Transactions on Circuits and Systems — I: Regular Papers. 2012. Vol. 59, N. 5. P. 1051—1060.

26. **Adhikari S. P., Changju Y., Hyongsuk K., Chua, L. O.** Memristor Bridge Synapse-Based Neural Network and its Learning // IEEE Transactions on Neural Networks and Learning Systems. 2012. Vol. 23, N. 9. P. 1426—1435.

27. **Sah M. P., Yang C., Budhathoki R. K., Kim H.** Features of memristor emulator-based artificial neural synapses // IEEE International Symposium Circuits and Systems (ISCAS). 19—23 May 2013, Beijing, 2013. P. 421—424.

28. **Sah M. P., Yang C., Kim H., Chua L. O.** Memristor Circuit for Artificial Synaptic Weighting of Pulse Inputs // IEEE International Symposium Circuits and Systems (ISCAS), 20—23 May 2012, Seoul, Korea (South). 2012. P. 1604—1607.

29. **Kim Y.-S., Min K.-S.** Synaptic weighting circuits for Cellular Neural Networks // 13th International Workshop Cellular Nanoscale Networks and Their Applications (CNNA), 29—31 Aug. 2012. P. 1—6

30. **Walls T. J., Likharev K. K.** Self-Organization in Autonomous, Recurrent, Firing-Rate CrossNets With Quasi-Hebbian Plasticity // IEEE Transactions on Neural Networks and Learning Systems. 2014. Vol. 25, N. 4. P. 819—824.

31. **Hebb D. O.** The organization of Behavior: A Neuropsychological Theory New York: Wiley, 1949. 335 p.

32. **Gorchetnikov A., Versace M., Ames H., Chandler B., Leveille J., Livitz G., Mingolla E., Snider G., Amerson R., Carter D., Abdalla H., Qureshi M. S.** Review and unification of learning framework in Cog Ex Machina platform for memristive neuromorphic hardware // Proceedings of International Joint Conference on Neural Networks, San Jose, California, USA, July 31 — August 5. 2011. P. 2601—2608.

33. **Seo J.-S., Brezzo B., Yong L., Parker B. D., Esser S. K., Montoyo R. K., Rajendran B., Tierno J. A., Chang L., Modha D. S., Friedman D. J.** A 45 nm CMOS neuromorphic chip with a scalable architecture for learning in networks of spiking neurons // IEEE Custom Integrated Circuits Conference (CICC), 2011. P. 1—4.

34. **Serrano-Gotarredona T., Prodromakis T., Linares-Barranco B.** A Proposal for Hybrid Memristor-CMOS Spiking Neuromorphic Learning Systems // IEEE Circuits and Systems Magazine. 2013. Vol. 13, Iss. 2. P. 74—88.

35. **Kim Y., Zhang Y., Li P.** A digital neuromorphic VLSI architecture with memristor crossbar synaptic array for machine learning // 2012 IEEE International SOC Conference (SOCC). 2012. P. 328—333.

36. **Azghadi M. R., Moradi S., Indiveri G.** Programmable neuromorphic circuits for spike-based neural dynamics // 2013 IEEE 11th International New Circuits and Systems Conference (NEWCAS). 2013. P. 1—4.

37. **Kaneko Y., Nishitani Yu., Ueda M., Tsujimura A.** Neural network based on a three-terminal ferroelectric memristor to enable on-chip pattern recognition // Symposium VLSI Technology (VLSIT). 2013. P. T238—T239.

38. **Галушкин А. И., Симоров С. Н.** Нейросетевые технологии в России (1982—2010 г.). М.: Горячая линия — Телеком, 2011. 316 с.

A. I. Galushkin, Prof., Deputy Head of Chair of the Moscow Institute of Physics and Technology, Dolgoprudny, neurocomputer@yandex.ru, Prof. of High School of Economics, agalushkin@hse.ru,

D. V. Pantiukhin, Senior Lecturer of High School of Economics, dpantiukhin@hse.ru, Lecturer of Chair of the Moscow Institute of Physics and Technology, Dolgoprudny, dim_beavis@mail.ru

Supercomputers and Memristors

The article describes the authors' opinion about the perspectives of development of one area of supercomputers — neurocomputers — with the advent of memristors. These perspectives are determined by ultra-high parallelism in connection with the transition to the neural network logical basis, the sharp decline in energy consumption in connection with the transition to the presentation of information in the form of a frequency sequence of narrow pulses, the transition to an analog implementation of the algorithms with a sharp increase in speed. Examples of such development of neurochips and neurosystems from recent years are described.

Due to developments in neural network theory, neuromathematics and neurocontrol, micro- and nanoelectronics and parallel hardware, it is possible today to change VonNeuman computing paradigm to neural network one. Neurocomputers with memristor will be important step to exascale computing.

This article show historical and today progress in neurocomputing area in a world and some projects of neurocomputers with memristors, including DARPA Neural network Study Projects, IBM TrueNorth neurochip and many approaches to memristor-based neural networks.

Authors postulate that major applications of such memristor-based neurocomputers will be:

- information security (intrusion, viruses and spam detection),
- (neuro-) cryptography tasks,
- mathematical physics tasks including adaptive computing mesh generation,
- image and 3D objects processing (detection, classification, clustering and oth.),
- brain modelling,
- and others applications.

It is also discussed about neurocomputing problem in Russia.

Authors hope that this article will stimulate discussion about such difficult problem as neurocomputers with memristor within Russia and abroad.

Keywords: neurocomputer, neurochip, memristor, exaflop computing

References

1. **Kivi B. K.** tochke kriticheskogo perekhoda (Toward critical point). *3D-News* (Daily Digital Digest). 29.05.2013.
2. **McKenzie A.** et al. Toward Exascale Computing through Neuromorphic Approaches. *Sandia Report*, Sand 2010 — 6312. September 2010.
3. **Galushkin A. I.** Strategiya razvitiya sovremennykh superneyrokomp'yutеров na puti k ekzaflopnyim vychisleniyam (Strategy of development modern superneurocomputers toward exaflop computing), *Prilozheniye k zhurnalu "Informatsionnyye tekhnologii"* (Appendix to journal "Information technology"), 2012, no. 2, 32 p. (in Russian).
4. **Galushkin A. I.** *Mnogosloynnye sistemy raspoznavaniya obrazov* (Multilayer systems for pattern recognition). Moscow: MIJeM, 1970, 168 p. (in Russian).
5. **Galushkin A. I.** *Sintez mnogosloynnykh sistem raspoznavaniya obrazov* (Synthesis of multilayer systems for pattern recognition). Moscow: Energy, 1974. 367 p. (in Russian).
6. **Omatu S., Khalid M., Yusof R.** *Neyroupravleniye i yego prilozheniya* (Neurocontrol and its applications). Moscow: IPRZHR, 2000, 272 p. (in Russian).
7. **Galushkin A. I., Tsyppin Ja. Z.** *Neyronnyye seti: istoriya razvitiya teorii. Kn. 5. Ser. "Neyrokomp'yutery i ikh primeneniye"* (Neural networks: history of theory evolution. Book 5. Ser. "Neurocomputers and their application"), Moscow: Radiotekhnika, 2001, 840 p. (in Russian).
8. **Galushkin A. I.** *Teoriya neyronnykh setey* (Theory on neural networks). Beijing: Sinhua Univ., 2003, 294 p. (in Chinese).
9. **Galushkin A. I.** *Neural Network Theory*. NY: Springer, 2007. 416 p.
10. **Galushkin A. I.** *Neyronnyye seti: osnovy teorii* (Neural networks: theory basics). Moscow: Goryachaya liniya — Telekom (Hotline — Telekom), 2010. 496 p. (in Russian).
11. **Neyromatematika. Kn. 6, Ser. "Neyrokomp'yutery i ikh primeneniye" (Neuromathematics. Book 6. Ser. "Neurocomputers and their application"). Moscow: Radiotekhnika, 2002. 448 p. (in Russian).**
12. **Galushkin A. I.** *Neyromatematika (problemy razvitiya)* (Neuromathematics (problems of evolution)) // Moscow: Neurocomputer, 2003. No. 1, pp. 1—40. (in Russian).
13. **Galushkin A. I.** O metodike resheniya zadach v neyrosetevom logicheskom bazise (About methodology of task solution in neural network logical basis), *Prilozheniye k zhurnalu "Informatsionnyye tekhnologii"* (Appendix to journal "Information technology"), 2006, no. 9, 24 p. (in Russian).
14. **Terekhov V. A., Efimov D. V., Tyukin I. Yu.** *Neyrosetevyye sistemy upravleniya. Kn. 8. Ser. "Neyrokomp'yutery i ikh primeneniye"* (Neural control systems. Book 8. Ser. "Neurocomputers and their application"). Moscow: Radiotekhnika, 2004. 480 p. (in Russian).
15. **Galushkin A. I.** Osnovy neyroupravleniya (Basics of neurocontrol), *Prilozheniye k zhurnalu "Informatsionnyye tekhnologii"* (Appendix to journal "Information technology"). 2002, no. 10, 24 p. (in Russian).
16. **Analogovyye zapominayushchiye i adaptivnyye elementy** (Analogous memorizing and adaptive elements). (Ed by B.S. Sotskov) Moscow: Energy, 1973. 260 p. (in Russian).
17. **Trejter V. V., Elizarov A. B.** *Elektrokhimicheskiye integriruyushchiye analogovyye zapominayushchiye elementy* (Electrochemical integral and analogous memorizing elements). Moscow: Energy, 1973. 96 p. (in Russian).
18. **Galushkin A. I.** *Neyrokomp'yutery. Kn. 1. Ser. "Neyrokomp'yutery i ikh primeneniye"* (Neurocomputers. Book 1. Ser. "Neurocomputers and their application"). Moscow: Radiotekhnika, 2000, 528 p. (in Russian).
19. **Alyautdinov M. A., Kazantsev P. A., Galushkin A. I., Ostapenko G. P.** *Neyrokomp'yutery: ot programmnoy k apparatnoy realizatsii* (Neurocomputers: from software to hardware realization). Moscow: Goryachaya liniya — Telekom (Hotline — Telekom), 2006. 152 p. (in Russian).
20. **Event-Based Neuromorphic Systems** (Eds by Shih-Chii Liu, Tobi Delbruck, Giacomo Indiveri, Adrian Whatley, Rodney Douglas). UK: Wiley, 2015. 440 p.
21. **Shin S., Kim K., Kang S. M.** Memristors and Their Applications for Nanocomputing, *IEEE Trans. on Nanotechnology*, Mar. 2011.
22. **Aljushin A. V., Galushkin A. I., Zavalin Ju. V., Zhanavskiy M. L., Nemudrov V. G., Pantiukhin D. V.** Superkomp'yutery s ispolzovaniyem memristorov (Supercomputers with memristors), *Moscow: Supercomputers*, 2014, no. 3, pp. 48—49 (in Russian).
23. **Galushkin A. I.** Memristory v razvitiy vysokoproizvoditelnoy vychislitelnoy tekhniki (Memristors in development of high performance computers), *Moscow: Informatsionnyye tekhnologii* (Information technology), 2015, no. 2, pp. 146—156 (in Russian).
24. **Merolla P. A.** et al. A million spiking-neuron integrated circuit with a scalable communication network and interface. *Science*, August 2014, no. 345, Iss. 6197, pp. 668.
25. **Manem H., Rajendran J., Rose G. S.** Stochastic Gradient Descent Inspired Training Technique for a CMOS/Nano Memristive Trainable Threshold Gate Array, *IEEE Transactions on Circuits and Systems — I: Regular Papers*, 2012, vol. 59, no. 5, pp. 1051—1060.
26. **Adhikari S. P., Changju Y., Hyongsuk K., Chua L. O.** Memristor Bridge Synapse-Based Neural Network and its Learning, *IEEE Transactions on Neural Networks and Learning Systems*, 2012, vol. 23, no. 9, pp. 1426—1435.
27. **Sah M. P., Yang C., Budhathoki R. K., Kim H.** Features of memristor emulator-based artificial neural synapses, *IEEE International Symposium Circuits and System (ISCAS)*, 19—23 May 2013, Beijing, 2013, pp. 421—424.
28. **Sah M. P., Yang C., Kim H., Chua L. O.** Memristor Circuit for Artificial Synaptic Weighting of Pulse Inputs, *IEEE International Symposium Circuits and Systems (ISCAS)*, 20—23 May 2012, Seoul, Korea (South), 2012, pp. 1604—1607.
29. **Kim Y.-S., Min K.-S.** Synaptic weighting circuits for Cellular Neural Networks, *13th International Workshop Cellular Nanoscale Networks and Their Applications (CNNA)*, 29—31 Aug. 2012, pp. 1—6.
30. **Walls T. J., Likharev K. K.** Self-Organization in Autonomous, Recurrent, Firing-Rate CrossNets With Quasi-Hebbian Plasticity, *IEEE Transactions on Neural Networks and Learning Systems*, 2014, vol. 25, no. 4, pp. 819—824.
31. **Hebb D. O.** *The organization of Behavior: A Neuropsychological Theory*. New York: Wiley, 1949, 335 p.
32. **Gorchetchnikov A., Versace M., Ames H., Chandler B., Leveille J., Livitz G., Mingolla E., Snider G., Amerson R., Carter D., Abdalla H., Qureshi M. S.** Review and unification of learning framework in Cog Ex Machina platform for memristive neuromorphic hardware, *Proceedings of International Joint Conference on Neural Networks, San Jose, California, USA, July 31—August 5, 2011*, pp. 2601—2608.
33. **Seo J.-S., Brezzo B., Yong L., Parker B. D., Esser S. K., Montoye R. K., Rajendran B., Tierno J. A., Chang L., Modha D. S., Friedman D. J.** A 45nm CMOS neuromorphic chip with a scalable architecture for learning in networks of spiking neurons, *IEEE Custom Integrated Circuits Conference (CICC)*, 2011, pp. 1—4.
34. **Serrano-Cotarredona T., Prodromakis T., Linares-Barranco B.** A Proposal for Hybrid Memristor-CMOS Spiking Neuromorphic Learning Systems, *IEEE Circuits and Systems Magazine*, 2013, vol. 13, Iss. 2, pp. 74—88.
35. **Kim Y., Zhang Y., Li P.** A digital neuromorphic VLSI architecture with memristor crossbar synaptic array for machine learning 2012, *IEEE International SOC Conference (SOCC)*, 2012, pp. 328—333.
36. **Azghadi M. R., Moradi S., Indiveri G.** Programmable neuromorphic circuits for spike-based neural dynamics, 2013, *IEEE 11th International New Circuits and Systems Conference (NEWCAS)*, 2013, pp. 1—4.
37. **Kaneko Y., Nishitani Yu., Ueda M., Tsujimura A.** Neural network based on a three-terminal ferroelectric memristor to enable on-chip pattern recognition, *Symposium VLSI Technology (VLSIT)*, 2013, pp. T238—T239.
38. **Galushkin A. I., Simorov S. N.** *Neyrosetevyye tekhnologii v Rossii (1982—2010 g.)* (Neural network technology in Russia (1982—2010)). Moscow: Goryachayaliniya — Telekom (Hotline — Telekom), 2011, 316 p. (in Russian).

А. И. Задаянчук¹, студент, e-mail: alex.goncharov@phystech.edu,
 М. С. Попова¹, студент, В. В. Стрижов², д-р физ.-мат. наук, вед. науч. сотр.

¹ Московский физико-технический институт

² Вычислительный центр им. А. А. Дородницына

Выбор оптимальной модели классификации временных рядов

Решается задача построения модели для точной и устойчивой классификации физической активности человека по временным рядам. Нейронные сети с недостаточным или избыточным числом нейронов обладают недостаточной обобщающей способностью и могут давать прогнозы, неустойчивые относительно изменения состава выборки. Исследуются модели из класса двухслойных нейронных сетей. Рассматриваются модели с избыточно сложной структурой. Структура модели оптимизируется путем удаления из нее наборов параметров — нейронов. Для оптимизации структуры нейронной сети и обеспечения устойчивости предлагается алгоритм генетического типа. Новизна работы заключается в том, что вероятность удаления наборов параметров определяется дисперсией параметров. В вычислительном эксперименте модели, порождаемые предложенной стратегией, сравниваются по двум критериям качества: точности и устойчивости. Модели оптимизируются на выборках, полученных путем выделения признаков из временных рядов.

Ключевые слова: классификация, нейронные сети, устойчивость, критерии прореживания, генетический алгоритм

Введение

Работа посвящается исследованию методов построения нейронной сети оптимальной структуры для решения задачи классификации временных рядов. В работе исследуются и сравниваются методы изменения размерности пространства параметров двухслойных нейронных сетей. При уменьшении размерности пространства параметров значительно уменьшается время оптимизации параметров, увеличивается обобщающая способность нейронной сети и, как следствие, уменьшается значение функции ошибки на контрольной выборке [1].

Оптимизировать размерность пространства можно на разных уровнях — на уровне нейронов (наборов параметров) [2] и отдельных параметров [3]. Структурные параметры модели — это параметры, управляющие включением нейрона в модель. В данной работе размерность пространства параметров оптимизируется на уровне нейронов путем изменения значений структурных параметров. Предложено несколько способов такой оптимизации: прореживание (*network pruning*) [4], наращивание (*network growing*) [5] и пошаговое чередование наращивания и прореживания [6, 7]. В настоящей работе рассматривается оптимизация размерности с помощью прореживания. Базовыми алгоритмами прореживания нейронных сетей являются оптимальные прореживания (англ. *optimal brain damage* [8] и *optimal brain surgery* [9]), основанные на вычислениях вторых производных функции ошибки.

Устойчивая и оптимальная модель описывается с помощью генетического алгоритма на уровне нейронов [10, 11] путем оптимизации структурных параметров. Базовые алгоритмы [8, 9] находят локальный минимум функции ошибки. В случае же, когда функция ошибки имеет значительное

число локальных минимумов, найденный минимум может не совпадать с глобальным. Для нахождения глобального минимума используется алгоритм отбора моделей путем случайного подбора, комбинирования и вариации структурных параметров [10] набора нейронных сетей. Вероятность комбинирования и вариации структурных параметров нейронной сети тем меньше, чем больше показатель выпуклости, используемый в *optimal brain damage* [8].

В вычислительном эксперименте рассматривается задача классификации физической активности человека по измерениям акселерометра. Эта задача решалась в исследованиях [12, 13] с помощью нейронных сетей. В вычислительном эксперименте оцениваются значения критериев качества для нейронных сетей, порождаемых предложенной стратегией. Временные ряды предварительно обрабатываются двумя способами — экспертным порождением признаков [12] и вводом метрики выравнивания временных рядов, с последующим выделением признаков, полученных как расстояние до центров классов [14].

1. Постановка задачи классификации

Дана выборка $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{t}_i)\}$, $i \in \mathcal{J} = \{1, \dots, m\}$, состоящая из m объектов, каждый из которых описывается n признаками $\mathbf{x}_i \in \mathbb{R}^n$ и принадлежит одному из z классов $\mathbf{t}_i \in \{0, 1\}^z$. Задано разбиение множества индексов выборки $\mathcal{J} = \mathcal{L} \sqcup \mathcal{C}$ на обучающую $(\mathbf{x}_l, \mathbf{t}_l)$, где $l \in \mathcal{L}$, и контрольную $(\mathbf{x}_c, \mathbf{t}_c)$, где $c \in \mathcal{C}$. Необходимо выбрать наиболее точную и при этом устойчивую модель классификации.

Определение 1. Моделью назовем отображение:

$$\mathbf{f}(\mathbf{w}, \mathbf{X}) \mapsto \mathbf{y}, \quad (1)$$

$$\mathbf{w} = [w_1, \dots, w_j, \dots, w_J]^T, j \in \mathcal{J} = \{1, \dots, J\},$$

где \mathbf{w} — вектор параметров модели, $\mathbf{X} \in \mathbb{R}^{n \times m}$ — матрица объект-признак, $\mathbf{y} \in \{0, 1\}^z$ — зависимая переменная.

Предполагается, что переменная \mathbf{y} — мультиномиально распределенная случайная величина, а переменная \mathbf{w} имеет нормальное распределение:

$$\mathbf{w} \sim \mathcal{N}(0, \mathbf{A}^{-1}), \quad (2)$$

\mathbf{A}^{-1} — ковариационная матрица. В данной работе рассматриваются модели \mathbf{f} , принадлежащие классу двуслойных нейронных сетей с функциями активации **tanh** и **softmax**

$$\mathbf{a}(\mathbf{x}) = \mathbf{W}_2^T \tanh(\mathbf{W}_1^T \mathbf{x}), \quad (3)$$

$$\mathbf{p}(\mathbf{x}) = \frac{\exp(\mathbf{a}(\mathbf{x}))}{\sum_I \exp(a_I(\mathbf{x}))}. \quad (4)$$

Вектор \mathbf{p} интерпретируется как вектор вероятностей: p_ξ есть вероятность того, что вектор \mathbf{x} принадлежит классу с номером ξ :

$$\mathbf{p}(\mathbf{x}) = \{p_\xi\}, 0 \leq p_\xi \leq 1, \sum p_\xi = 1, \xi = 1, \dots, z.$$

Под вектором параметров двуслойной нейронной сети будем понимать $\mathbf{w} = \text{vec}(\mathbf{W}_1^T | \mathbf{W}_2^T)$, где $\mathbf{W}_1, \mathbf{W}_2$ — присоединенные матрицы весов первого и второго слоев нейронной сети. Вектор $\mathbf{y} = [y_1, \dots, y_\xi, \dots, y_z]^T$ определим следующим образом:

$$y_\xi = \begin{cases} 1, & \text{если } \xi = \arg \max_{\xi \in \{1, \dots, z\}} (p_\xi), \\ 0 & \text{иначе.} \end{cases} \quad (5)$$

Моделью \mathbf{f} является суперпозиция функций (3)–(5). В данной работе мы будем исследовать модели, отличающиеся друг от друга на структурном уровне, а именно — содержащие разное число групп связей — нейронов.

Определение 2. *Нейроном u_k назовем k -ю компоненту вектор-функции $\tanh(\mathbf{W}_1^T \mathbf{x})$ — сомножитель (3).*

Каждый нейрон задается весами в соответствующей строке матрицы \mathbf{W}_1^T .

Определение 3. *Нейрон назовем неактивным, если $u_k = 0$.*

Нейрон неактивен, если k -я строка матрицы \mathbf{W}_1^T нулевая.

Определение 4. *Нейронной структурой $\mathcal{A} = \{k, u_k \neq 0\}$ модели \mathbf{f} назовем множество активных нейронов.*

Каждая нейронная структура \mathcal{A} однозначно задает некоторую модель (1):

$$\mathbf{f}_{\mathcal{A}}: \hat{\mathbf{w}}_{\mathcal{A}} \in \mathbb{R}^k,$$

где $\mathbf{f}_{\mathcal{A}}$ — модель со структурой \mathcal{A} , а $\hat{\mathbf{w}}_{\mathcal{A}} \in \mathbb{R}^k$ — оптимальный вектор параметров модели $\mathbf{f}_{\mathcal{A}}$, определение которому будет дано ниже. Объединение всех $\mathbf{f}_{\mathcal{A}}$ назовем множеством допустимых моделей:

$$\mathfrak{F} = \bigcup_{\mathcal{A} \subseteq J} \{\mathbf{f}_{\mathcal{A}}\}. \quad (6)$$

Оптимальную модель $\hat{\mathbf{f}}_{\mathcal{A}}$ будем выбирать из множества допустимых моделей $\mathbf{f}_{\mathcal{A}}$.

В качестве функции ошибки выберем функцию:

$$S(\mathbf{w}|\mathcal{L}) = - \sum_{i \in \mathcal{L}} \sum_{\xi=1}^z t_{i\xi} \ln(p_\xi(\mathbf{x}_i, \mathbf{w})). \quad (7)$$

Определение 5. *Устойчивостью $\eta = \eta(\hat{\mathbf{w}})$ модели \mathbf{f} с вектором параметров \mathbf{w} назовем число η , равное числу*

обусловленности матрицы \mathbf{A} , т. е. $\eta(\hat{\mathbf{w}}) = \frac{\lambda_{\max}}{\lambda_{\min}}$, где

λ_{\max} — максимальное, а λ_{\min} — минимальное собственные числа матрицы \mathbf{A} .

Чем лучше обусловлена матрица \mathbf{A} , тем более устойчива модель.

Матрица ковариации вычисляется с учетом предположения (2) о нулевом математическом ожидании вектора параметров \mathbf{w} :

$$\begin{aligned} \mathbf{A}^{-1} &= \text{cov}(\mathbf{W}) = \\ &= \mathbb{E}(\mathbf{W}^T \mathbf{W}) - \mathbb{E}(\mathbf{W})\mathbb{E}(\mathbf{W}^T) = \mathbb{E}(\mathbf{W}^T \mathbf{W}), \end{aligned}$$

где \mathbf{W} — это матрица реализаций оптимального вектора параметров $\hat{\mathbf{w}}$.

Определение 6. *Под точностью S модели \mathbf{f} с вектором параметров $\hat{\mathbf{w}}$ будем понимать значение функции ошибки (7) на контрольной выборке.*

Чем больше значение функции ошибки, тем меньше точность модели.

Определение 7. *Оптимальным вектором параметров модели $\mathbf{f}_{\mathcal{A}}$ назовем такой вектор $\hat{\mathbf{w}}_{\mathcal{A}}$, который является решением следующей задачи оптимизации:*

$$\hat{\mathbf{w}}_{\mathcal{A}} = \arg \min_{\mathbf{w}_{\mathcal{A}} \in \mathbb{R}^k} S(\mathbf{w}_{\mathcal{A}}|\mathcal{L}, \mathbf{f}_{\mathcal{A}}, \hat{\mathbf{A}}). \quad (8)$$

Задача выбора оптимальной модели состоит в том, чтобы найти модель $\mathbf{f} \in \mathfrak{F}$, для которой функция ошибки будет минимальной.

$$\hat{\mathbf{f}}_{\mathcal{A}} = \arg \min_{\mathbf{f}_{\mathcal{A}} \in \mathfrak{F}} S(\mathbf{f}_{\mathcal{A}}|C). \quad (9)$$

Устойчивость модели будет дополнительным критерием качества.

2. Описание алгоритма

Для получения оптимальной структуры модели в работе предлагается генетический алгоритм оптимизации структуры нейронной сети. Для сравнения также реализован базовый алгоритм NODE-OBDD.

2.1. Алгоритм прореживания структуры нейронной сети NODE-OBDD

Предлагаемый алгоритм определяет индекс нейрона, удаление которого приведет к минимизации приращения функции ошибки (7). Удаление нейрона эквивалентно обнулению соответствующего столбца матрицы \mathbf{W}_2 , т. е. удалению сразу группы параметров вектора \mathbf{w} . В этом разделе, для краткости изложения, будем обозначать матрицу \mathbf{W}_2 как \mathbf{W} . Предполагаем, что удаляемый нейрон наименьшим образом влияет на функцию ошибки. Для нахождения таких нейронов аппроксимируем функцию ошибки вблизи локального минимума матрицы \mathbf{W}_0 :

$$S(\mathbf{W}_0 + \Delta\mathbf{W}) = S(\mathbf{W}_0) + \mathbf{g}^T(\mathbf{W}_0)\Delta\mathbf{W} + \frac{1}{2}\Delta\mathbf{W}^T\mathbf{H}\Delta\mathbf{W} + O(\|\Delta\mathbf{W}\|^3),$$

где $\Delta\mathbf{W}$ — возмущение матрицы параметров в данной точке \mathbf{W}_0 ; $\mathbf{g}(\mathbf{W}_0)$ — вектор градиента, вычисленный в точке \mathbf{W}_0 ; $\mathbf{H} = \mathbf{H}(\mathbf{W}_0)$ — матрица вторых производных функции ошибки. Предполагается, что функция ошибки S находится в окрестности локального минимума. Тогда ее аппроксимация записывается в следующем виде:

$$\Delta S = \frac{1}{2}\Delta\mathbf{W}^T\mathbf{H}\Delta\mathbf{W}.$$

Пусть \mathbf{W}_k — набор параметров, соответствующий нейрону u_k , т. е. столбец матрицы \mathbf{W} , $\mathbf{W}_k = \mathbf{W}e_k$. Удаление этого нейрона (присвоение всем его параметрам нулевого значения) эквивалентно выполнению условия

$$\Delta\mathbf{W}e_k + \mathbf{W}_k = 0.$$

Получаем задачу условной минимизации

$$\Delta S = \frac{1}{2}\Delta\mathbf{W}^T\mathbf{H}\Delta\mathbf{W} \rightarrow \min, \Delta\mathbf{W}e_k + \mathbf{W}_k = 0.$$

Для решения этой задачи строим лагранжиан

$$L = \frac{1}{2}\Delta\mathbf{W}^T\mathbf{H}\Delta\mathbf{W} - \lambda(\Delta\mathbf{W}e_k + \mathbf{W}_k).$$

Продифференцировав L по $\Delta\mathbf{W}$, получаем значение выпуклости L_k для элемента \mathbf{W}_k :

$$L_k = \frac{\mathbf{w}_k^T\mathbf{w}_k}{2[\mathbf{H}^{-1}]_{k,k}},$$

где \mathbf{H}^{-1} — матрица, обратная гессиану \mathbf{H} ; $[\mathbf{H}^{-1}]_{k,k}$ — k -й диагональный элемент этой матрицы. Критерию оптимального прореживания отвечает группа параметров $\mathbf{W}_{\hat{k}}$ с минимальным значением выпуклости:

$$\hat{k} = \arg \min_{k \in \mathcal{A}} L_k.$$

Далее, используя функцию выпуклости L_k как величину, определяющую вероятность комбинирования и вариации структурных параметров, будет предложен недетерминированный вариант алгоритма *optimal brain damage*.

2.2. Генетический алгоритм оптимизации структуры нейронной сети

Функция ошибки (7) является многоэкстремальной функцией вектора параметров \mathbf{w} . Поэтому при ее минимизации одним из детерминированных алгоритмов [8, 9] определяется локальный минимум, который может не совпадать с глобальным минимумом. Для нахождения глобального минимума целесообразно воспользоваться недетерминированным генетическим алгоритмом.

Нейронная структура \mathcal{A} задается бинарным вектором $\mathbf{a} = [a_1, \dots, a_K]$:

$$\begin{cases} a_q = 1, & \text{если } k \in \mathcal{A}; \\ a_q = 0 & \text{иначе.} \end{cases}$$

Рассмотрим множество из M нейронных сетей с нейронными структурами \mathcal{A}_m , $m = [1, \dots, M]$, которому соответствует множество бинарных векторов $\mathcal{F}_0 = \{\mathbf{a}_m\}$, $m = [1, \dots, M]$. Назовем \mathcal{F}_0 популяцией. Для каждого вектора \mathbf{a}_m из множества \mathcal{F}_0 оценивается вектор параметров $\hat{\mathbf{w}}_{\mathcal{A}_m}$ соответствующей нейронной сети $\mathbf{f}_{\mathcal{A}_m}$ и вычисляется значение функции ошибки (7). Каждая из $\mathbf{f}_{\mathcal{A}_m}$ оптимизирована алгоритмом обратного распространения ошибки [14]. Опишем процедуру порождения новой популяции \mathcal{F}_1 из популяции \mathcal{F}_0 .

1. На множестве \mathcal{F}_0 задается случайная величина θ , которая принимает значение \mathbf{a}_m с вероятностью

$$p_m = \frac{\exp - \frac{Q_m}{Q_{\max}}}{\sum_{l=1}^K \exp - \frac{Q_l}{Q_{\max}}}, \quad (10)$$

где суммарная выпуклость $Q_l = \sum_{k=1}^K L_k^{\mathcal{A}_l}$ (здесь K — это число активных нейронов для нейронной сети

со структурой \mathcal{A}) всех активных нейронов для нейронной сети со структурой \mathcal{A}_p , а $Q_{\max} = \max_{l \in [1, \dots, M]} Q_l$.

Затем генерируется P реализаций случайной величины θ . Без ограничения общности будем считать, что P — четное число. Полученное множество векторов обозначим $F' = \{\mathbf{a}_1^T, \dots, \mathbf{a}_P^T\}$.

2. Множество F' случайным образом разбивается на пары $(\mathbf{a}_s^T, \mathbf{a}_t^T)$, где $s, t = 1, \dots, P, s \neq t$.

3. С каждой парой $(\mathbf{a}_s^T, \mathbf{a}_t^T)$ проводится операция скрещивания:

- генерируется случайное число $\zeta \in \{1, \dots, K-1\}$;
- векторы $(\mathbf{a}_s^T, \mathbf{a}_t^T)$ разделяются на две части и смешиваются следующим образом:

$$[a_s^1, \dots, a_s^\zeta, a_t^{\zeta+1}, \dots, a_t^K] \rightarrow \mathbf{a}_s,$$

$$[a_t^1, \dots, a_t^\zeta, a_s^{\zeta+1}, \dots, a_s^K] \rightarrow \mathbf{a}_t.$$

4. С каждым вектором из F' проводится операция модификации:

- генерируется случайное число $\eta < K$;
- инвертируется значение позиций η вектора \mathbf{a}_p и определяется вектор $\mathbf{a}_{p'}$.

Полученное множество векторов обозначается как $F_1 = \{\mathbf{a}_{p'}\}_{l=1}^P$ и является новой популяцией.

Таким образом, найден алгоритм для нахождения глобального минимума функции ошибки (7). Этот алгоритм использует выпуклость для определения вероятности использования структуры сети для комбинирования и вариации. Тем самым модели с меньшей выпуклостью будут с большей вероятностью использоваться для дальнейшего поиска оптимальной структуры сети.

4. Вычислительный эксперимент

Целью вычислительного эксперимента ставилось сравнение функции ошибки (7) генетического

алгоритма оптимизации структуры нейронной сети с алгоритмом NODE-OBD.

Использовались данные с акселерометра мобильного телефона. Показания акселерометра записывались при шести видах физической активности: ходьба, бег, сидение, стояние, подъем и спуск. Далее эти показания обрабатывались экспертным порождением признаков и метрическим алгоритмом. При использовании временных рядов порождались следующие признаки: проекции среднего ускорения на координатные оси, среднеквадратические отклонения от проекций среднего ускорения на каждую из трех координатных осей, время между пиками синусоидального сигнала в миллисекундах.

В метрической выборке признаки были получены как расстояния до центроидов классов. Расстояние между рядами задавалось с помощью метода выравнивания. С методом выделения центроидов и введения расстояния между временными рядами можно ознакомиться в работе [13].

В вычислительном эксперименте оптимизировалась двуслойная нейронная сеть с 40 нейронами в скрытом слое. Оптимизация проводилась по модифицированному OBD из раздела 2.1, а также по генетическому алгоритму из раздела 2.2. Для полученных на каждой итерации моделей (для лучшей модели в поколении популяции) подсчитана функция (7), по значению которой и сравнивалось качество моделей. На рис. 1 (см. третью сторону обложки) сплошной линией представлена зависимость функции ошибки (7) и штрих-пунктирной линией — зависимость устойчивости от числа итераций алгоритмов из разделов 2.1 и 2.2, рис. 1, а и б соответствуют функциям ошибки для генетического алгоритма и NODE-OBD для выборки, полученной с помощью ручного выделения признаков, рис. 1, в и г соответствуют функциям ошибки для генетического алгоритма и NODE-OBD для метрической выборки.

Как видно из рис. 1, генетический алгоритм позволяет получить большую точность за меньшее число итераций для обеих выборок. При этом в случае экспертного выделения признаков устойчивость уменьшается, тогда как в метрической выборке она увеличивается.

В случае базового алгоритма NODE-OBD функция ошибки (7) уменьшается незначительно или даже возрастает в случае экспертного порождения признаков. Устойчивость при этом в обоих случаях уменьшается.

На рис. 2 отображено значение параметра выпуклости L_j (*Saliency*) для всех активных нейронов. На каждой итерации число активных нейронов уменьшается, а абсолютные значения выпуклости у оставшихся нейронов становятся больше по абсолютному значению, также увеличивается раз-

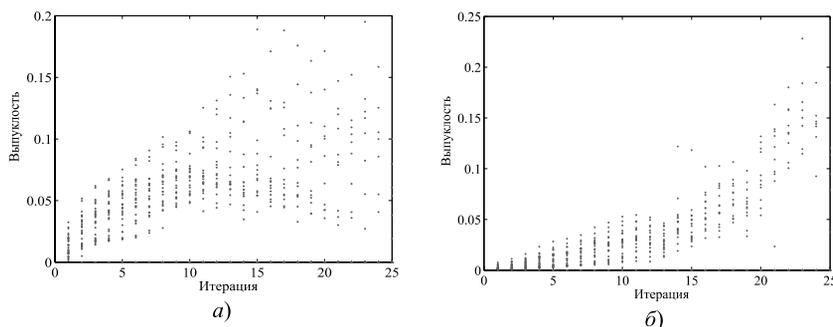


Рис. 2. Зависимость значений функции выпуклости нейронов сети от номера итерации:

а — экспертное порождение признаков; б — метрическая выборка

ность между значениями выпуклости активных нейронов.

На рис. 3 была визуализирована структура наиболее точной нейронной сети на каждой итерации. По горизонтали отложен номер итерации. Черная клетка означает, что нейрон активный, белая клетка — нейрон неактивный.

Представленные структуры получены при оптимизации сети, на вход которой подавались выборка с выделенными вручную 43 признаками [12] и выборка, полученная с помощью метода выравнивания расстояния до центроидов классов.

Для сравнения результатов, полученных с помощью исследуемых алгоритмов, с результатами других исследователей подсчитаны показатели Precision и Recall для наилучшей модели каждого из алгоритмов:

$$P = \frac{TP}{TP + FP}, R = \frac{TP}{TP + FN},$$

где TP , FP , FN — это число истинно положительных, ложноположительных, ложноотрицательных объектов данного класса соответственно. Эти показатели для каждого из классов занесены в таблицу. В ней отображены показатели для двух выборок для каждого из экспериментов.

Заключение

В работе предложены два алгоритма оптимизации структуры нейронной сети — генетический алгоритм прореживания и алгоритм NODE-OBD. Эти алгоритмы сравнивались по значениям функции ошибки и устойчивости. Вычислительный эксперимент показал, что NODE-OBD позволяет значительно уменьшить число активных нейронов, не увеличивая функцию ошибки модели, а генетический алгоритм позволяет получить модель с таким же числом нейронов, как и NODE-OBD, при этом уменьшая значения функции ошибки модели. Проведено сравнение работы алгоритма для двух видов обработки временных рядов. Наиболее точные результаты, сравнимые с результатами [11], получаются при использовании метрической выборки [14] и генетического алгоритма.

Список литературы

1. Ghosh J., Turner K. Structural adaptation and generalization in supervised feed-forward networks // J. of Artificial Neural Networks, 1994. Vol. 1, N. 4. P. 431–458.
2. Chung F., Lee T. A node pruning algorithm for backpropagation networks // Int. J. Neural Syst, 1992. Vol. 3, N. 3. P. 301–314.

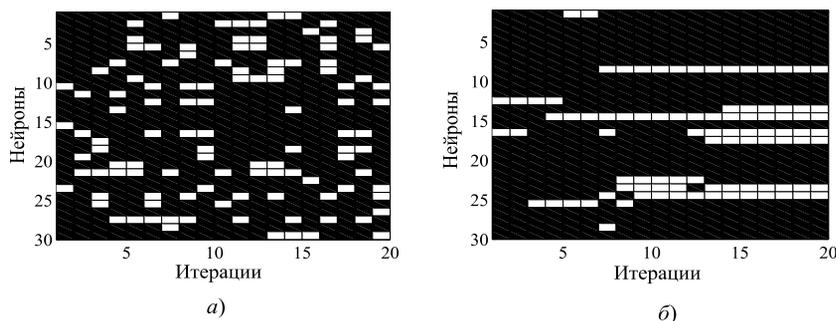


Рис. 3. Изменение структуры оптимальной сети из популяции: а — генетический алгоритм, экспертное порождение признаков; б — генетический алгоритм, метрическая выборка

Показатели самых точных моделей каждого из алгоритмов

Вид физической активности	OBD				Генетический алгоритм			
	Первая выборка		Метрическая выборка		Первая выборка		Метрическая выборка	
	P, %	R, %	P, %	R, %	P, %	R, %	P, %	R, %
Бег	83,2	84,9	100,0	98,0	88,3	86,9	100,0	98,0
Ходьба	95,1	96,3	90,9	89,2	98,0	96,7	89,1	90,7
Подъем	51,1	46,5	85,1	83,3	55,3	51,0	85,1	90,9
Спуск	46,7	46,5	82,6	90,4	40,2	50,3	93,4	89,6
Сидение	92,5	91,3	98,1	98,1	90,4	89,4	98,1	98,1
Стояние	93,1	92,1	100,0	98,0	93,1	92,1	100,0	98,0

3. Попова М. С., Стрижов В. В. Выбор оптимальной модели классификации физической активности по измерениям акселерометра // Информатика и ее применения, 2015. Т. 9. Вып. 1. С. 79–89.
4. Suisse M. V., Thimm G., Fiesler E., Thimm G., Fiesler E. Pruning of neural networks // Technical report, 2014.
5. MacLeod C., Maxwell G. M. Incremental evolution in ANNs: Neural nets which grow // Artif. Intell. Rev, 2001. Vol. 16, N. 3. P. 201–224.
6. Vukovic N., Miljkovic Z. A growing and pruning sequential learning algorithm of hyper basis function neural network for function approximation // Neural Networks, 2013. Vol. 46. P. 210–226.
7. Strijov V. V., Krymova E. V., Weber G. W. Evidence optimization for consequently generated models // Mathematical and Computer Modelling, 2013. Vol. 57, N. 1–2. P. 50–56.
8. Cun Y. L., Denker J. S., Solla S. A. Optimal brain damage // Advances in neural information processing systems, 1990. Vol. 2. P. 598–605.
9. Hassibi B., Stork D. G. Second order derivatives for network pruning: Optimal brain surgeon // Advances in Neural Information Processing Systems, 1993. Vol. 5, N. 1. P. 164–172.
10. Leung F. H., Lam H., Ling S., Tat P. K. Tuning of the structure and parameters of a neural network using an improved genetic algorithm // IEEE Transactions on Neural Networks, 2003. Vol. 14, N. 1. P. 79–88.
11. Oh I., Lee J., Moon B. R. Hybrid genetic algorithms for feature selection // IEEE Trans. Pattern Anal. Mach. Intell, 2004. Vol. 26, N. 11. P. 1424–1437.
12. Kwapisz J. R., Weiss G. M., Moore S. Activity recognition using cell phone accelerometers // SIGKDD Explorations, 2010. Vol. 12, N. 2. P. 74–82.
13. Rasekh A., Chen C., Lu Y. Human activity recognition using smartphone // Technical report, 2014.
14. Гончаров А. В., Стрижов В. В., Попова М. С. Метрическая классификация временных рядов с выравниванием относительно центроидов классов // Системы и средства информатики, 2015. Т. 1. № 4.

A. I. Zadayanchuk¹, Student, e-mail: alex.goncharov@phystech.edu,
M. S. Popova¹, Student, V. V. Strijov², Leading Scientist
¹ Moscow Institute of Physics and Technology
² Dorodnicyn Computing Center, Russian Academy of Sciences

Selection of Optimal Time Series Classification Model

This paper solves the problem of selecting optimal stable models for classification of physical activity. We select optimal models from the class of two-layer artificial neural networks. There are three different ways to change structure of neurons: network pruning, network growing, and their combination. We construct models by removing its neurons. Neural networks with insufficient or excess number of neurons have insufficient generalization ability and can make unstable predictions. Proposed genetic algorithm optimizes the neural network structure. The novelty of the work lies in the fact that the probability of removing neurons is determined by the variance of parameters. In the computing experiment, models are generated by optimization two quality criteria — accuracy and stability.

Keywords: classification, artificial neural networks, accuracy, genetic algorithm, pruning criteria

References

1. Ghosh J., Turner K. Structural adaptation and generalization in supervised feed-forward Networks, *Jl. of Artificial Neural Networks*, 1994, vol. 1, no. 4, pp. 431–458.
2. Chung F., Lee T. A node pruning algorithm for backpropagation networks, *Int. J. Neural Syst.*, 1992, vol. 3, no. 3, pp. 301–314.
3. Strijov V. V., Popova M. S. Vybor optimal'noj modeli klasifikacii fizicheskoy aktivnosti po izmerenijam akselerometra, *Informatika i ee primenenija*, 2015, vol. 9, no. 1, pp. 79–89.
4. Suisse M. V., Thimm G., Fiesler E., Thimm G., Fiesler E. Pruning of neural networks, *Technical report*, 2014.
5. MacLeod C., Maxwell G. M. Incremental evolution in ANNs: Neural nets which grow, *Artif. Intell. Rev.*, 2001, vol. 16, no. 3, pp. 201–224.
6. Vukovic N., Miljkovic Z. A growing and pruning sequential learning algorithm of hyper basis function neural network for function approximation, *Neural Networks*, 2013, vol. 46, pp. 210–226.
7. Strijov V. V., Krymova E. V., Weber G. W. Evidence optimization for consequently generated models, *Mathematical and Computer Modelling*, 2013, vol. 57, no. 1–2, pp. 50–56.
8. Cun Y. L., Denker J. S., Solla S. A. Optimal brain damage, *Advances in neural information processing systems*, 1990, vol. 2, pp. 598–605.
9. Hassibi B., Stork D. G. Second order derivatives for network pruning: Optimal brain surgeon, *Advances in Neural Information Processing Systems*, 1993, vol. 5, no. 1, pp. 164–172.
10. Leung F. H., Lam H., Ling S., Tam P. K. Tuning of the structure and parameters of a neural network using an improved genetic algorithm, *IEEE Transactions on Neural Networks*, 2003, vol. 14, no. 1, pp. 79–88.
11. Oh I., Lee J., Moon B. R. Hybrid genetic algorithms for feature selection, *IEEE Trans. Pattern Anal. Mach. Intell.*, 2004, vol. 26, no. 11, pp. 1424–1437.
12. Kwapisz J. R., Weiss G. M., Moore S. Activity recognition using cell phone accelerometers, *SIGKDD Explorations*, 2010, vol. 12, no. 2, pp. 74–82.
13. Rasekh A., Chen C., Lu Y. Human activity recognition using smartphone, *Technical report*, 2014.
14. Goncharov A. V., Strijov V. V., Popova M. S. Metricheskaja klassifikacija vremennyh rjadov s vyvavnivaniem otnositel'no centroidov klassov, *Sistemy i sredstva informatiki*, 2015, vol. 1, no. 4.

ИНФОРМАЦИЯ

В Волгограде с 1 по 3 июня 2016 года состоится выставка-форум "Мир ТелеКом" —

ХІІІ выставка информационных технологий.

Выставка зарекомендовала себя высокоэффективной площадкой для проведения деловых встреч и переговоров и демонстрации технических достижений.

Разделы выставки:

IT городской среды: Облачные решения. Телефония. Системы доставки корреспонденции. Электронное правительство. Системы электронных торгов. Геоинформационные системы. Управление ИТ. ИТ системы для библиотек. ИТ технологии коммунального хозяйства, обращения с отходами, охраны окружающей среды. Городское видеонаблюдение. "Зеленые" ИТ технологии.

Телекоммуникационные технологии: Проводные, беспроводные, мобильные, спутниковые технологии. Проектирование, строительство, эксплуатация сетей. Интернет технологии. Веб-дизайн. Телевидение. IP-технологии / VoIP. Data Center Технологии и решения. Телематика. М2М — межкомпьютерное взаимодействие. РС, ноутбуки, нетбуки, планшетные ПК. Hardware & Software, приложения. Электронная коммерция. Цифровой маркетинг. Биометрия.

Развитие ИТ: Start-up/Spin-off. Образование, здравоохранение, культура. Электронное обучение. Венчурный капитал. Инвестиции и финансирование. Web-Медиа. Интерактивное телевидение. Умный дом. Автоматизация. Автомобильные ИТ решения. Робототехника (игровые и сервис роботы). Прикладные и фундаментальные исследования.

ИТ для бизнеса: ИТ-услуги / Аутсорсинг. Программные продукты для предприятия. Системы планирования на предприятиях (ERP). Управление взаимоотношениями с клиентами (CRM). Управление бизнес-процессами (BPM, ESM). Управление человеческими ресурсами (HR). Системы автоматической идентификации пользователей. Логистика ИТ. Open Source системы. Виртуализация, облачные вычисления. Интеграция и корпоративные решения.

ИТ-безопасность: ИТ технологии общественной безопасности и внутренней безопасности. Mobile Security. Коммуникации, защита. Anti-Spam & Anti-Virus — решения. Безопасность облачных решений. Web-безопасность. Сетевая безопасность.

Подробная информация по выставке: <http://www.regionex.ru/exhibits/2016/telekom/>

По вопросу участия и посещения обращаться в Волгоградский Выставочный Центр "Регион".

С уважением,

Волгоградский Выставочный Центр "Регион"

Тел./факс: (8442) 26-51-86, 24-26-02, 23-28-99

E-mail: bez2@regionex.ru

IS & IT'16

Официальный сайт конгресса
<http://icai.tti.sfedu.ru/>

Международный конгресс по интеллектуальным системам и информационным технологиям

2—9 сентября 2016 г.

Россия, Черноморское побережье, Геленджик—Дивноморское

Основные даты:

Прием заявок	до 01.05.16 г.
Прием текстов докладов	до 01.06.16 г.
Финальная версия	до 15.07.16 г.
Регистрация участников	с 02.09.16 г.
Начало заседаний	с 03.09.16 г.

Организаторы конгресса:

- Министерство образования и науки РФ;
- Российская Академия наук;
- Российская Академия естественных наук;
- Южный федеральный университет;
- НКБ "Миус" ЮФУ;
- Администрация г. Таганрога;
- Российская ассоциация искусственного интеллекта (РАИИ);
- Университет Артуа (Франция);
- Университет Мюнстер (Германия);
- Университет Монс (Бельгия).

Тематика конгресса:

- Биоинформатика;
- Интеллектуальные САПР, CASE-, CALS-технологии;
- Искусственный интеллект и мягкие вычисления;
- Представление и извлечение знаний;
- Многоагентные системы и принятие решений;
- Перспективные информационные технологии;
- Проблемы образования;
- Синергетика и моделирование сложных систем;
- Эволюционное моделирование и генетические алгоритмы;
- Экспертные системы;
- Информационная безопасность;
- SAP-технологии;
- Инструментальные, математические и информационные средства экономики

Мероприятия, проводимые в рамках конгресса:

- Международная научно-техническая конференция <Интеллектуальные системы и информационные технологии IS&IT'16>
- Международная научно-техническая конференция <Интеллектуальные САПР CAD—2016>
- Молодежная научно-техническая конференция <Информационные системы и технологии—2016>
- Международный форум корпорации **EMC**
- Круглые столы по проблемам интеллектуальных систем и информационных технологий
- Ярмарка-выставка программных продуктов

Рабочие языки конгресса: русский и английский

Труды конгресса будут изданы к началу заседаний и включены в РИНЦ

Адрес оргкомитета:

IS & IT'16, проф. В. В. Курейчику, ЮФУ. 44, пер. Некрасовский,
г. Таганрог, 347928, Российская Федерация.

E-mail: vkur@tsure.ru, nev@tsure.ru, kur@tsure.ru, ivr@tsure.ru

Телефоны: 8634-37-16-51, 8634-39-32-60, 8634-38-34-51 **Факс:** 8634-37-16-5



Институт прикладной математики им. М. В. Келдыша РАН проводит
с 19 по 24 сентября 2016 г.

XVIII Всероссийскую научную конференцию "Научный сервис в сети Интернет".

Основная цель данной конференции — предоставить возможность для обсуждения, апробации и обмена мнениями о наиболее значимых результатах, полученных ведущими российскими учеными за последнее время в данной области деятельности. Конференция посвящена основным направлениям и тенденциям использования интернет-технологий в современных научных исследованиях.

Основные тематические направления конференции:

- Научные исследования и интернет, интернет-представительство научных организаций и проектов
- Интернет-проекты в области параллельных вычислений и распределенной обработки данных, вычислительные сервисы
- Удаленный доступ к научным исследовательским комплексам
- Математическое моделирование и интернет
- Технологии и системы распределенного хранения данных
- Модели и методы построения поисковых систем и систем навигации в интернете
- Технологии и опыт построения информационных систем и баз данных, документации и результатов эксперимента на основе интернет-технологий.
- Цифровые библиотеки и библиографические базы, семантический веб, наукометрия в интернете
- Онлайн-научная публикация, открытая наука, живая публикация, онлайн-рецензирование, мультимедийные иллюстрации
- Популярный научный интернет, онлайн-энциклопедии, история науки в интернете
- Интернет-активность ученого, персональная страница, профили ученого в библиографических базах, аттестация в интернете

Конференция проводится в пансионате "Моряк", расположенном в 20 км от Новороссийска

Ключевые даты конференции:

1 июня 2016 г. — завершение регистрации и представления докладов

1 июля — уведомление о включении доклада в программу конференции

15 июля — представление окончательного варианта статьи

30 августа — оплата оргвзноса и проживания

15 сентября — завершение регистрации участников конференции

19 сентября — день заезда

20–23 сентября — **рабочие дни конференции**

24 сентября — день отъезда

Сайт конференции <http://agora.guru.ru/abrau2016>

Адрес редакции:

107076, Москва, Стромынский пер., 4

Телефон редакции журнала **(499) 269-5510**

E-mail: it@novtex.ru

Технический редактор *Е. В. Конова*.

Корректор *Т. В. Пчелкина*.

Сдано в набор 09.02.2016. Подписано в печать 24.03.2016. Формат 60×88 1/8. Бумага офсетная.

Усл. печ. л. 8,86. Заказ IT416. 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.