DOI 10.17587/issn.1684-6400

ISSN 1684-6400

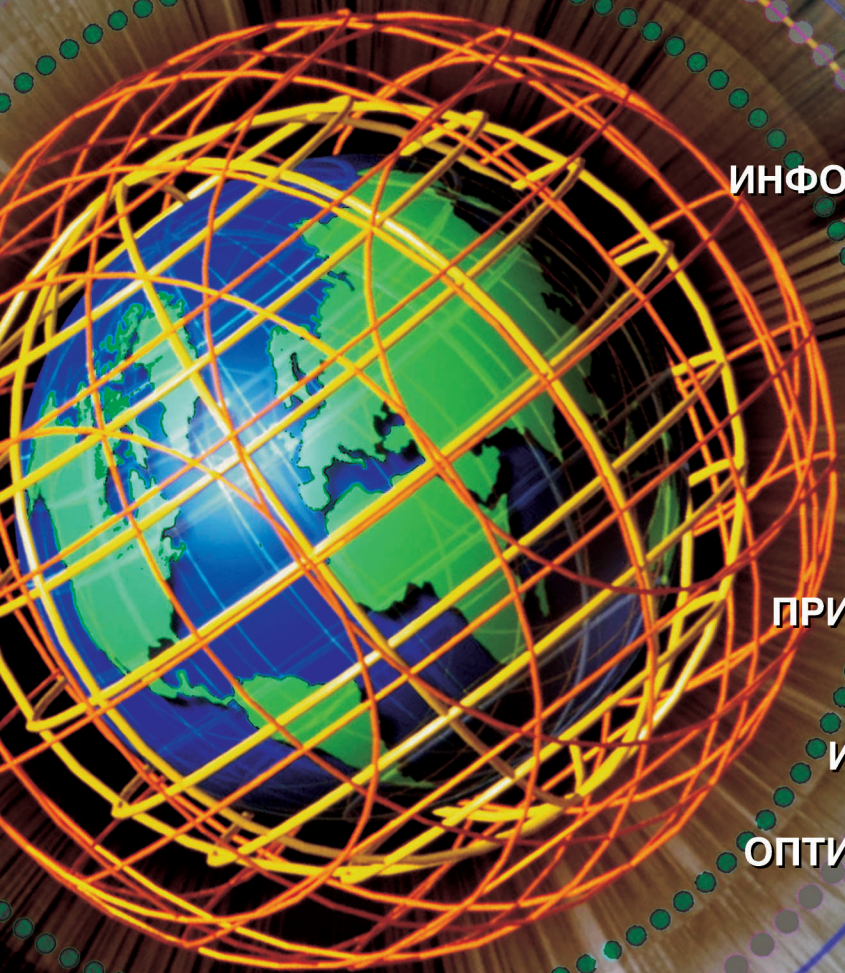
ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Том 25

2019

№ 9

ТЕОРЕТИЧЕСКИЙ И ПРИКЛАДНОЙ НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ



САПР

КОМПЬЮТЕРНАЯ ГРАФИКА

МЕТОДЫ ПРОГРАММИРОВАНИЯ

ОПЕРАЦИОННЫЕ СИСТЕМЫ И СРЕДЫ

ТЕЛЕКОММУНИКАЦИИ И ВЫЧИСЛИТЕЛЬНЫЕ СЕТИ

ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ

НЕЙРОСЕТИ И НЕЙРОКОМПЬЮТЕРЫ

СТРУКТУРНЫЙ СИНТЕЗ

ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ

ПРИКЛАДНЫЕ ИНФОРМАЦИОННЫЕ СИСТЕМЫ

ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ

ОПТИМИЗАЦИЯ И МОДЕЛИРОВАНИЕ

ИТ В ОБРАЗОВАНИИ

ГИС

Рисунки к статье А. Ю. Романова, Е. А. Ведмидь, Э. А. Монаховой

«ПРОЕКТИРОВАНИЕ СЕТЕЙ НА КРИСТАЛЛЕ С ТОПОЛОГИЕЙ КОЛЬЦЕВОЙ ЦИРКУЛЯНТ С ТРЕМЯ ОБРАЗУЮЩИМИ: РАЗРАБОТКА АЛГОРИТМОВ МАРШРУТИЗАЦИИ»

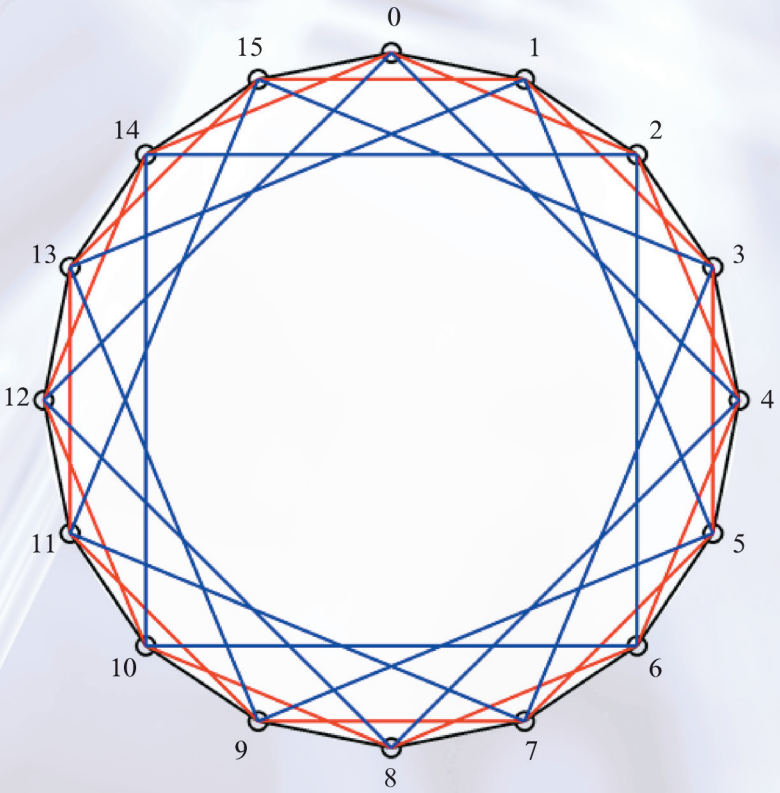


Рис. 1. Кольцевой циркулянт $G(16; 1, 2, 4)$

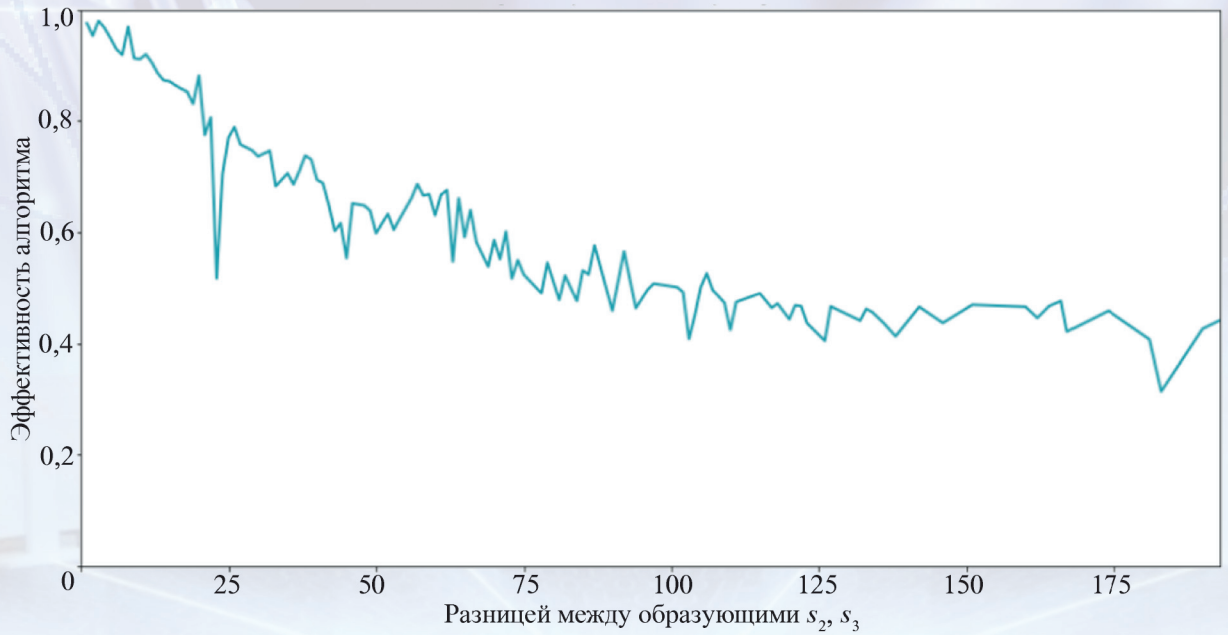


Рис. 2. Зависимость между эффективностью алгоритма и разницей между образующими s_2, s_3

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Том 25
2019
№ 9

ТЕОРЕТИЧЕСКИЙ И ПРИКЛАДНОЙ НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ

Издается с ноября 1995 г. DOI 10.17587/issn.1684-6400

УЧРЕДИТЕЛЬ
Издательство "Новые технологии"

СОДЕРЖАНИЕ

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

INFORMATION TECHNOLOGIES
INFORMACIONNYE TEHNOLOGII

Vol. 25
2019
No. 9

THEORETICAL AND APPLIED SCIENTIFIC AND TECHNICAL JOURNAL

Published since November 1995

ISSN 1684-6400

CONTENTS

Complete Internet version of the journal at site: **http://novtex.ru/IT.**
According to the decision of the Higher Certifying Commission of the Ministry of Education of Russian Federation, the journal is inscribed in "The List of the Leading Scientific Journals and Editions wherein Main Scientific Results of Theses for Doctor's or Candidate's Degrees Should Be Published"

УДК 681.36 DOI: 10.17587/it.25.515-521

С. А. Инютин, д-р техн. наук, проф., e-mail: inyutin_int@mail.ru,
Московский авиационный институт (национальный исследовательский университет) (МАИ)

Дробно-рациональные конструкции в компьютерной модулярной арифметике

Анализируется расширение числовых модулярных представлений на множество рациональных дробей с фиксированным знаменателем. Разработаны методы выполнения основных компьютерных операций для таких представлений в разрядной сетке вычислительной реконфигурируемой системы. Разработаны и обоснованы алгоритмы их выполнения, получены оценки временной и табличной сложностей. Доказана целесообразность применения квадратичной модулярной системы для уменьшения сложности отдельных итерационных процедур, лежащих в основе операций над введенными модулярными представлениями, что обеспечивает квадратичную сложность. Введенные модулярные представления и соответствующие компьютерные форматы применимы в модулярных реконфигурируемых вычислительных системах SIMD-архитектуры.

Ключевые слова: *многопроцессорные реконфигурируемые вычислительные системы, SIMD-архитектура, вычислительный модулярный процесс, квадратичная сложность вычислений, параллельные компьютерные форматы для рациональных дробей*

Введение

Известно биективное отображение целых числовых величин из конечного числового подмножества на множество векторов модулярного представления, являющихся теоретической базой параллельных компьютерных форматов данных, обрабатываемых в модулярных процессорах [1]:

$$A \in \left[0, \ldots \prod_{i=1}^{n} p_i\right) \leftrightarrow (\alpha'_1, \alpha'_2, \ldots, \alpha'_n), \qquad (1)$$

где $\alpha'_i = |A|_{p_i}$ — наименьший неотрицательный вычет по модулю p_i целой числовой величины $A \in \left[0, \ldots \prod_{i=1}^{n} p_i\right)$, при условии $(p_i, p_j) = 1$, $\forall i, j \in \{1, \ldots, n\}$.

Целые простые (попарно взаимно простые) модули p_i называются основаниями модулярной (параллельной) системы счисления. Произведение оснований называется модулярным числовым диапазоном $P = \prod_{i=1}^{n} p_i$, порождающим полную систему (множество) наименьших неотрицательных вычетов. Этому множеству принадлежат числовые величины, обрабатываемые в модулярном процессоре. Модулярное представление для целых чисел является вектором. Компоненты модулярных векторов могут быть классическими вычетами или нормированными вычетами.

Для векторного модулярного представления числовых величин (1) получим нормированное модулярное представление:

$$A \leftrightarrow (\alpha'_1, \alpha'_2, \ldots, \alpha'_n) \leftrightarrow (\alpha_1, \alpha_2, \ldots, \alpha_n), \qquad (2)$$

где нормированные компоненты модулярного вектора $\alpha_i = \left|\alpha'_i \left|\frac{P}{p_i}\right|_{p_i}^{-1}\right|_{p_i} < p_i$.

Прямыми скобками с модулем в правой нижней части обозначена постфиксная форма бинарной операции $f(x, p) = |x|_p : \mathbf{N} \times \mathbf{N} \to |\circ|_p$, где $|\circ|_p$ — полная система наименьших неотрицательных вычетов по модулю p; $\mathbf{N} \times \mathbf{N}$ — декартово произведение множеств натуральных чисел.

Для числовых величин, имеющих модулярное векторное представление (2), выполняется аддитивное разложение

$$A = \sum_{i=1}^{n} \frac{\alpha_i}{p_i} P - \left[\sum_{i=1}^{n} \frac{\alpha_i}{p_i}\right] P = \sum_{i=1}^{n} \alpha_i P_i - \left[\sum_{i=1}^{n} \frac{\alpha_i}{p_i}\right] P, \qquad (3)$$

где $P_i = P/p_i$.

Квадратными скобками обозначена дискретная функция — наибольшая целая часть, не превышающая числовую величину в скобках.

Позиционной характеристикой — нормированным рангом числовой величины — называют введенную В. М. Амербаевым [1] функцию от нормированных компонент вектора

$$R_A = \varphi(\alpha_1,\ldots\alpha_n) = \left[\sum_{i=1}^{n}\frac{\alpha_i}{p_i}\right].$$

Ранг необходим для выполнения ряда операций над модулярными компьютерными форматами данных. Для точного вычисления нормированного ранга модулярного вектора необходим вычислительный диапазон P, который при эмуляции модулярной арифметики на серийных компьютерах для больших значений n и p_i значительно превосходит типовой машинный, что приводит к большой сложности вычисления функции. Разработан ряд методов вычисления ранга, один из которых с линейной сложностью принадлежит автору [2].

Введем ряд понятий, опираясь на терминологию работы [4]. Одинарная модулярная система построена с использованием простых оснований, и числовую величину в ней как элемент полной системы наименьших неотрицательных вычетов по модулю запишем в классическом виде $A(\mathrm{mod}P)$. Квадратичная модулярная система построена с использованием квадратов простых оснований, и числовую величину в ней запишем в виде $A(\mathrm{mod}P^2)$. Смешанная (одинарно-квадратичная) модулярная система построена с использованием простых оснований и квадратов оснований.

Модулярная разрядная сетка вычислительной системы — аппаратно-программное устройство (модель), состоящая из n элементарных ячеек, предназначенных для размещения и обработки вычетов по модулям, являющимся компонентами вектора модулярного представления числовой величины.

Модулярное представление для правильных рациональных дробей

Для области специализированной вычислительной техники, которая опирается на модулярное представление целых числовых величин, является актуальным обобщение этого представления на множество правильных рациональных дробей, что порождает модулярный дробно-рациональный формат данных. Введем модулярное представление для правильной рациональной дроби со знаменателем, равным модулярному диапазону. Получим аддитивное разложение для правильной модулярной дроби с нормированными компонентами модулярного представления:

$$\frac{A}{P} = \sum_{i=1}^{n}\frac{\alpha_i}{p_i} - R_A. \qquad (4)$$

Для отдельных дробей из правой части равенства (4) получим:

$$\forall j \in \{1\ldots n\} : \frac{A}{p_j} = \frac{\alpha_j P_j}{p_j} +$$

$$+\frac{1}{p_j}\sum_{i=1,\ i\neq j}^{n}\frac{\alpha_i P_j p_j}{p_i} - \frac{1}{p_j}R_A P_j p_j.$$

Вычет от числителя по выбранному модулю даст соответствующую компоненту модулярного представления в виде правильной дроби.

Разрядная сетка модулярных вычислительных систем предназначена для отображения вычетов по модулям. Совокупность таких вычетов по соответствующим модулям в зависимости от поставленной задачи можно считать разрядами модулярного представления целого числа или разрядами правильной рациональной дроби. Это различие влияет на алгоритмы выполнения операций.

Оценим приближение произвольной правильной рациональной дроби $\frac{V}{U}$ модулярной дробью $\frac{A}{P}$ следующим соотношением: $\left|\frac{V}{U}-\frac{A}{P}\right| \to \min$ или $\left|\left[\frac{VP}{U}\right]-A\right| \to \min$, где прямыми скобками обозначена абсолютная величина числа. Выполняются утверждения:

— в общем случае погрешность приближения не превышает значения $\frac{1}{P}$;

— для минимизации погрешности приближения необходимо выбирать максимальное из возможных значение P;

— нулевая погрешность приближения будет получена, если P есть мультипликативное каноническое разложение числа U.

Операции с модулярными дробями

Пусть даны модулярные представления правильных дробей:

$$\frac{A}{P} \leftrightarrow (\alpha_1,\alpha_2,\ldots,\alpha_n);$$

$$\frac{B}{P} \leftrightarrow (\beta_1,\beta_2,\ldots,\beta_n).$$

Для них выполняются аддитивные разложения:

$$\frac{A}{P}=\sum_{i=1}^{n}\frac{\alpha_i}{p_i}-R_A;\ \ \frac{B}{P}=\sum_{i=1}^{n}\frac{\beta_i}{p_i}-R_B.$$

Рассмотрим операцию сложения двух правильных модулярных дробей.

В кодонезависимой записи (независимой от выбора системы счисления для представления числовых величин) получим:

$$\frac{M}{P}=\frac{A}{P}+\frac{B}{P}=\frac{A+B}{P}=\frac{|A+B|_P}{P}+0\mid 1,$$

альтернативные варианты разделены вертикальной чертой, причем единица свидетельствует о переполнении вычислительного диапазона.

В кодозависимой записи для одинарной модулярной системы получим

$$\frac{M}{P}=\frac{|A+B|_P}{P}\leftrightarrow$$
$$\leftrightarrow(|\alpha_1+\beta_1|_{p_1},\ldots|\alpha_i+\beta_i|_{p_i},\ldots)=(\mu_1,\ldots\mu_i\ldots).$$

При суммировании двух операндов возможно переполнение вычислительного диапазона, которое устанавливается анализом нормированных рангов операндов и результата на основании следующей теоремы.

Теорема 1. Признаком переполнения одинарного вычислительного диапазона является выполнение выражения:

$$1=R_M+\sum_{i=1}^{n}\left[\frac{\alpha_i+\beta_i}{p_i}\right]-R_A-R_B. \qquad (5)$$

Доказательство.

При суммировании аддитивных разложений модулярных дробей получим

$$\frac{M}{P}=\frac{A}{P}+\frac{B}{P}=\sum_{i=1}^{n}\frac{\alpha_i}{p_i}-R_A+\sum_{i=1}^{n}\frac{\beta_i}{p_i}-R_B=$$
$$=\sum_{i=1}^{n}\frac{\alpha_i+\beta_i}{p_i}-R_A-R_B=$$
$$=\sum_{i=1}^{n}\frac{|\alpha_i+\beta_i|_{p_i}}{p_i}+\sum_{i=1}^{n}\left[\frac{\alpha_i+\beta_i}{p_i}\right]-R_A-R_B.$$

Вместе с тем выполняется соотношение

$$\frac{M}{P}=\frac{|A+B|_P}{P}=\sum_{i=1}^{n}\frac{|\alpha_i+\beta_i|_{p_i}}{p_i}-\left[\sum_{i=1}^{n}\frac{|\alpha_i+\beta_i|_{p_i}}{p_i}\right]=$$
$$=\sum_{i=1}^{n}\frac{|\alpha_i+\beta_i|_{p_i}}{p_i}-R_M.$$

Следовательно, признаком переполнения модулярного вычислительного диапазона является выполнение следующего равенства:

$$R_M+\sum_{i=1}^{n}\left[\frac{\alpha_i+\beta_i}{p_i}\right]-R_A-R_B=1.$$

Следствие. Для выполнения операции сложения с определением переполнения (5) необходимо вычисление трех позиционных характеристик — рангов числовых величин A, B, M, что при определенных условиях возможно с линейной сложностью [2].

Рассмотрим операции умножения двух дробей и дроби на целое число и выделим в них общие этапы.

Результатом операции умножения правильной рациональной дроби на целое число является неправильная дробь — целое число и правильная рациональная дробь. В кодонезависимой записи получим: $\frac{A}{P}\cdot D=\left[\frac{A\cdot D}{P}\right]+\frac{|A\cdot D|_P}{P}$.

Рассмотрим операцию умножения двух правильных модулярных дробей. Это произведение вычисляется с погрешностью из-за ограниченности разрядной сетки вычислительной системы, что требует выполнения операции деления на P в соответствии с кодонезависимым соотношением:

$$\frac{C}{P}=\frac{A}{P}\cdot\frac{B}{P}=\frac{\left[\frac{A\cdot B}{P}\right]+\frac{|A\cdot B|_P}{P}}{P}\cong\frac{\left[\frac{A\cdot B}{P}\right]}{P}.$$

Рассмотрим аналогичные соотношения в модулярной системе. Для операции умножения правильной рациональной дроби на целое число результатом является неправильная дробь в модулярном представлении:

$$\frac{|A|_p}{P}\cdot|D|_P=\frac{|A\cdot D|_{P^2}}{P}=$$
$$=\left[\frac{A\cdot D}{P}\right](\mathrm{mod}\ P\ )+\frac{A\cdot D(\mathrm{mod}\ P)}{P}. \qquad (6)$$

Для операции умножения двух правильных рациональных дробей получим соотношения, зависимые от модулярной системы счисления:

$$\frac{C(\mathrm{mod}\ P)}{P}=\frac{A(\mathrm{mod}\ P)}{P}\cdot\frac{B(\mathrm{mod}\ P)}{P}=$$
$$=\frac{A\cdot B(\mathrm{mod}\ P^2)}{P^2}= \qquad (7)$$
$$=\frac{\left[\frac{|A\cdot B|_{P^2}}{P}\right]+\frac{|A\cdot B|_P}{P}}{P}\approx\frac{\left[\frac{A\cdot B(\mathrm{mod}\ P^2)}{P}\right](\mathrm{mod}\ P)}{P}.$$

В соотношениях (6), (7) присутствует операция вычисления целой части от деления на одинарный диапазон произведения модулярных величин. Для ее выполнения можно использовать квадратичную модулярную систему. Рассмотрим алгоритм выполнения этой операции и проанализируем соотношения, лежащие в его основе.

Теорема 2. Произведение двух целых модулярных величин, заданных в одинарной модулярной системе, в квадратичной модулярной системе имеет вид

$$C = A \cdot B (\mathrm{mod}\, P^2) = \left(|A \cdot B|_P + \left[\frac{A \cdot B}{P} \right] P \right) (\mathrm{mod}\, P^2);$$

$$C = A (\mathrm{mod}\, P) \cdot B (\mathrm{mod}\, P) =$$

$$= A \cdot B (\mathrm{mod}\, P^2) \leftrightarrow \left(\tilde{s}_1 (\mathrm{mod}\, p_1^2) \ldots \tilde{s}_i (\mathrm{mod}\, p_i^2) \ldots \right) =$$

$$= (s_i + k_i p_i (\mathrm{mod}\, p_i^2), \ldots s_i + k_i p_i (\mathrm{mod}\, p_i^2), \ldots),$$

где для числовых коэффициентов выполняются неравенства $s_i, k_i < p_i$; $\tilde{s}_i < p_i^2$.

Доказательство.

Пусть две величины представлены ненормированными вычетами в одинарной модулярной системе. Запишем их аддитивные разложения:

$$A = \sum_{i=1}^{n} \frac{\alpha'_i m_i P}{p_i} - \left[\sum_{i=1}^{n} \frac{\alpha'_i m_i}{p_i} \right] P =$$

$$= \frac{\alpha'_j m_j P}{p_j} + \sum_{i \neq j}^{n} \frac{\alpha_i m_i P}{p_i} - \left[\sum_{i=1}^{n} \frac{\alpha'_i m_i}{p_i} \right] P;$$

$$B = \sum_{i=1}^{n} \frac{\beta'_i m_i P}{p_i} - \left[\sum_{i=1}^{n} \frac{\beta'_i m_i}{p_i} \right] P =$$

$$= \frac{\beta'_j m_j P}{p_j} + \sum_{i \neq j}^{n} \frac{\beta'_i m_i P}{p_i} - \left[\sum_{i=1}^{n} \frac{\beta'_i m_i}{p_i} \right] P.$$

Результат произведения $A \cdot B (\mathrm{mod} P)$ в квадратичной модулярной системе имеет вид:

$$|A \cdot B|_{P^2} \leftrightarrow (\alpha'_1 \beta'_1 m_1 m_1 (\mathrm{mod}\, p_1^2), \ldots$$

$$\ldots \alpha'_i \beta'_i m_i m_i (\mathrm{mod}\, p_i^2) \ldots) =$$

$$= ((\tilde{\alpha}_1 + k_1 p_1) \cdot (\tilde{\beta}_1 + d_1 p_1) (\mathrm{mod}\, p_1^2), \ldots$$

$$\ldots (\tilde{\alpha}_i + k_i p_i) \cdot (\tilde{\beta}_i + d_i p_i) (\mathrm{mod}\, p_i^2) \ldots) =$$

$$= (x_1 + y_1 p_1 (\mathrm{mod}\, p_1^2), \ldots x_i + y_i p_i (\mathrm{mod}\, p_i^2) \ldots).$$

Для векторных компонент, возникающих на промежуточных этапах, выполняются соотношения $\tilde{\alpha}_i, \tilde{\beta}_i, x_1, k_i, d_i, y_i < p_i$.

Найдем произведение аддитивных разложений числовых величин:

$$A \cdot B = \frac{\alpha'_j \beta'_j m_j^2 P_j^2}{p_j^2} + \left(\sum_{i \neq j}^{n} \frac{\alpha'_i m_i P_j}{p_i} \cdot \sum_{i \neq j}^{n} \frac{\beta'_i m_i P_j}{p_i} \ldots \right) p_j +$$

$$+ \left[\sum_{i=1}^{n} \frac{\beta'_i m_i}{p_i} \right] \cdot \left[\sum_{i=1}^{n} \frac{\alpha'_i m_i}{p_i} \right] P^2.$$

Вычислим ненормированные вычеты произведения в квадратичной модулярной системе на примере вычисления одной компоненты вектора представления:

$$|A \cdot B|_{p^2} = \left| \alpha'_j \beta'_j \right|_{p^2} + k_j p_j (\mathrm{mod}\, p_j^2) =$$

$$= \left| \alpha'_j \beta'_j \right|_{p_j} + \left[\frac{\alpha'_j \beta'_j}{p_j} \right] p_j + k_j p_j (\mathrm{mod}\, p_j^2) =$$

$$= \left| \alpha'_j \beta'_j \right|_{p_j} + \left| \left[\frac{\alpha'_j \beta'_j}{p_j} \right] + k_j \right|_{p_j} p_j (\mathrm{mod}\, p_j^2),$$

где для числового коэффициента выполняется соотношение $k_j < p_j$.

Теорема доказана.

Рассмотрим операцию отображения в квадратичный модулярный диапазон числовой величины, заданной в одинарном модулярном диапазоне:

$$|A|_{P^2} = \left| \sum_{i=1}^{n} \frac{\alpha_i}{p_i} P - R_A P \right|_{P^2}.$$

Вычислим компоненты вектора представления в квадратичном диапазоне на примере вычисления одной компоненты:

$$|A|_{p_j^2} = \left| \sum_{i=1}^{n} \frac{\alpha_i}{p_i} P - R_A P \right|_{p_j^2} =$$

$$= \left| \alpha_j P_j + \sum_{i=1, i \neq j}^{n-1} \alpha_i P_i - R_A P \right|_{p_j^2} = \left| \alpha_j P_j \right|_{p_j^2} + x_j p_j =$$

$$= \left| \alpha_j P_j \right|_{p_j} + \left| \left[\frac{\left| \alpha_j P_j \right|_{p_j^2}}{p_j} \right] + x_j \right|_{p_j} p_j (\mathrm{mod}\, p_j^2),$$

где выполняется условие $x_j < p_j$ для числа, полученного при промежуточных вычислениях.

Рассмотрим алгоритм деления числовой величины на числовой диапазон одинарной модулярной системы в модулярной системе квадратичного диапазона:

$$C = A \cdot B (\mathrm{mod}\, P^2) = |A \cdot B|_P + \left[\frac{A \cdot B}{P} \right] P (\mathrm{mod}\, P^2).$$

На входе алгоритма дано произведение модулярных представлений числовых величин в квадратичной модулярной системе:

$$C = A(\mathrm{mod}\, P)\cdot B(\mathrm{mod}\, P) = A\cdot B(\mathrm{mod}\, P^2) \leftrightarrow$$
$$\leftrightarrow (\tilde{s}_1(\mathrm{mod}\, p_1^2)\ldots\tilde{s}_i(\mathrm{mod}\, p_i^2)\ldots) =$$
$$= (s_1 + d_1 p_1(\mathrm{mod}\, p_1^2),\ldots s_i + d_i p_i(\mathrm{mod}\, p_i^2),\ldots)$$
$$\forall i \in \{1,\ldots n\}; s_i, d_i < p, \tilde{s}_i < p_i^2.$$

Обозначим

$$C = C^1 \leftrightarrow$$
$$\leftrightarrow (s_1^1 + d_1^1 p_1(\mathrm{mod}\, p_1^2),\ldots s_i^1 + d_i^1 p_i(\mathrm{mod}\, p_i^2),\ldots),$$

где верхний индекс в векторных компонентах означает номер итерации в алгоритме.

Алгоритм деления.

1. Вычитание элемента компоненты s_1^1 и деление на основание p_1 в каждой векторной компоненте модулярного представления числовой величины:

$$\frac{C^1 - s_1^1}{p_1} \leftrightarrow$$
$$\leftrightarrow (d_1^1(\mathrm{mod}\, p_1),\ldots(s_i^1 + d_i^1 p_i - s_1^1)|p_1|_{p_i^2}^{-1}(\mathrm{mod}\, p_i^2),\ldots) =$$
$$= (d_1^1(\mathrm{mod}\, p_1),\ldots(s_i^2 + d_i^2 p_i)(\mathrm{mod}\, p_i^2),\ldots) =$$
$$= (d_1^2(\mathrm{mod}\, p_1),\ldots(s_i^2 + d_i^2 p_i)(\mathrm{mod}\, p_i^2),\ldots) = C^2.$$

2. Вычитание элемента s_2^2 следующей компоненты и деление на основание p_2 в каждой компоненте представления числовой величины:

$$\frac{C^2 - s_2^2}{p_2} \leftrightarrow ((d_1^2 - s_2^2)|p_2|_{p_1}^{-1}(\mathrm{mod}\, p_1),\ldots$$
$$\ldots(s_i^2 + d_i^2 p_i - s_2^2)|p_2|_{p_i^2}^{-1}(\mathrm{mod}\, p_i^2),\ldots) =$$
$$= (s_1^3(\mathrm{mod}\, p_1),\ldots s_i^3 + d_i^3 p_i(\mathrm{mod}\, p_i^2),\ldots) = C^3.$$

3. Операции из пунктов 1—2 выполняются для всех остальных оснований модулярной системы.

Выражения на различных этапах алгоритма представлены в смешанных одинарно-квадратичных модулярных системах.

На выходе алгоритма после выполнения n этапов получен результат в одинарной модулярной системе $\dfrac{C - |A\cdot B|_P}{P} \equiv \left[\dfrac{A\cdot B}{P}\right](\mathrm{mod}\, P)$.

Одинарный диапазон в квадратичной модулярной системе имеет представление:

$$P(\mathrm{mod}\, P^2) \leftrightarrow$$
$$\leftrightarrow (|P_1|_{p_1} p_1(\mathrm{mod}\, p_1^2);\ldots.|P_i|_{p_i} p_i(\mathrm{mod}\, p_i^2),\ldots).$$

Это позволяет получить представление модулярной величины в квадратичном диапазоне:

$$|A\cdot B|_P = C - \left[\frac{A\cdot B}{P}\right] P(\mathrm{mod}\, P^2).$$

Запишем преобразование, выполняемое на каждом шаге приведенного алгоритма деления, на алгоритмическом псевдоязыке в нотации работы [3]. Такой подход позволяет получить компактную запись сложных алгоритмов, выделив повторяющиеся последовательности преобразований в отдельные процедуры, вызываемые в последующих этапах и алгоритмах.

Algorithm SD

На входе задан одномерный массив, элементами которого являются вычеты по модулям — квадратам оснований $C(1:n) \leftrightarrow (\ldots\tilde{s}_i\ldots)$.

1. $i \leftarrow 1$
2. $\tilde{s}_i \leftarrow \left(\tilde{s}_i - |\tilde{s}_j|_{p_j}\right)\cdot |p_j|_{p_i}^{-1}$
3. $s_i \leftarrow |\tilde{s}_i|_{p_i}$
4. $d_i \leftarrow \left[\dfrac{\tilde{s}_i - s_i}{p_i}\right]$
5. $\tilde{C}(i) \leftarrow (s_i, d_i)$
6. $i \leftarrow i + 1$
7. *if* $i \leqslant n$ *then goto* 2 *else exit*
8. *exit* : $\tilde{C}(1:n)$
9. *comment*: $\tilde{C}(1:n) \leftrightarrow (\ldots(s_i, d_i)\ldots)$

Для оценки сложности запишем алгоритм деления на алгоритмическом псевдоязыке.

Algorithm Division

$$C^1 \leftrightarrow (\ldots(s_i^1, d_i^1)\ldots) \leftarrow C \leftrightarrow (\ldots(s_i, d_i)\ldots)$$

1. $j \leftarrow 1$
2. $C^{j+1} \leftarrow call\ SD(C^j)$
3. *comment* $C^{j+1} \leftarrow \left[\dfrac{C^j - s^j}{p^j}\right]$
4. $C^{j+1} \leftrightarrow (\ldots(s_i^{j+1}, d_i^{j+1})\ldots)$
5. $j \leftarrow j + 1$
6. *if* $j \leqslant n$ *then goto* 2 *else exit*
7. *exit*: C^{n+1}
8. *comment*: $C^{n+1} = \left[\dfrac{A\cdot B(\mathrm{mod}\, P^2)}{P}\right](\mathrm{mod}\, P);$
$|A\cdot B|_P(\mathrm{mod}\, P^2) = C^1 - C^{j+1}P(\mathrm{mod}\, P^2)$

Для n-разрядного модулярного процессора (*n-modular processor*) найдем оценки сложности. Временная сложность алгоритма $time = O(n^2)$. Таблицы констант $\left|p_i\right|_j^{-1}$, используемых на n этапах алгоритма, имеют квадратичную размерность (оба индекса принадлежат одному множеству $\forall i, j \in \{1,\ldots n\}$). Назовем оценку их объема табличной сложностью алгоритма $const = O(n^2)$ [4]. Таблицы с константами целесообразно хранить в дополнительной машинной памяти и загружать оперативно по мере необходимости в процессе вычислений [5].

Запишем на алгоритмическом псевдоязыке алгоритмы и получим оценки временной и табличной сложностей рассмотренных операций сложения и умножения модулярных рациональных дробей, учитывая оценки сложности алгоритма деления. В рассматриваемых операциях используется единый набор констант.

Algorithm Addition

На входе алгоритма модулярные представления дробей $\frac{A}{P} \leftrightarrow (\ldots\alpha_i\ldots); \frac{B}{P} \leftrightarrow (\ldots\beta_i\ldots)$.

1. $i \leftarrow 1; m \leftarrow 0$
2. $\mu_i \leftarrow \left|\alpha_i + \beta_i\right|_{p_i}$
3. $M(i) \leftarrow \mu_i$
4. $m \leftarrow m + \left[\frac{\alpha_i + \beta_i}{p_i}\right]$
5. $i \leftarrow i + 1$
6. *if* $i \leqslant n$ *then goto* 2 *else* 7
7. $rang(A) \leftarrow Call\ \ Rang(A)$
8. $rang(B) \leftarrow Call\ \ Rang(B)$
9. $rang(M) \leftarrow Call\ \ Rang(M)$
10. $priznak \leftarrow rang(A) + rang(B) - rang(M) - m$
11. *exit*: $M(1:n); priznak$
12. *comment*: $\frac{M(1:n)}{P} \leftrightarrow (\ldots\mu_i\ldots);$
    $priznak = rang(A) + rang(B) - rang(M) - m$

Сложность операции сложения с определением переполнения

$$time = 3T_{rang} + O(1),\ const = O(n^2),$$

где T_{rang} — временная сложность операции вычисления ранга — позиционной характеристики модулярной числовой величины. При определенных условиях операция вычисления ранга имеет линейную сложность, получим $time = O(n) + O(1)$, $const = O(n^2)$.

Algorithm Multiplication (for fraction)

На входе алгоритма

$$\frac{A}{P} \leftrightarrow (\alpha_1,\ldots\alpha_i\ldots); \frac{B}{P} \leftrightarrow (\beta_1,\ldots\beta_i\ldots)$$

1. $i \leftarrow 1$
2. $\mu_i \leftarrow \left|\alpha_i \cdot \beta_i\right|_{p_i^2}$
3. $C(i) \leftarrow \mu_i$
4. $i \leftarrow i + 1$
5. *if* $i \leqslant n$ *then goto* 2 *else* 6
6. $W = Call\ \ Division(C)$
7. *exit*: $W(1:n)$
8. *comment*: $W(1:n) \leftrightarrow (\ldots\omega_i\ldots)(\mathrm{mod}\ P)$

Найдем сложность операции умножения с округлением результата до дроби со знаменателем — одинарным модулярным диапазоном $time = O(n^2) + O(1)$; $const = O(n^2)$.

Рассмотрим модулярные рациональные дроби с произвольным знаменателем. Определим вычет от рационального числа по простому модулю p следующим образом:

$$\left|\frac{a}{b}\right|_P = \left|a \cdot \left|b\right|_P^{-1}\right|_P = n \in \{0,1,\ldots p-1\} \subset \mathbf{N};\ (b,p) = 1.$$

При условии $p > \max\{a, b\}$ однозначно восстанавливается числитель рациональной дроби при известном знаменателе, что обеспечивает биективность отображения.

В операции сложения вычетов для однозначности восстановления результата, т.е. перехода от вычетов к рациональным дробям, условием изоморфизма отображения является неравенство $p > \max\{ad + cb, bd\}$:

$$\left|\left|\frac{a}{b}\right|_P + \left|\frac{c}{d}\right|_P\right|_p = \left|(ad + cb)\left|d\right|_P^{-1}\left|b\right|_P^{-1}\right|_P =$$

$$= \left|(a\left|b\right|_P^{-1} + c\left|d\right|_P^{-1})\right|_P = \left|\frac{ad + cb}{bd}\right|_P =$$

$$= \left|n_1 + n_2\right|_P = n \in \{0,1,\ldots p-1\} \subset \mathbf{N}.$$

Условие изоморфизма для дробно-рационального модулярного представления имеет вид $\tilde{P} \geqslant \max\left\{\sum_{i=1}^{n} \frac{\alpha_i}{p_i} P, P\right\}$ или $\tilde{P} \geqslant nP$, что соответствует условию точного вычисления нормированного ранга [6].

Заключение

В классах вычислительных процессов, в которых операнды и результаты операций являются рациональными дробями, модулярные представления для правильных и неправильных рациональных дробей позволяют организовать высокопроизводительный параллельный вычислительный процесс на системах с SIMD-архитектурой. Такие вычислительные архитектуры в явном или латентном виде используются в современных многопроцессорных системах. Эти системы содержат в своем составе множество центральных процессоров CPU и графических ускорителей GPU, что позволяет эффективно организовать параллельные многоядерные вычислительные процессы с данными в целочисленных и дробно-рациональных модулярных форматах.

Введенные соотношения показывают непосредственную связь дробно-рационального модулярного представления числовых величин с вычетами по модулю от правильных рациональных дробей. Использование квадратичной модулярной системы для отдельных операций в модулярной арифметике позволяет ряд вычислительно-сложных процедур выполнять с квадратичной временной и табличной сложностью, что удовлетворяет требованиям, предъявляемым к аппаратным и программным реализациям модулярной арифметики [7].

Список литературы

1. **Амербаев В. М.** Теоретические основы машинной арифметики. Алма-Ата: Наука, 1976. 320 с.
2. **Инютин С. А.** Метод вычисления характеристики отношения порядка для параллельных форматов данных // Информационные технологии. Т. 24, № 7. 2017. С. 343—347.
3. **Ноден П., Китте К.** Алгебраическая алгоритмика. М.: Мир, 1999. 720 с.
4. **Инютин С. А.** Основы модулярной алгоритмики. Ханты-Мансийск: Полиграфист, 2009. 237 с.
5. **Столярский Е. З., Шилов В. В.** Организация и работа кэш-памяти // Информационные технологии. 2000. № 7. С. 2—8.
6. **Инютин С. А.** Анализ сложности многоразрядных вычислительных процессов // Научные труды МАТИ. 2014. Вып. 22 (94). С. 154—159.
7. **Амербаев В. М., Стемпковский А. Л., Соловьев Р. А.** Принципы рекурсивных модулярных вычислений // Информационные технологии. 2013. № 2. С. 22—27.

S. A. Inyutin, Dr. Tech. Sc. (PhD), Full Professor, e-mail: inyutin_int@mail.ru,
Moscow Aviation Institute (Nation Research University) (MAI)

Fraction-Rational Constructions in Computer Modular Arithmetic

The expansion of modular representations to a set of rational fractions with a fixed denominator is analyzed. Methods have been developed for performing basic computer operations for such representations in the bit grid of a computational reconfigurable system. Given in pseudo-language and justified algorithms for their implementation, obtained estimates of their temporal and tabular complexities. The expediency of using a quadratic modular system to reduce the complexity of individual iterative procedures that underlie operations on the introduced modular representations and which have quadratic complexity has been proved. The representations introduced are intended for use in modular reconfigurable computing SIMD systems.

Keywords: *multiprocessor reconfigurable calculation systems SIMD architecture, modular computational process, quadratic computational complexity, parallel computer formats for rational fractions*

DOI: 10.17587/it.25.515-521

References

1. **Amerbaev V. M.** Theoretic base computer arithmetic, Alma-Ata, Nauka, 1976, 320 p. (in Russian).
2. **Inyutin S. A.** *Informatsionnyie Tehnologii*, 2017, no. 7, vol. 24, pp. 343—347 (in Russian).
3. **Noden P., Kitte K.** Algebra algorithmic, Moscow, Mir, 1999, 720 p. (in Russian).
4. **Inyutin S. A.** Base at modular algorithmic, Hantyi-Mansiysk, Poligrafist, 2009, 237 p. (in Russian).
5. **Shilov V. V., Stolyarskiy E. Z.** *Informatsionnyie Tehnologii*, 2000, no. 7, pp. 2—8 (in Russian).
6. **Inyutin S. A**. *Nauchnyie Trudyi MATI*, 2014, vol. 22 (94), pp. 154—159 (in Russian).
7. **Amerbaev V. M., Stempkovsky A. L., Solovjev R. A.** Informatsionnyie tehnologii, 2013, no. 2, vol. 5, pp. 22—27 (in Russian).

УДК 004.722.23 DOI: 10.17587/it.25.522-530

А. Ю. Романов, канд. техн. наук, доц., e-mail: a.romanov@hse.ru,
Е. А. Ведмидь, студент, e-mail: eavedmid@edu.hse.ru,
Национальный исследовательский университет "Высшая школа экономики", Москва,
Э. А. Монахова, канд. техн. наук, доц., ст. науч. сотр., e-mail: emilia@rav.sscc.ru,
Институт вычислительной математики и математической геофизики СО РАН, Новосибирск

Проектирование сетей на кристалле с топологией кольцевой циркулянт с тремя образующими: разработка алгоритмов маршрутизации

Представлена реализация нескольких алгоритмов маршрутизации динамического типа, предназначенных для использования в сетях на кристалле с циркулянтной топологией типа $C(N; 1, s_2, s_3)$ для поиска кратчайших маршрутов между любыми двумя узлами сети. Разработанные алгоритмы могут быть реализованы в виде цифровых автоматов для выбора направления движения пакетов в маршрутизаторах. Проведено тестирование алгоритмов на различных наборах оптимальных циркулянтов и выполнено их сравнение по эффективности, скорости и занимаемым в памяти ресурсам.

Ключевые слова: *сеть на кристалле, алгоритм Дейкстры, кольцевой циркулянт с тремя образующими, алгоритмы маршрутизации*

Введение

Одним из важных и актуальных направлений исследований в области информатики и вычислительных систем на современном этапе является построение многоядерных процессоров [1]. При этом в условиях роста популярности технологий построения систем на кристалле (Systems on Chip, SoCs) [2] и мультипроцессорных систем на кристалле (Multi-Processor Systems on Chip, MPSoCs) [3] также возрастает и распространение сетей на кристалле (Networks-on-Chip, NoCs, СтнК) [4], которые способны решать множество проблем, присущих системам, построенным по технологии общей шины [5].

Одной из самых актуальных и важных проблем в области исследований сетей на кристалле является поиск оптимальных топологий для их построения. Стандартные регулярные топологии, такие как mesh, torus, hypercube или spidergon [6—8], не всегда способны удовлетворить современным требованиям к сетям на кристалле [9, 10]. Поэтому остро стоит вопрос поиска новых топологий, и перспективными выглядят циркулянтные топологии [11], поскольку они имеют лучшие характеристики по сравнению с классическими топологиями [12]. При этом требуется разработка простых алгоритмов маршрутизации, применимых в циркулянтных сетях.

Использовать классический алгоритм Дейкстры для маршрутизации в сетях на кристалле слишком ресурсоемко из-за большой сложности реализации алгоритма на уровне маршрутизатора СтнК или IP-ядра [13, 14]. При табличной же маршрутизации необходимо хранить всю маршрутную информацию на уровне каждого маршрутизатора, что также является ресурсозатратным решением [2]. Поэтому задачей данной работы является разработка простых алгоритмов различных типов, которые могут быть реализованы в виде RTL цифровых автоматов на уровне маршрутизаторов в СтнК.

1. Кольцевые циркулянты с тремя образующими

Циркулянтные графы вида $C(N; 1, s_2, s_3)$, где $2 \leqslant s_2 < s_3 < N$, называются кольцевыми циркулянтами с тремя образующими, являющимися частным случаем кольцевых графов [15, 16]. Пример такого графа приведен на рис. 1 (см. вторую сторону обложки).

Циркулянты с тремя образующими являются перспективной топологией для проектирования сетей на кристалле. Так же, как и для топологий 3D-mesh и 3D-torus [12], маршрутизаторы в таких сетях содержат по шесть внешних портов, но представление таких топологий возможно в двумерном виде, что лучше всего

подходит для современных ASIC и ПЛИС, выполняемых по планарной технологии. Кроме того, циркулянты обладают лучшими характеристиками диаметра и среднего расстояния между узлами по сравнению с классическими регулярными топологиями [17].

1.1. Разработка структуры пакета при статической маршрутизации в сетях с топологией на основе кольцевых циркулянтов с тремя образующими

В большинстве сетей на кристалле используется парная маршрутизация [18], когда пакет передается из маршрутизатора источника данных в маршрутизатор узла назначения. Для организации такой маршрутизации можно воспользоваться любым алгоритмом поиска кратчайшего пути, например алгоритмом Дейкстры [16, 19]. Обычно используют статический тип маршрутизации [2], где каждый маршрутизатор каждого узла хранит список, каждый элемент которого является одним из узлов сети и представляет собой другой список с номерами узлов, с которыми соединен данный узел. При этом каждый маршрутизатор знает свой порядковый номер. На вход маршрутизатору, который должен будет передать пакет с данными, поступает номер узла назначения (маршрутизатора приемника). Маршрутизатор знает структуру сети и поэтому может рассчитать кратчайший путь по одному из алгоритмов.

Суть алгоритма Дейкстры [18] заключается в следующем: каждой вершине сопоставляется метка, которая содержит минимальное известное расстояние от данной вершины до вершины A (если расстояние неизвестно, то оно считается равным бесконечности или достаточно большому числу, чтобы можно было считать его бесконечно большим). Алгоритм пошагово перебирает каждую вершину и проверяет, можно ли с помощью этой вершины (с помощью пути от стартовой вершины до текущей) уменьшить расстояние (метку) от начального узла до вершины-соседа. Работа алгоритма Дейкстры длится до тех пор, пока все вершины не будут посещены.

Алгоритм Дейкстры достаточно универсален и подходит для любых графов, на основе которых и построены циркулянты, а следовательно, он подойдет для любых типов циркулянтов, включая кольцевые с любым числом и значением образующих [13]. Основная проблема данного алгоритма заключается в том, что при увеличении числа узлов время работы и потребляемая память значительно увеличиваются. Поэтому существует необходимость в разработке специализированного алгоритма, который позволял бы маршрутизаторам рассчитывать следующий шаг пакета в сети на основе его маршрутной информации. В литературе довольно мало работ, посвященных поиску кратчайших путей в циркулянтах с тремя образующими. Существуют решения [20, 21] для отдельных семейств циркулянтов с порядком $N = O(3d^2)$, где d — диаметр. Также в работе [22] представлен простой аналитический метод поиска кратчайшего пути в циркулянтах максимального порядка при заданном диаметре. Универсального алгоритма маршрутизации для циркулянтов с тремя образующими нет, так же как и для семейства кольцевых циркулянтов.

1.2. Разработка специализированных алгоритмов маршрутизации для сетей на кристалле на основе топологии кольцевой циркулянт с тремя образующими

Число портов маршрутизатора определяется степенью вершин циркулянта как $p = 2k$, где k — размерность графа (число его образующих) [12]. Таким образом, маршрутизатор циркулянта вида $C(N; 1, s_2, s_3)$ имеет шесть соединений с другими маршрутизаторами.

Самым очевидным алгоритмом для навигации в циркулянтных сетях можно считать алгоритм табличной маршрутизации, описанный в работе [16]. Таблица маршрутизации представляет собой квадратную матрицу $N \times N$, где N — число узлов (маршрутизаторов), а ячейки содержат номера портов, в которые нужно отправить пакет, чтобы он достиг узла назначения. Каждый маршрутизатор хранит только свою строку из таблицы.

В работе [16] количество памяти, которое занимает такая таблица, описано с помощью функции

$$M = N^2 \lceil \log_2 p \rceil, \qquad (1)$$

где N — число узлов в сети; $\lceil \log_2 p \rceil$ — необходимое количество памяти в битах для хранения номеров портов маршрутизатора; p — число портов маршрутизатора.

С одной стороны, использование алгоритма табличной маршрутизации требует хранения больших объемов данных, а с другой — реализация такого алгоритма в виде цифрового автомата не занимает много места на кристалле и является довольно простой.

1.3. Алгоритм обхода графа по часовой стрелке

Для навигации в циркулянтных сетях можно использовать алгоритм, который основан на итерационном вычислении маршрута между узлами, при котором каждый маршрутизатор принимает решение о коммутации пакета в следующий маршрутизатор только на один шаг. Поскольку циркулянты симметричны, для любого узла не важен его порядковый номер, а только расстояние в хопах (переходах) от него к другим узлам. Поэтому для уменьшения размера адресного поля в пакете в качестве адреса передается разница между номерами узлов (источника данных и приемника). Нагрузка на пакет (размер адресного поля в битах) может быть вычислена по следующей формуле:

$$P = \lceil \log_2 N \rceil, \quad (2)$$

где N — число узлов в сети.

Дополнительно в маршрутизаторе необходимо хранить число узлов и значения образующих s_2 и s_3. Таким образом, общий размер хранимых данных может быть вычислен по следующей формуле:

$$M = N\left(\lceil \log_2 N \rceil + \left\lceil \log_2 \frac{N}{2} \right\rceil + \left\lceil \log_2 \left(\frac{N}{2} - 1 \right) \right\rceil\right), \quad (3)$$

где N — число узлов в сети; $\lceil \log_2 N \rceil$ — необходимое количество памяти в битах для хранения числа маршрутизаторов в сети; $\left\lceil \log_2 \frac{N}{2} \right\rceil$ — необходимое количество памяти в битах для хранения образующей s_3; $\left\lceil \log_2 \left(\frac{N}{2} - 1 \right) \right\rceil$ — необходимое количество памяти в битах для хранения образующей s_2.

Для хранения значения образующей s_3 необходимо $\left\lceil \log_2 \frac{N}{2} \right\rceil$ бит, так как образующая s_3 гарантированно будет меньше, чем половина числа узлов [3], а образующая s_2 будет как минимум на единицу меньше, чем s_3.

Вычисление перехода происходит следующим образом: сначала определяется направление перехода (по направлению или против направления движения часовой стрелки), затем происходит выбор образующей. Если разница между узлом-источником и узлом-приемником меньше, чем половина числа узлов, то выбирается движение в направлении движения часовой стрелки, если больше — то в противоположном направлении. Если выбрано направление движения по часовой стрелке, то выбор образующей происходит следующим образом:

- пока разница между узлом-источником и узлом-приемником больше значения s_3, переход будет происходить по бóльшей образующей;
- если больше s_2, но меньше s_3, то переход будет осуществляться по образующей s_2, в противном случае — по образующей $s_1 = 1$.

Значение адресного поля головного флита пересчитывается путем вычитания длины образующей, по которой будет выполнен переход. Равенство нулю значения адресного поля головного флита является критерием окончания передачи пакета. Если было выбрано движение против часовой стрелки, общий алгоритм выбора текущего шага такой же, но сравнение происходит между образующими и разницей значения числа узлов в сети со значением, хранящимся в адресном поле головного флита. Перед переходом к адресному значению в головном флите прибавляется значение длины образующей, по которой произойдет переход. Критерием окончания передачи пакета в данном случае является равенство значения адресного поля головного флита числу узлов в сети. Предложенный алгоритм имеет много общего с похожим алгоритмом для двумерных кольцевых циркулянтов, описанным в работе [23].

Описание предложенного алгоритма приведено ниже:

algorithm Find_route_Clockwise is
Input: *startNode* — start node, *endNode* — end node, N — count of nodes, s_1 — first generator, s_2 — second generator, s_3 — third generator.
Output: *startNode* — next start node.

```
1:  S ← endNode − startNode
2:  If S = 0 then
3:     return startNode
4:  If S < 0 then
5:     S ← S + N
6:  If S ⩽ N/2 then
7:     If S ⩾ s3 then
8:        startNode ← (s3 + startNode) modN
9:     else
10:       If S ⩾ s2 then
11:          startNode ← (s2 + startNode) modN
12:       else
13:          startNode ← (s1 + startNode) modN
14: else
15:    S ← N − S
16:    If S ⩾ s3 then
17:       startNode ← (N − s3 + startNode) modN
18:    else
19:       If S ⩾ s2 then
20:          startNode ← (N − s2 + startNode) modN
21:       else
22:          startNode ← (N − s1 + startNode) modN
23: If startNode = 0 then
24:    startNode ← N
25: return startNode
```

Представленный алгоритм не является оптимальным, так как в некоторых случаях он будет предлагать пути, длина которых в хопах больше диаметра сети, но зато будет существенно экономить занимаемую маршрутизатором память.

Можно немного оптимизировать данный алгоритм следующим образом: проводить сравнение разности источника и приемника с $\frac{s_3+s_2}{2}$ и $\frac{s_1+s_2}{2}$. Если разность больше первого значения, то переход будет осуществляться по образующей s_3, если разность находится между этими значениями, то по s_2, иначе — по s_1.

Общий размер хранимых данных для этого алгоритма будет равен базовому (3). Данный алгоритм работает все еще недостаточно эффективно.

1.4. Алгоритм выбора направлений

Развитием предложенного подхода является алгоритм выбора направлений, который может менять направление движения при вычислении перехода в следующий узел по аналогии с тем, как делается в работе [23]. В данном алгоритме предлагается в качестве адреса в головном флите хранить номер узла назначения. Нагрузка на пакет остается той же, что и в алгоритме обхода графа по часовой стрелке (2). Дополнительно в маршрутизаторе необходимо хранить его номер, число узлов в сети и длину образующих s_2 и s_3. Общий размер хранимых данных вычисляется по формуле

$$M = N\left(2\lceil\log_2 N\rceil + \left\lceil\log_2\frac{N}{2}\right\rceil + \left\lceil\log_2\left(\frac{N}{2}-1\right)\right\rceil\right), \quad (4)$$

где N — число узлов в сети; $\lceil\log_2 N\rceil$ — необходимое количество памяти в битах для хранения номера маршрутизатора и числа маршрутизаторов в сети; $\left\lceil\log_2\frac{N}{2}\right\rceil$ — необходимое количество памяти в битах для хранения образующей s_3; $\left\lceil\log_2\left(\frac{N}{2}-1\right)\right\rceil$ — необходимое количество памяти в битах для хранения образующей s_2.

Работу алгоритма логически можно разделить на две части. В первой части происходит выбор последовательности передачи номеров узлов источника и приемника пакета, которые передаются во вторую часть алгоритма. Это возможно из-за того, что граф является неориентированным и вершинно-транзитивным [12]. Данная процедура требуется для упрощения алгоритма расчета направления движения из-за того, что он будет работать только с положительными числами. Кроме того, в первой части алгоритма происходит нормализация полученного направления движения пакета. Во второй части алгоритма непосредственно происходит вычисление следующего шага движения пакета. Алгоритмическое описание приведено ниже.

algorithm Find_Route_Selection is
Input: *startNode* — start node, *endNode* — end node, N — count of nodes, s_1 — first generator, s_2 — second generator, s_3 — third generator.
Output: startNode — next start node.
1: If star*tNode* > *endNode* **then**
2: *startNode* ← *startNode* − *Step*(*endNode*, *startNode*, N, s_1, s_2, s_3)
3: **else**
4: *startNode* ← *startNode* + *Step*(*endNode*, *startNode*, N, s_1, s_2, s_3)
5: **If** *startNode* > N **then**
6: *startNode* ← *startNode* − N
7: **else**
8: **If** *startNode* ⩽ 0 **then**
9: *startNode* ← *startNode* + N
10: **return** ***startNode***

function Step is
Input: *startNode* — start node, *endNode* — end node, N — count of nodes, s_1 — first generator, s_2 — second generator, s_3 — third generator.
Output: the function returns the best step (direction is also selected)
1: $bestWayR \leftarrow 0$, $stepR \leftarrow 0$, $bestWayL \leftarrow 0$, $S \leftarrow endNode - startNode$
2: $R_1 \leftarrow \frac{S-1}{s_2} + S \bmod N$, $R_2 \leftarrow \frac{S-1}{s_2} - S \bmod s_2 + s_2 + 1$
3: $R_3 \leftarrow \frac{S-1}{s_2} - \left(s_2 * \left(\frac{S}{s_2}+1\right) - S\right) + s_2 + 1$,
$R_4 \leftarrow \frac{S-1}{s_3} - S \bmod s_3 + s_3 + 1$
4: $R_5 \leftarrow \frac{S-1}{s_3} - \left(s_3 * \left(\frac{S}{s_3}+1\right) - S\right) + s_3 + 1$
5: **If** $R_3 > R_2$ **then**
6: $R_3 > R_2$
7: **If** $R_5 > R_4$ **then**
8: $R_5 > R_4$
9: **If** $R_1 < R_3$ **then**
10: **If** $R_1 < R_5$ **then**
11: $bestWayR \leftarrow R_1$, $stepR \leftarrow s_1$
12: **else**
13: $bestWayR \leftarrow R_5$, $stepR \leftarrow s_3$
14: **else**
15: **If** $R_3 < R_5$ **then**
16: $bestWayR \leftarrow R_3$, $stepR \leftarrow s_2$
17: **else**
18: $bestWayR \leftarrow R_5$, $stepR \leftarrow s_3$
19: $S \leftarrow endNode - startNode + N$
20: $L_1 \leftarrow \frac{S-1}{s_2} + S \bmod N$, $L_2 \leftarrow \frac{S-1}{s_2} - S \bmod s_2 + s_2 + 1$
21: $L_3 \leftarrow \frac{S-1}{s_2} - \left(s_2 * \left(\frac{S}{s_2}+1\right) - S\right) + s_2 + 1$,
$L_4 \leftarrow \frac{S-1}{s_3} - S \bmod s_3 + s_3 + 1$

22: $L_5 \leftarrow \frac{S-1}{s_3} - \left(s_3 * \left(\frac{S}{s_3} + 1 \right) - S \right) + s_3 + 1$
23: **If** $L_3 > L_2$ **then**
24: $L_3 > L_2$
25: **If** $L_5 > L_4$ **then**
26: $L_5 > L_4$
27: **If** $L_1 < L_3$ **then**
28: **If** $L_1 < L_5$ **then**
29: $bestWayL \leftarrow L_1$, $stepL \leftarrow -s_1$
30: **else**
31: $bestWayL \leftarrow L_5$, $stepL \leftarrow -s_3$
32: **else**
33: **If** $L_3 < L_5$ **then**
34: $bestWayL \leftarrow L_3$, $stepL \leftarrow -$s2
35: **else**
36: $bestWayL \leftarrow L_5$, $stepL \leftarrow -s_3$
37: **If** $bestWayR < bestWayL$ **then**
38: **return stepR**
39: **else**
40: **return stepL**

В предложенном алгоритме учтены циклы и ситуации, когда выгоднее сначала перейти по старшей образующей, а потом вернуться по средней. Тем не менее при проверке алгоритма на графах оказалось, что он все равно не в состоянии всегда гарантировать, что пакет будет двигаться по кратчайшему пути между узлами.

Поэтому был разработан еще один вариант алгоритма, принцип работы которого приближен к алгоритму Дейкстры, но учитывает особенности рассматриваемых циркулянтов [19].

1.5. Алгоритм поиска коэффициентов при образующих графа

Можно представить поиск кратчайшего пути для топологии, основанной на кольцевом циркулянте, как следующую оптимизационную задачу:

$$Nk + s = a_1 + a_2 s_2 + a_3 s_3, \quad (5)$$

где N — число узлов в сети; k — номер рассматриваемого цикла (может быть отрицательным); s — длина пути от узла-источника к узлу-приемнику; s_2, s_3 _ образующие; a_1, a_2, a_3 — коэффициенты при образующих s_1, s_2, s_3 соответственно.

Задача оптимизации заключается в минимизации суммы абсолютных величин коэффициентов a_1, a_2, a_3. Если выразить все переменные через переменную a_1, получится уравнение, где остается три неизвестных — a_2, a_3, k:

$$a_1 = a_2 s_2 + a_3 s_3 - Nk - s. \quad (6)$$

Далее нужно выбрать, в каких границах будут изменяться переменные a_2, a_3, k, и с помощью простого циклического перебора находить значение a_1, после чего выбирать тот набор коэффициентов, сумма которых наименьшая.

Общий размер хранимых данных для такого алгоритма вычисляется по формуле

$$M = N\left(2\lceil \log_2 N \rceil + \left\lceil \log_2 \frac{N}{2} \right\rceil + \right.$$
$$\left. + \left\lceil \log_2 \left(\frac{N}{2} - 1 \right) \right\rceil + \lceil \log_2 \alpha \rceil + \lceil \log_2 \beta \rceil + \lceil \log_2 \tau \rceil \right), \quad (7)$$

где N — число узлов в сети; $\lceil \log_2 N \rceil$ — необходимое количество памяти в битах для хранения номера маршрутизатора и числа маршрутизаторов в сети; $\left\lceil \log_2 \frac{N}{2} \right\rceil$ — необходимое количество памяти в битах для хранения образующей s_3; $\left\lceil \log_2 \left(\frac{N}{2} - 1 \right) \right\rceil$ — необходимое количество памяти в битах для хранения образующей s_2; α, β, τ — коэффициенты, отвечающие за число циклов и образующих, которые будут рассматриваться в алгоритме, соответственно; $\lceil \log_2 \alpha \rceil + \lceil \log_2 \beta \rceil + \lceil \log_2 \tau \rceil$ — необходимое количество памяти в битах для хранения коэффициентов α, β, τ.

Коэффициент τ задается вручную как число циклов, которые могут быть пройдены в обе стороны. Тогда коэффициенты $\alpha = \left\lceil \frac{\tau N}{s_3} \right\rceil$ и $\beta = \left\lceil \frac{\tau N}{s_2} \right\rceil$.

Алгоритмическое описание данного варианта приведено ниже.

function Find_route_coefficients is
Input: *startNode* — start node, *endNode* — end node, N — count of nodes, s_1 — first generator, s_2 — second generator, s_3 — third generator.
Output: the function returns the best coefficients a_1, a_2, a_3
1: $bestA1 \leftarrow$ maxint, $bestA2 \leftarrow$ maxint, $bestA3 \leftarrow$ maxint
2: $S \leftarrow endNode - startNode$, $a1 \leftarrow 0$
3: $zero \leftarrow 10$, $alpha \leftarrow (zero*N)divs_3$, $beta \leftarrow (zero*N)divs_2$
4: **For all** $k \in (-zero, zero)$:
5: **For all** $a_3 \in (-alpha, alpha)$:
6: **For all** $a_2 \in (-beta, beta)$:
7: $a_1 \leftarrow k*N + S - a_3*s_3 - a_2*s_2$
8: **If** $|a_1| + |a_2| + |a_3| < |bestA1| + |bestA2| + |bestA3|$ **then**
9: $bestA1 \leftarrow a1$, $bestA2 \leftarrow a2$, $bestA3 \leftarrow a3$
10: **return** $bestA1$, $bestA2$, $bestA3$

2. Тестирование предложенных алгоритмов

Для оценки работы алгоритмов предлагается использовать критерий эффективности, учитывающий число требуемых шагов для

прохождения пакетом из первого узла во все остальные [18]. В качестве оптимального алгоритма принят алгоритм Дейкстры [19], который гарантированно находит кратчайшие пути между всеми узлами любого связного графа. Следовательно, критерий эффективности определяется формулой

$$K = \frac{\sum_{i=1}^{N-1} H^{D}_{(0-i)}}{\sum_{i=1}^{N-1} H^{A}_{(0-i)}}, \qquad (8)$$

где N — число узлов в сети; $\sum_{i=1}^{N-1} H^{A}_{(0-i)}$ — число хопов из нулевого узла во все остальные, рассчитанное по используемому алгоритму; $\sum_{i=1}^{N-1} H^{D}_{(0-i)}$ — число хопов из нулевого узла во все остальные, рассчитанное по алгоритму Дейкстры.

Для тестирования алгоритмов выбраны кольцевые циркулянты вида $C(N; 1, s_2, s_3)$, где N — число узлов в сети — определяется формулой $N = n^2$ при $N \leqslant 100$ и 150, 200, 300, 400, 500 (n — натуральное число, чтобы можно было сравнить результаты работы сетей с топологией кольцевого циркулянта с тремя образующими с топологиями torus и mesh). Тестируемые циркулянты являются оптимальными и имеют минимальный диаметр среди кольцевых циркулянтов с таким же числом узлов [16]. Результаты тестирования приведены в табл. 1.

Для алгоритмов Дейкстры и табличной маршрутизации коэффициент эффективности будет всегда равен 1, так как первый из них является эталонным, а второй предполагает наличие оптимального пути. Для алгоритма обхода по часовой стрелке эффективность сильно зависит от большей образующей и разности между образующими s_2 и s_3 — чем они больше, тем меньше эффективность алгоритма. Для алгоритма поиска коэффициентов эффективность оказалась равна 1 для всех циркулянтов.

В табл. 2 представлены результаты сравнения разработанных алгоритмов маршрутизации по длине максимального пути в графе.

Таким образом, для алгоритма обхода по часовой стрелке и его улучшенной версии полученный максимальный путь в графе отличается значительно в худшую сторону от аналогичного показателя, полученного с помощью алгоритма Дейкстры (для некоторых случаев в несколько раз). Алгоритм выбора направлений показывает лучшее качество, но в некото-

Таблица 1

Сравнение эффективности разработанных алгоритмов

Циркулянт	Алгоритм			
	Обход по часовой стрелке	Улучшенный обход по часовой стрелке	Выбор направлений	Поиск коэффициентов
C(9; 1, 2, 4)	1	1	1	1
C(16; 1, 4, 8)	0,818	1	1	1
C(25; 1, 6, 10)	0,742	0,821	0,939	1
C(36; 1, 8, 15)	0,656	0,687	0,955	1
C(49; 1, 10, 23)	0,527	0,685	0,887	1
C(64; 1, 12, 30)	0,481	0,643	0,909	1
C(81; 1, 15, 37)	0,474	0,646	0,959	1
C(100; 1, 17, 40)	0,441	0,588	0,781	1
C(100; 1, 10, 30)	0,689	1	1	1
C(150; 1, 33, 59)	0,329	0,536	0,795	1
C(200; 1, 56, 87)	0,291	0,525	0,804	1
C(300; 1, 74, 138)	0,148	0,279	0,837	1
C(400; 1, 69, 195)	0,195	0,321	0,968	1
C(400; 1, 65, 199)	0,342	0,368	0,988	1
C(500; 1, 34, 200)	0,537	0,947	0,968	1

Таблица 2

Сравнение максимальных путей в графе для разработанных алгоритмов

Циркулянт	Алгоритм				
	Дейкстры	Обход по часовой стрелке	Улучшенный обход по часовой стрелке	Выбор направлений	Поиск коэффициентов
C(9; 1, 3, 5)	2	2	2	2	2
C(16; 1, 4, 8)	3	4	3	3	3
C(25; 1, 6, 10)	3	5	4	3	3
C(36; 1, 8, 15)	4	7	5	5	4
C(49; 1, 10, 23)	4	10	6	5	4
C(64; 1, 12, 30)	4	12	7	6	4
C(81; 1, 15, 37)	5	15	9	6	5
C(100; 1, 17, 40)	6	17	10	9	6
C(100; 1, 10, 30)	7	11	7	7	7
C(150; 1, 33, 59)	8	32	17	11	8
C(200; 1, 56, 87)	12	55	28	15	12
C(300; 1, 74, 138)	8	73	37	10	8
C(400; 1, 69, 195)	11	66	36	13	11
C(400; 1, 65, 199)	9	66	34	23	9
C(500; 1, 34, 200)	18	37	19	20	18

рых случаях различие между значениями максимального пути графа достигает 14. Алгоритм поиска коэффициентов показывает такой же результат, как и эталонный алгоритм.

Работа алгоритма поиска коэффициентов была проверена на 502 оптимальных графах из датасета [24], полученных с помощью программного обеспечения, описанного в работе [16], в результате чего подтверждено, что при заданных коэффициентах α, β, τ, равных 10, 20 и 30 соответственно, эффективность алгоритма не опускается ниже 1.

Также была проведена проверка алгоритма выбора направлений на зависимость эффективности алгоритма от разницы между образующими s_2 и s_3. В результате тестирования алгоритма на данных работы [24], среди которых было 502 оптимальных кольцевых циркулянта, было получено, что при увеличении разницы между образующими s_2, s_3 прослеживается тенденция к снижению эффективности алгоритма.

На рис. 2 (см. вторую сторону обложки) представлен график зависимости между эффективностью алгоритма выбора направлений и разницей между его старшими образующими.

Исходя из приведенных выше формул (3), (4), (7) расчет памяти в битах, необходимой для работы алгоритма, приведен в табл. 3.

Рассматриваемые алгоритмы реализованы на языке программирования Python 3, что позволило оценить время их работы для поиска всех путей в кольцевом графе. Тестирование проведено на компьютере с операционной системой Windows 8.1, оперативной памятью 12 Гб и четырехъядерным процессором с частотой 2,4 ГГц. Алгоритм табличной маршрутизации не рассматривался, так как он не подразумевает сложных вычислений. Результаты алгоритма обхода по часовой стрелке практически не отличаются от результатов улучшенного алгоритма обхода по часовой стрелке и поэтому объединены в один столбец. Полученные показатели времени работы алгоритмов в миллисекундах представлены в табл. 4.

Заключение

Проведено исследование использования циркулянтных топологий размерности три для проектирования сетей на кристалле. Для кольцевых циркулянтов с тремя образующими разработан ряд алгоритмов парной маршрутизации: алгоритм на основе табличной маршру-

Таблица 3

Занимаемые ресурсы памяти в битах в зависимости от типа алгоритма

Циркулянт	Алгоритм			
	Табличная маршрутизация	Улучшенный обход по часовой стрелке	Выбор направлений	Поиск коэффициентов
C(9; 1, 3, 5)	243	108	144	270
C(16; 1, 4, 8)	768	192	256	480
C(25; 1, 6, 10)	1875	375	500	850
C(36; 1, 8, 15)	3888	648	864	1368
C(49; 1, 10, 23)	7203	882	1176	1862
C(64; 1, 12, 30)	12 288	1152	1536	2432
C(81; 1, 15, 37)	19 683	1701	2268	3402
C(100; 1, 17, 40)	30 000	2100	2800	4200
C(150; 1, 33, 59)	67 500	3600	4800	6900
C(200; 1, 56, 87)	120 000	4800	6400	9200
C(300; 1, 74, 138)	270 000	8100	10 800	15 000
C(400; 1, 65, 199)	480 000	10 800	14 400	20 000
C(500; 1, 34, 200)	750 000	13 500	18 000	25 000

Примечание: при подсчете памяти для алгоритма поиска коэффициентов константы α, β, τ выбраны равными 10, 20, 30 соответственно

Таблица 4

Время работы алгоритмов, мс

Циркулянт	Алгоритм		
	Улучшенного обхода по часовой стрелке	Выбора направлений	Поиска коэффициентов
C(9; 1, 3, 5)	0,003504	0,006079	28,323078
C(16; 1, 4, 8)	0,003981	0,012421	50,129890
C(25; 1, 6, 10)	0,005578	0,019598	96,979403
C(36; 1, 8, 15)	0,010770	0,036001	116,926932
C(49; 1, 10, 23)	0,018406	0,054717	153,540992
C(64; 1, 12, 30)	0,037407	0,080180	202,219200
C(81; 1, 15, 37)	0,044202	0,113415	261,775398
C(100; 1, 17, 40)	0,090599	0,269699	328,511905
C(150; 1, 33, 59)	0,298190	0,450205	510,957384
C(200; 1, 56, 87)	0,491786	0,728797	656,596207
C(300; 1, 74, 138)	0,867891	0,984096	994,344091
C(400; 1, 65, 199)	0,913595	2,111387	1324,184012
C(500; 1, 34, 200)	0,977516	2,547621	1682,586193

тизации, алгоритм обхода по часовой стрелке, алгоритм выбора направлений и оптимальный алгоритм, основанный на поиске коэффициентов. Показано, что классический алгоритм с табличной маршрутизацией в сетях на кристалле может быть заменен на алгоритм поиска коэффициентов при образующих, так как он обеспечивает такое же число хопов между узлами и является оптимальным, при этом его реализация на аппаратном уровне требует значительно меньше ресурсов памяти. Также альтернативными вариантами при низких требованиях к пропускной способности СтнК, но в условиях ограниченности аппаратных ресурсов, могут быть алгоритм обхода по часовой стрелке и его улучшенная версия, а также алгоритм выбора направлений, использование которых позволяет почти в два раза снизить затраты аппаратных ресурсов, но приводит к значительному увеличению длины максимального пути в графе и среднего расстояния между узлам.

Проведено сравнение сложности алгоритмов, а также ресурсов, занимаемых синтезированными подсистемами связи СтнК в ПЛИС. Поскольку предлагаемые алгоритмы, в отличие от классического алгоритма Дейкстры, не требуют рассчитывать весь путь прохождения пакета, а определяют только номер порта для следующего шага, гарантируя, что пакет достигнет узла назначения, они могут быть легко реализованы в виде цифрового автомата в маршрутизаторах для сетей на кристалле.

Публикация подготовлена в ходе проведения исследования (№ 18-01-0074) в рамках Программы "Научный фонд Национального исследовательского университета "Высшая школа экономики" (НИУ ВШЭ)" в 2018—2019 гг. и в рамках государственной поддержки ведущих университетов Российской Федерации "5-100".

Исследование Монаховой Э. А. выполнено в рамках проекта ИВМиМГ СО РАН 0315-2016-0006.

Список литературы

1. **Dahir N. S. et al.** Modeling and tools for power supply variations analysis in networks-on-chip // IEEE Trans. Comput. 2014. Vol. 63, N. 3. P. 679—690.

2. **Benmessaoud Gabis A., Koudil M.** NoC routing protocols — objective-based classification // J. Syst. Archit. Elsevier B. V., 2016. Vol. 66—67. P. 14—32.

3. **Paul J. et al.** Resource-awareness on heterogeneous MPSoCs for image processing // J. Syst. Archit. 2015. Vol. 61, N. 10. P. 668—680.

4. **Abdelfattah M. S., Bitar A., Betz V.** Design and Applications for Embedded Networks-on-Chip on FPGAs // IEEE Trans. Comput. 2017. Vol. 66, N. 6. P. 1008—1021.

5. **Angiolini F. et al.** A layout-aware analysis of networks-on-chip and traditional interconnects for MPSoCs // IEEE Trans. Comput. Des. Integr. Circuits Syst. 2007. Vol. 26, N. 3. P. 421—434.

6. **Marvasti M. B., Szymanski T. H.** The performance of hypermesh NoCs in FPGAs // IEEE International Conference on Computer Design: VLSI in Computers and Processors. 2012. P. 492—493.

7. **Bishnoi R. et al.** Distributed adaptive routing for spidergon NoC // 18th International Symposium on VLSI Design and Test, VDAT 2014. 2014. P. 1—6.

8. **Deb D. et al.** Cost effective routing techniques in 2D mesh NoC using on-chip transmission lines // J. Parallel Distrib. Comput. Academic Press, 2019. Vol. 123. P. 118—129.

9. **Dally W. J., Towles B. P.** Principles and Practices of Interconnection Networks. Elsevier, 2003. 581 p.

10. **Kumar A., Tyagi S., Jha C. K.** Performance analysis of network-on-chip topologies // J. Inf. Optim. Sci. 2017. Vol. 38, N. 6. P. 989—997.

11. **Vilfred V.** A few properties of circulant graphs: Self-complementary, isomorphism, Cartesian product and factorization // 2017 7th International Conference on Modeling, Simulation, and Applied Optimization, ICMSAO 2017. 2017. P. 1—5.

12. **Монахова Э. А.** Структурные и коммуникативные свойства циркулянтных сетей // Прикладная дискретная математика. 2011. Т. 3, № 13. С. 92—115.

13. **Cai J. Y. et al.** On routing in circulant graphs // Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, 1999. Vol. 1627. P. 360—369.

14. **Кузнецов Н. А., Фетисов В. Н.** Алгоритм Дейкстры с улучшенной робастностью для управления маршрутизацией в IP-сетях // Автоматика и телемеханика. 2008. Т. 2. С. 80—85.

15. **Монахова Э. А.** Мультипликативные циркулянтные сети // Дискретный анализ и исследование операций. 2010. Т. 17. С. 56—66.

16. **Romanov A. Yu., Romanova I. I., Glukhikh A. Yu.** Development of a Universal Adaptive Fast Algorithm for the Synthesis of Circulant Topologies for Networks-on-Chip Implementations // 2018 IEEE 38th International Scientific Conference on Electronics and Nanotechnology, ELNANO 2018. 2018. P. 110—115.

17. **Монахова Э. А., Монахов О. Г.** О некоторых характеристиках циркулянтных и тороидальных структур вычислительных систем // Вестник СибГУТИ. 2013. № 3. С. 63—69.

18. **Dijkstra E. W.** A note on two problems in connexion with graphs // Numer. Math. 1959. Vol. 1. P. 269—271.

19. **Левитин А. В.** Жадные методы: Алгоритм Дейкстры // Алгоритмы. Введение в разработку и анализ. 2006. С. 189—195.

20. **Barrière L. et al.** Fault-tolerant routings in chordal ring networks // Networks. 2000. Vol. 36, N. 3. P. 180—190.

21. **Liestman A. L., Opatrny J., Zaragoza M.** Network properties of double and triple fixed step graphs // Int. J. Found. Comput. Sci. 1998. Vol. 9, N. 1. P. 57—76.

22. **Monakhova E. A.** Optimal Triple Loop Networks with Given Transmission Delay: Topological Design and Routing // International Network Optimization Conference. Paris, 2003. P. 410—415.

23. **Romanov A. Yu.** Development of routing algorithms in networks-on-chip based on ring circulant topologies // Heliyon. Elsevier, 2019. Vol. 5, N. 4. P. e01516.

24. **Romanov A. Yu.** Optimal Circulants Dataset [Electronic resource]. URL: https://github.com/RomeoMe5/circulantGraphs/ (accessed: 21.03.2019).

A. Yu. Romanov, PhD, Associate Professor, e-mail: a.romanov@hse.ru,
E. A. Vedmid, Student, e-mail: eavedmid@edu.hse.com,
National Research University Higher School of Economics, Moscow, Russian Federation,
E. A. Monakhova, PhD, Associate Professor, Senior Researcher, e-mail: emilia@rav.sscc.ru,
ICMMG SB RAS, Novosibirsk, Russian Federation

Designing Networks-on-Chip Based on Triple Loop (Circulant) Networks: Routing Algorithm Development

This paper presents implementation of several dynamic routing algorithms designed for using in networks-on-chip based on circulant topology of type C(N; 1, s_2, s_3) to search for the shortest routes between nodes. The developed algorithms can be implemented as RTL state machine for choosing the direction of packets in routers. Algorithms were tested on various sets of optimal triple loop circulants and compared in terms of efficiency, speed, and resources held in memory. The relationship between efficiency and the difference between the two generatrices was obtained, and the most effective one was found — the coefficient search algorithm. For all tested circulants, algorithm shows maximum efficiency, but the execution time of this algorithm is significantly higher than its considered counterparts. In addition, the efficiency and speed of the algorithm directly depend on the chosen calculation coefficients. Compared with the classic Dijkstra algorithm, the proposed algorithms do not require calculation of the entire packet path, but determine the port number to which the packet should be sent, so that it can reach the destination node. This makes it possible to significantly simplify the structure of the network-on-chip router.

Keywords: *network-on-chip, Dijkstra's algorithm, third loop circulants, routing algorithms*

DOI: 10.17587/it.25.522-530

References

1. **Dahir N. S.** et al. Modeling and tools for power supply variations analysis in networks-on-chip, *IEEE Trans. Comput.*, 2014, vol. 63, no. 3, pp. 679—690.

2. **Benmessaoud Gabis A., Koudil M.** NoC routing protocols — objective-based classification, *J. Syst. Archit. Elsevier B. V.,* 2016, vol. 66—67, pp. 14—32.

3. **Paul J.** et al. Resource-awareness on heterogeneous MPSoCs for image processing, *J. Syst. Archit.*, 2015, vol. 61, no. 10, pp. 668—680.

4. **Abdelfattah M. S., Bitar A., Betz V.** Design and Applications for Embedded Networks-on-Chip on FPGAs, *IEEE Trans. Comput.,* 2017, vol. 66, no. 6, pp. 1008—1021.

5. **Angiolini F.** et al. A layout-aware analysis of networks-on-chip and traditional interconnects for MPSoCs, *IEEE Trans. Comput. Des. Integr. Circuits Syst.*, 2007, vol. 26, no. 3, pp. 421—434.

6. **Marvasti M. B., Szymanski T. H.** The performance of hypermesh NoCs in FPGAs, *IEEE International Conference on Computer Design: VLSI in Computers and Processors*, 2012, pp. 492—493.

7. **Bishnoi R.** et al. Distributed adaptive routing for spidergon NoC, *18th International Symposium on VLSI Design and Test, VDAT 2014*, 2014, pp. 1—6.

8. **Deb D.** et al. Cost effective routing techniques in 2D mesh NoC using on-chip transmission lines, J. *Parallel Distrib. Comput. Academic Press*, 2019, vol. 123, pp. 118—129.

9. **Dally W. J., Towles B. P.** Principles and Practices of Interconnection Networks. Elsevier, 2003, 581 p.

10. **Kumar A., Tyagi S., Jha C. K.** Performance analysis of network-on-chip topologies, *J. Inf. Optim. Sci.*, 2017, vol. 38, no. 6, pp. 989—997.

11. **Vilfred V.** A few properties of circulant graphs: Self-complementary, isomorphism, Cartesian product and factorization, *2017 7th International Conference on Modeling, Simulation, and Applied Optimization, ICMSAO 2017,* 2017, pp. 1—5.

12. **Monakhova E. A.** Structural and communicative properties of circulant networks, *Prikladnaya Diskretnaya Matematika*, 2011, vol. 3, no. 13, pp. 92—115 (in Russian).

13. **Cai J. Y.** et al. On routing in circulant graphs, *Lecture Notes in Computer Science,* Springer, Berlin, Heidelberg, 1999, vol. 1627, pp. 360—369.

14. **Kuznetsov N. A., Fetisov V. N.** Algoritm Dijkstry s uluchshennoj robastnost'ju dlja upravlenija marshrutizaciej v IP-setjah, *Avtom. i Telemeh.*, 2008, vol. 2, pp. 80—85 (in Russian).

15. **Monakhova E. A.** Multiplicative circulant networks**,** *Diskretniy Analiz i Issledovanie Operaciy*, 2010, vol. 17, pp. 56—66 (in Russian).

16. **Romanov A. Yu., Romanova I. I., Glukhikh A. Yu.** Development of a Universal Adaptive Fast Algorithm for the Synthesis of Circulant Topologies for Networks-on-Chip Implementations, *2018 IEEE 38th International Scientific Conference on Electronics and Nanotechnology, ELNANO 2018*, 2018, pp. 110—115.

17. **Monakhova E. A., Monakhov O. G.** On some characteristics of circulant and toroidal structures of computing systems, *Vestnik SibGUTI*, 2013, no. 3, pp. 63—69 (in Russian).

18. **Dijkstra E. W.** A note on two problems in connexion with graphs, *Numer. Math.,* 1959, vol. 1, pp. 269—271.

19. **Levitin A. V.** Greedy methods: Dijkstra's Algorithm, *Algoritmi. Vvedenie v Razrabotku i Analiz*, 2006, pp. 189—195 (in Russian).

20. **Barrière L.** et al. Fault-tolerant routings in chordal ring networks, *Networks,* 2000, vol. 36, no. 3, pp. 180—190.

21. **Liestman A. L., Opatrny J., Zaragoza M.** Network properties of double and triple fixed step graphs, *Int. J. Found. Comput. Sci.,* 1998, vol. 9, no. 1, pp. 57—76.

22. **Monakhova E. A.** Optimal Triple Loop Networks with Given Transmission Delay: Topological Design and Routing, *International Network Optimization Conference,* Paris, 2003, pp. 410—415.

23. **Romanov A. Yu.** Development of routing algorithms in networks-on-chip based on ring circulant topologies, *Heliyon,* Elsevier, 2019, vol. 5, no. 4, pp. e01516.

24. **Romanov A. Yu.** Optimal Circulants Dataset, available at: https://github.com/RomeoMe5/circulantGraphs/ (accessed: 21.03.2019).

УДК 621.391.1: 621.395 DOI: 10.17587/it.25.531-537

В. Н. Тарасов, д-р техн. наук, проф., зав. каф., e-mail: veniamin_tarasov@mail.ru,
Н. Ф. Бахарева, д-р техн. наук, проф., зав. каф., e-mail: nadin1956_04@inbox.ru,
Када Отхмане, аспирант,
Поволжский государственный университет телекоммуникаций и информатики, г. Самара

Моделирование телетрафика на основе системы $HE_2/H_2/1$

Статья посвящена исследованию системы массового обслуживания (СМО) $HE_2/H_2/1$ типа G/G/1 с гиперэрланговским входным распределением второго порядка и гиперэкспоненциальным законом времени обслуживания в целях получения решения для среднего времени ожидания требований в очереди в случае стационарного режима. Для этого использован классический метод спектрального разложения решения интегрального уравнения Линдли. Для практического применения полученных результатов использован метод моментов. Оказывается, что гиперэрланговский закон распределения HE_2, как и гиперэкспоненциальный H_2, являющийся трехпараметрическим, может определяться как двумя первыми моментами, так и тремя первыми моментами. Выбор таких законов распределения вероятностей обусловлен тем, что они являются наиболее общими распределениями неотрицательных непрерывных случайных величин, поскольку коэффициент вариации для распределения HE_2 $c_\tau \geqslant 1/\sqrt{2}$ и охватывает более широкий диапазон, чем у гиперэкспоненциального распределения, для которого $c_\tau \geqslant 1$. Определение главной характеристики СМО типа G/G/1 — среднего времени ожидания — является актуальной задачей в связи с тем, что для такой СМО не существует решения в общем случае. Метод спектрального разложения решения интегрального уравнения Линдли для СМО $HE_2/H_2/1$ позволяет получить решение в замкнутой форме.

Ключевые слова: *гиперэрланговский и гиперэкспоненциальный законы распределения, интегральное уравнение Линдли, метод спектрального разложения, преобразование Лапласа*

Введение

Настоящая статья посвящена анализу систем массового обслуживания (СМО) $HE_2/H_2/1$ типа G/G/1 с произвольными законами распределений входного потока требований и времени обслуживания, для которых в общем случае не может быть найдено решение для главной характеристики — среднего времени ожидания требований в очереди. Поэтому системы типа G/G/1 могут быть исследованы только при конкретных законах распределений входного потока [1—4].

Как известно, например, из работы [1], для системы G/G/1 среднее время ожидания определяется выражением

$$\bar{W} = \frac{D_\lambda + D_\mu + (1-\rho)^2/\lambda^2}{2(1-\rho)/\lambda} - \frac{\overline{I^2}}{2\bar{I}}, \qquad (1)$$

где ρ — коэффициент загрузки системы ($0 < \rho = \lambda/\mu < 1$); λ — интенсивность входного потока; μ — интенсивность обслуживания; D_λ, D_μ — соответственно дисперсии интервалов поступления и времени обслуживания; $\bar{I}$, $\overline{I^2}$ — соответственно среднее значение и второй начальный момент периода простоя. Так как выражение (1) связано с коэффициентами вариаций интервалов поступления и обслуживания квадратичной зависимостью, роль последних для среднего времени ожидания значительна. Второе слагаемое в правой части (1) остается неизвестным, и вполне вероятно, что оно может зависеть от моментов интервалов поступления и времени обслуживания более высокого порядка, чем первые два. Поэтому при анализе СМО G/G/1 необходимо учитывать не только первые два момента случайных интервалов времен поступления и обслуживания, но и моменты более высокого порядка.

В теории телетрафика по среднему времени ожидания, например, оценивают задержки пакетов в сетях пакетной коммутации при их моделировании с помощью СМО.

В исследовании систем G/G/1 важную роль играет метод спектрального разложения решения интегрального уравнения Линдли, и большинство результатов в теории массового обслуживания получены именно с помощью данного метода. Обозначив:

$W(y)$ — функция распределения вероятностей (ФРВ) времени ожидания требования в очереди;

$C(u) = P(\tilde{u} < u)$ — ФРВ случайной величины $\tilde{u} = \tilde{x} - \tilde{t}$, где, в свою очередь, $\tilde{x}$ — случайное время обслуживания требования, $\tilde{t}$ — случайная величина — интервал времени между поступлениями требований, приведем одну из форм интегрального уравнения Линдли [1—4]:

$$W(y) = \begin{cases} \int\limits_{-\infty}^{y} W(y-u)dC(u), & y \geqslant 0; \\ 0, & y < 0. \end{cases}$$

В научной литературе нет данных по рассматриваемой системе, и, видимо, это связано с достаточной сложностью гиперэрланговского закона распределения. Выбор законов распределений HE_2 и H_2 можно обосновать несколькими причинами. Во-первых, они являются наиболее общими распределениями неотрицательных непрерывных случайных величин, поскольку для HE_R коэффициент вариации $c_\tau > 0$ [5, 6], в частности для HE_2 $c_\tau \geqslant 1/\sqrt{2}$. Как известно из работы [1], для гиперэкспоненциального закона распределения H_2 коэффициент вариации $c_\tau \geqslant 1$. Таким образом, оба рассматриваемых закона HE_2 и H_2 имеют широкий диапазон изменения коэффициентов вариаций. Кроме того, законы распределений HE_2 и H_2, начиная со значения коэффициента вариации, равного 4, имеют так называемый тяжелый хвост [5, 6] и, следовательно, в теории телетрафика могут быть использованы для описания трафика с "тяжелым" контентом: мультимедиа и др. И наконец, их отличительная особенность заключается в том, что они однозначно могут быть определены с использованием как двух первых моментов, так и трех моментов.

Постановка задачи

Ставится задача определения среднего времени ожидания в системе $HE_2/H_2/1$ на основе классического метода спектрального разложения решения интегрального уравнения Линдли (ИУЛ) для данной системы в замкнутой форме и, тем самым, дополнения известной формулы (1).

Решение задачи

Для системы $HE_2/H_2/1$ законы распределения интервалов входного потока и времени обслуживания задаются функциями плотности вида

$$a(t) = 4p\lambda_1^2 t\mathbf{e}^{-2\lambda_1 t} + 4(1-p)\lambda_2^2 t\mathbf{e}^{-2\lambda_2 t}; \qquad (2)$$

$$b(t) = q\mu_1\mathbf{e}^{-\mu_1 t} + (1-q)\mu_2\mathbf{e}^{-\mu_2 t}. \qquad (3)$$

Использование классического метода спектрального разложения решения ИУЛ, как это показано в работах [7—9], позволит определить не только среднее время ожидания, но и моменты высших порядков времени ожидания.

Согласно методу спектрального разложения нам для нахождения закона распределения времени ожидания необходимо найти следующее спектральное разложение: $A^*(-s)B^*(s) - 1 = \psi_+(s)/\psi_-(s)$, где $\psi_+(s)$ и $\psi_-(s)$ — некоторые рациональные функции от s, а $A^*(s)$, $B^*(s)$ — преобразования Лапласа функций плотности (2) и (3) соответственно.

Преобразования Лапласа функций (2) и (3) имеют следующий вид:

$$A^*(s) = p\left(\frac{2\lambda_1}{s+2\lambda_1}\right)^2 + (1-p)\left(\frac{2\lambda_2}{s+2\lambda_2}\right)^2;$$

$$B^*(s) = q\frac{\mu_1}{s+\mu_1} + (1-q)\frac{\mu_2}{s+\mu_2}.$$

Тогда спектральное разложение решения ИУЛ для системы $HE_2/H_2/1$ $A^*(-s)B^*(s) - 1 = \psi_+(s)/\psi_-(s)$ примет вид:

$$\frac{\psi_+(s)}{\psi_-(s)} = \left[p\left(\frac{2\lambda_1}{2\lambda_1 - s}\right)^2 + (1-p)\left(\frac{2\lambda_2}{2\lambda_2 - s}\right)^2\right] \times \left[q\frac{\mu_1}{\mu_1 + s} + (1-q)\frac{\mu_2}{\mu_2 + s}\right] - 1.$$

Выражение, стоящее в первых квадратных скобках, представим в виде

$$\left[p\left(\frac{2\lambda_1}{2\lambda_1 - s}\right)^2 + (1-p)\left(\frac{2\lambda_2}{2\lambda_2 - s}\right)^2\right] = \frac{p\left(16\lambda_1^2\lambda_2^2 - 16\lambda_1^2\lambda_2 s + 4\lambda_1^2 s^2\right)}{(2\lambda_1 - s)^2(2\lambda_2 - s)^2} + \frac{(1-p)\left(16\lambda_1^2\lambda_2^2 - 16\lambda_1\lambda_2^2 s + 4\lambda_2^2 s^2\right)}{(2\lambda_1 - s)^2(2\lambda_2 - s)^2} = \frac{a_0 - a_1 s + a_2 s^2}{(2\lambda_1 - s)^2(2\lambda_2 - s)^2},$$

где промежуточные параметры, введенные для сокращения записи, равны $a_0 = 16\lambda_1^2\lambda_2^2$, $a_1 = 16\lambda_1\lambda_2[p\lambda_1 + (1-p)\lambda_2]$, $a_2 = 4[p\lambda_1^2 + (1-p)\lambda_2^2]$.

Аналогично представим второй сомножитель:

$$\left[q\frac{\mu_1}{\mu_1 + s} + (1-q)\frac{\mu_2}{\mu_2 + s}\right] = \frac{\mu_1\mu_2 + \left[q\mu_1 + (1-q)\mu_2\right]s}{(\mu_1 + s)(\mu_2 + s)} = \frac{b_0 + b_1 s}{(\mu_1 + s)(\mu_2 + s)},$$

где $b_0 = \mu_1\mu_2$, $b_1 = q\mu_1 + (1-q)\mu_2$.

Продолжая разложение, получим:

$$\frac{\psi_+(s)}{\psi_-(s)} = \frac{(a_0 - a_1 s + a_2 s^2)(b_0 + b_1 s) -}{(2\lambda_1 - s)^2 (2\lambda_2 - s)^2 \times} \rightarrow$$

$$\rightarrow \frac{-(2\lambda_1 - s)^2 (2\lambda_2 - s)^2 (\mu_1 + s)(\mu_2 + s)}{\times (\mu_1 + s)(\mu_2 + s)} =$$

$$= \frac{-s(s^5 - c_4 s^4 - c_3 s^3 - c_2 s^2 - c_1 s - c_0)}{(2\lambda_1 - s)^2 (2\lambda_2 - s)^2 (\mu_1 + s)(\mu_2 + s)} =$$

$$= \frac{-s(s + s_1)(s + s_2)(s - s_3)(s - s_4)(s - s_5)}{(2\lambda_1 - s)^2 (2\lambda_2 - s)^2 (\mu_1 + s)(\mu_2 + s)}.$$

Многочлен в числителе в правой части такого разложения, как правило, всегда имеет один нуль $s = 0$ [1]. В данном случае свободный член разложения также равен 0: $a_0 b_0 - 16\lambda_1^2 \lambda_2^2 \mu_1 \mu_2 \equiv 0$.

Окончательно спектральное разложение решения ИУЛ для системы $HE_2/H_2/1$ имеет вид

$$\frac{\psi_+(s)}{\psi_-(s)} = \frac{-s(s + s_1)(s + s_2)(s - s_3)(s - s_4)(s - s_5)}{(2\lambda_1 - s)^2 (2\lambda_2 - s)^2 (\mu_1 + s)(\mu_2 + s)}. \quad (4)$$

Выпишем многочлен пятой степени в числителе разложения

$$s^5 - c_4 s^4 - c_3 s^3 - c_2 s^2 - c_1 s - c_0. \quad (5)$$

Исследование полученного многочлена и нахождение его корней является важной частью метода спектрального разложения решения ИУЛ для любых систем. В нашем случае коэффициенты многочлена (5) имеют следующий вид:

$$c_0 = a_0 b_1 - a_1 b_0 - a_0(\mu_1 + \mu_2) + 16\lambda_1 \lambda_2 \mu_1 \mu_2 (\lambda_1 + \lambda_2);$$

$$c_1 = -a_1 b_1 + a_2 b_0 - a_0 + 16\lambda_1 \lambda_2 (\lambda_1 + \lambda_2)(\mu_1 + \mu_2) -$$
$$- 4\mu_1 \mu_2 [(\lambda_1 + \lambda_2)^2 + 2\lambda_1 \lambda_2];$$

$$c_2 = a_2 b_1 + 4(\lambda_1 + \lambda_2)(4\lambda_1 \lambda_2 + \mu_1 \mu_2) -$$
$$- 4[(\lambda_1 + \lambda_2)^2 + 2\lambda_1 \lambda_2](\mu_1 + \mu_2);$$

$$c_3 = 4(\lambda_1 + \lambda_2)(\mu_1 + \mu_2) - 4(\lambda_1^2 + \lambda_1^2) - 16\lambda_1 \lambda_2 - \mu_1 \mu_2;$$

$$c_4 = 4(\lambda_1 + \lambda_2) - \mu_1 - \mu_2.$$

Эти коэффициенты получены с помощью символьных операций пакета MathCAD, так как изначально в числителе спектрального разложения имеем 42 слагаемых.

Многочлен (5) имеет два действительных отрицательных корня и три положительных корня (либо вместо последних один действительный положительный и два комплексно сопряженных с положительной вещественной частью). Исследование знака младшего коэффициента c_0 показывает, что $c_0 > 0$ всегда в случае стабильной системы, когда $0 < \rho < 1$. С учетом знака минус, стоящего перед c_0 в многочлене (5), это также подтверждает предположение о наличии таких корней многочлена.

Согласно методу спектрального разложения функции $\psi_+(s)$ и $\psi_-(s)$ должны удовлетворять следующим условиям [1]:

1) для $\mathrm{Re}(s) > 0$ функция $\psi_+(s)$ является аналитической без нулей в этой полуплоскости;
2) для $\mathrm{Re}(s) < D$ функция $\psi_-(s)$ является аналитической без нулей в этой полуплоскости, (6)

где D — некоторая положительная константа, определяемая из условия: $\lim\limits_{t\to\infty} \frac{a(t)}{\mathbf{e}^{-Dt}} < \infty$.

Кроме того, функции $\psi_+(s)$ и $\psi_-(s)$ должны обладать следующими свойствами:

$$\lim_{|s|\to\infty,\, \mathrm{Re}(s)>0} \frac{\psi_+(s)}{s} = 1; \quad \lim_{|s|\to\infty,\, \mathrm{Re}(s)<D} \frac{\psi_-(s)}{s} = -1. \quad (7)$$

Теперь с учетом условий (6) и (7) строим рациональные функции $\psi_+(s)$ и $\psi_-(s)$ для рассматриваемого случая:

$$\psi_+(s) = \frac{s(s + s_1)(s + s_2)}{(\mu_1 + s)(\mu_2 + s)},$$

так как нули многочлена (5): $s = 0$, $s = -s_1$, $s = -s_2$ и полюсы $s = -\mu_1$, $s = -\mu_2$ лежат в области $\mathrm{Re}(s) \leqslant 0$;

$$\psi_-(s) = -\frac{(2\lambda_1 - s)^2 (2\lambda_2 - s)^2}{(s - s_3)(s - s_4)(s - s_5)},$$

так как ее нули и полюсы лежат в области $\mathrm{Re}(s) < D$, определенной условием (6). Выполнение условий (6) и (7) спектрального разложения для построенных функций $\psi_+(s)$ и $\psi_-(s)$ также подтверждается рис. 1.

При построении этих функций удобнее нули и полюсы отношения $\psi_+(s)/\psi_-(s)$ отметить на комплексной s-плоскости для исключения ошибок построения функций $\psi_+(s)$ и $\psi_-(s)$.

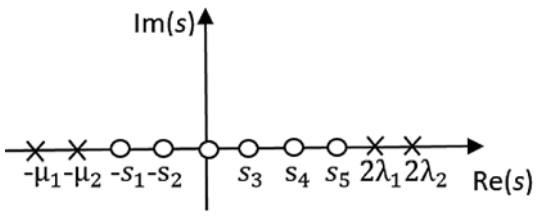


Рис. 1. Нули и полюсы функции $\psi_+(s)/\psi_-(s)$ для системы $HE_2/H_2/1$

На рис. 1 полюсы отмечены крестиками, а нули — кружками.

Далее по методике спектрального разложения найдем константу K:

$$K = \lim_{s\to 0} \frac{\psi_+(s)}{s} = \frac{s_1 s_2}{\mu_1 \mu_2},$$

где s_1, s_2 — абсолютные значения отрицательных корней $-s_1$, $-s_2$. Постоянная K определяет вероятность того, что поступающее в систему требование застает ее свободной.

Для нахождения преобразования Лапласа функции плотности времени ожидания построим функцию

$$\Phi_+(s) = \frac{K}{\psi_+(s)} = \frac{s_1 s_2 (s+\mu_1)(s+\mu_2)}{s\mu_1\mu_2 (s+s_1)(s+s_2)}.$$

Отсюда преобразование Лапласа функции плотности времени ожидания $W^*(s) = s\Phi_+(s)$ будет равно

$$W^*(s) = \frac{s_1 s_2 (s+\mu_1)(s+\mu_2)}{\mu_1\mu_2 (s+s_1)(s+s_2)}. \quad (8)$$

Для нахождения среднего времени ожидания найдем производную от функции (8) $W^*(s)$ со знаком минус в точке $s = 0$:

$$\bar{W} = -\frac{dW^*(s)}{ds}\bigg|_{s=0} = \frac{1}{s_1} + \frac{1}{s_2} - \frac{1}{\mu_1} - \frac{1}{\mu_2}.$$

Окончательно среднее время ожидания для системы $HE_2/H_2/1$ составляет

$$\bar{W} = \frac{1}{s_1} + \frac{1}{s_2} - \frac{1}{\mu_1} - \frac{1}{\mu_2}. \quad (9)$$

Из выражения (8) также можно определить дисперсию времени ожидания. Вторая производная от преобразования (8) в точке $s = 0$ дает второй начальный момент времени ожидания, что позволяет определить дисперсию времени ожидания. Учитывая определение джиттера в телекоммуникациях как разброс времени ожидания от среднего значения [10], тем самым получим возможность его определения через дисперсию. Это является важным результатом для анализа трафика, чувствительного к задержкам.

Для практического применения выражения (9) необходимо определить числовые характеристики распределений (2) HE_2 и (3) H_2. Для этого воспользуемся свойством преобразования Лапласа воспроизведения моментов и запишем начальные моменты до второго порядка для распределения (2):

$$\bar{\tau}_\lambda = \frac{p}{\lambda_1} + \frac{(1-p)}{\lambda_2}; \quad (10)$$

$$\overline{\tau_\lambda^2} = \frac{3}{2}\left[\frac{p}{\lambda_1^2} + \frac{(1-p)}{\lambda_2^2}\right]. \quad (11)$$

Аппроксимация законов распределений HE_2 и H_2 с использованием двух первых моментов

Рассматривая равенства (10) и (11) как запись метода моментов, найдем неизвестные параметры распределения (2) λ_1, λ_2, p. Система двух уравнений (10), (11) при этом является недоопределенной, поэтому к ней добавим выражение для квадрата коэффициента вариации

$$c_\lambda^2 = \frac{\overline{\tau_\lambda^2} - (\bar{\tau}_\lambda)^2}{(\bar{\tau}_\lambda)^2} \quad (12)$$

как связующее условие между (10) и (11). Кроме того, коэффициент вариации будем использовать в расчетах в качестве входного параметра системы. Исходя из вида уравнения (10) положим

$$\lambda_1 = 2p/\bar{\tau}_\lambda;\ \lambda_2 = 2(1-p)/\bar{\tau}_\lambda \quad (13)$$

и потребуем выполнения условия (12). Подставим выражения (10), (11) и частное решение (13) в выражение (12) и получим уравнение четвертой степени $p(1-p)[8(1+c_\lambda^2)p^2 - -8(1+c_\lambda^2)p + 3] = 0$ относительно параметра p. С учетом условия $0 < p < 1$ имеем квадратное уравнение $8(1+c_\lambda^2)p^2 - 8(1+c_\lambda^2)p + 3 = 0$. Решив его, получим

$$p = \frac{1}{2} \pm \sqrt{\frac{2(1+c_\lambda^2) - 3}{8(1+c_\lambda^2)}} \quad (14)$$

и можем выбрать для однозначности, например, наибольшее значение p.

Отсюда следует, что коэффициент вариации интервалов входного потока $c_\lambda \geqslant 1/\sqrt{2}$. Подставив значение параметра p из (14) в соотношение (13), определим значения параметров распределения (2) λ_1, λ_2. Таким образом, получено частное решение недоопределенной системы уравнений (10) и (11) методом подбора.

Аналогично поступим с распределением (3). В этом случае два первых начальных момента будут равны

$$\bar{\tau}_\mu = \frac{q}{\mu_1} + \frac{(1-q)}{\mu_2};\ \overline{\tau_\mu^2} = \frac{2q}{\mu_1^2} + \frac{2(1-q)}{\mu_2^2}.$$

Рассуждая аналогично, положив

$$\mu_1 = 2q/\overline{\tau}_\mu;\ \mu_2 = 2(1-q)/\overline{\tau}_\mu, \qquad (15)$$

для параметра q получим выражение [7]

$$q = \frac{1}{2}\left(1 \pm \sqrt{\frac{c_\mu^2 - 1}{c_\mu^2 + 1}}\right). \qquad (16)$$

Подставив значение параметра q из выражения (16) в соотношение (15), определим значения параметров распределения (3) μ_1, μ_2.

Аппроксимация законов распределений HE_2 и H_2 с использованием трех первых моментов

Учитывая тот факт, что распределения HE_2 и H_2 являются трехпараметрическими, аппроксимацию можно выполнить и на уровне трех первых моментов. Для этого запишем выражения для начального момента 3-го порядка, полученное через преобразование Лапласа $A^*(s)$:

$$\overline{\tau_\lambda^3} = \frac{3p}{\lambda_1^3} + \frac{3(1-p)}{\lambda_2^3}. \qquad (17)$$

Присоединив уравнение (17) к уравнениям моментов (10), (11) и решив систему трех нелинейных уравнений с тремя неизвестными при заданных значениях начальных моментов до третьего порядка включительно, в пакете MathCAD находим все три параметра λ_1, λ_2, p. Аналогично, присоединив уравнение для третьего начального момента распределения (3) $\overline{\tau_\mu^3} = \frac{6q}{\mu_1^3} + \frac{6(1-q)}{\mu_2^3}$ к двум предыдущим уравнениям для первых двух начальных моментов и решив систему трех уравнений в пакете MathCAD, находим все три параметра μ_1, μ_2, q.

Такой подход к аппроксимации законов распределения гиперэкспоненциальным распределением H_2 применен в работах автора [7—9], кроме того, в работе [8] подробно на графике продемонстрирована разница между аппроксимацией на уровне двух первых моментов и на уровне трех первых моментов для распределения H_2. Как показано в работе [7] на примере гиперэкспоненциальных входных распределений, аппроксимация с использованием двух первых моментов в отличие от трех моментов может занижать среднее время ожидания до 10 % в зависимости от значений загрузки и 3-го момента.

Необходимым и достаточным условием существования решения для этой системы трех нелинейных уравнений (следовательно, для аппроксимации распределения H_2 на уровне трех первых моментов) будет выполнение неравенства [11]

$$\overline{\tau_\lambda^3}\,\overline{\tau_\lambda} \geqslant 1{,}5\overline{\tau_\lambda^2}. \qquad (18)$$

Теперь сравним начальные моменты для распределений HE_2 и H_2. Выражения для начальных моментов первого порядка у них совпадают, моментов второго порядка для распределения HE_2 меньше в 4/3 раза, а моментов третьего порядка меньше в 2 раза. С учетом этого факта можно получить условие, аналогичное (18) для аппроксимации распределения HE_2 на уровне трех первых моментов:

$$\overline{\tau_\lambda^3}\,\overline{\tau_\lambda} \geqslant \overline{\tau_\lambda^2}. \qquad (19)$$

Таким образом, гиперэрланговский закон распределения второго порядка, так же как и гиперэкспоненциальный, может однозначно определяться полностью как двумя первыми моментами, так и тремя моментами и перекрывать весь диапазон изменения коэффициента вариации от $1/\sqrt{2}$ до ∞, что шире, чем у гиперэкспоненциального распределения (1, ∞).

Рассмотрим пример аппроксимации входного потока гиперэкспоненциальным распределением (аналогичным будет и вариант с гиперэрланговским законом распределения). Пусть средний интервал между поступлениями $\overline{\tau}_\lambda = 10/9$ (единиц времени), а коэффициент вариации случайной величины — интервала времени между поступлениями — $c_\lambda = 4$. Тогда второй начальный момент $\overline{\tau_\lambda^2} = 17 \cdot (10/9)^2$. Аппроксимация с использованием двух первых моментов дает: $p \approx 0{,}9697$, $\lambda_1 \approx 1{,}7454$, $\lambda_2 \approx 0{,}0546$.

Для аппроксимации с использованием трех первых моментов введем в качестве третьего момента коэффициент асимметрии $A_{S\lambda}$ и для определенности положим $A_{S\lambda} = 7$. Тогда необходимое и достаточное условие аппроксимации (18) будет выполнено. Как известно, для пуассоновского потока параметры $c_\lambda = 1$ и $A_{S\lambda} = 2$. Тогда третий начальный момент будет равен $\overline{\tau_\lambda^3} = 497 \cdot (10/9)^3$ и решение системы трех уравнений метода моментов даст $p \approx 0{,}9111$, $\lambda_1 \approx 6{,}2291$, $\lambda_2 \approx 0{,}0922$. Графики функции плотности H_2 с полученными параметрами приведены на рис. 2.

Величины $\overline{\tau}_\lambda$, $\overline{\tau}_\mu$, c_λ, c_μ будем считать входными параметрами для расчета среднего времени ожидания для системы $HE_2/H_2/1$. Тогда алгоритм расчета сведется:

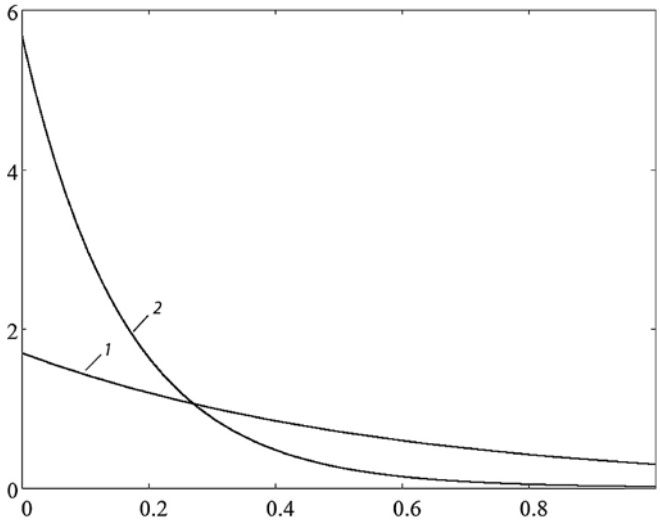


Рис. 2. Графики функции плотности H_2:
1 — аппроксимация закона распределения H_2 на уровне двух моментов; *2* — на уровне трех моментов

- к последовательному определению параметров распределений (2) λ_1, λ_2, p и (3) μ_1, μ_2, q через значения входных параметров $\overline{\tau}_\lambda$, $\overline{\tau}_\mu$, c_λ, c_μ;
- к нахождению нужных корней многочлена (5) $-s_1$, $-s_2$, а затем к использованию расчетного выражения (8).

Результаты экспериментов

В таблице приведены данные расчетов в пакете MathCAD для системы $HE_2/H_2/1$ для случаев малой, средней и высокой нагрузки ρ = 0,1; 0,5; 0,9. Для сравнения в правой колонке приведены данные для системы $H_2/H_2/1$, для которой c_λ, $c_\mu \geqslant 1$ [7]. Коэффициент загрузки ρ в обеих таблицах определяется отношением средних интервалов обслуживания и поступления требований $\rho = \overline{\tau}_\mu/\overline{\tau}_\lambda$. Расчеты, приведенные в таблице, выполнены для нормированного времени обслуживания $\overline{\tau}_\mu = 1$. Заметим, что система $H_2/H_2/1$ применима только при $c_\lambda \geqslant 1$ и $c_\mu \geqslant 1$, поэтому в таблице для случая $c_\lambda < 1$ стоят прочерки.

Результаты экспериментов для СМО $HE_2/H_2/1$ в сравнении с $H_2/H_2/1$

Входные параметры		Среднее время ожидания	
ρ	(c_λ, c_μ)	для системы $HE_2/H_2/1$	для системы $H_2/H_2/1$
0,1	(0,71;1)	0,030	—
	(2,2)	0,335	0,445
	(4,4)	1,666	1,779
	(8,8)	7,10	7,112
0,5	(0,71;1)	0,620	—
	(2,2)	3,974	4,044
	(4,4)	16,392	16,129
	(8,8)	65,967	64,178
0,9	(0,71;1)	6,607	—
	(2,2)	36,271	36,20
	(4,4)	145,465	144,833
	(8,8)	580,822	577,861

Значения среднего времени ожидания в системах $HE_2/H_2/1$ и $H_2/H_2/1$ достаточно близки при средней и высокой нагрузках систем, хотя начальные моменты распределений (начиная со второго) HE_2 и H_2 разнятся. Полученные расчетные данные хорошо согласуются с результатами работы [12] в той области параметров, при которых определена система $HE_2/H_2/1$.

Заключение

В работе получено спектральное разложение решения интегрального уравнения Линдли для системы $HE_2/H_2/1$, с помощью которого выведено расчетное выражение для среднего времени ожидания в очереди для этой системы в замкнутой форме. Результаты расчетов сравниваются с результатами аналогичной системы $H_2/H_2/1$ с гиперэкспоненциальным распределением 2-го порядка для входного потока и времени обслуживания.

Полученное расчетное выражение для среднего времени ожидания расширяет и дополняет известную формулу теории массового обслуживания для среднего времени ожидания для систем типа G/G/1 с произвольными законами распределений входного потока и времени обслуживания. При этом диапазон изменения параметров у системы $HE_2/H_2/1$ шире, чем у системы $H_2/H_2/1$.

Полученный результат с успехом может быть применен в современной теории телетрафика, где задержки пакетов входящего трафика играют первостепенную роль. Для этого необходимо знать числовые характеристики интервалов входящего трафика и времени обслуживания на уровне двух или трех первых моментов, что не вызывает трудностей при использовании современных анализаторов трафика [7].

Заметим, что реальные системы передачи данных имеют ограниченную емкость буфера, поэтому полученные результаты по СМО с бесконечной очередью могут служить лишь первым приближением телетрафика.

Список литературы

1. **Клейнрок Л.** Теория массового обслуживания. Пер. с англ. под редакцией В. И. Неймана. М.: Машиностроение, 1979. 432 с.
2. **Brannstrom N.** A Queueing Theory analysis of wireless radio systems — Appllied to HS-DSCH. Lulea university of technology, 2004. 79 p.
3. **Whitt W.** Approximating a point process by a renewal process: two basic methods // Operation Research. 1982. N. 1. P. 125—147.
4. **Бочаров П. П., Печинкин А. В.** Теория массового обслуживания. М.: Изд-во РУДН, 1995. 529 с.
5. **Алиев Т. И.** Основы моделирования дискретных систем. СПб: СПбГУ ИТМО, 2009. 363 с.
6. **Алиев Т. И.** Аппроксимация вероятностных распределений в моделях массового обслуживания // Научно-технический вестник информационных технологий, механики и оптики. 2013. № 2(84). С. 88—93.
7. **Тарасов В. Н.** Исследование систем массового обслуживания с гиперэкспоненциальными входными распределениями // Проблемы передачи информации. 2016. № 1. С. 16—26.
8. **Тарасов В. Н., Бахарева Н. Ф., Липилина Л. В.** Математическая модель телетрафика на основе системы G/M/1 и результаты вычислительных экспериментов // Информационные технологии. 2016. № 2. С. 121—126.
9. **Тарасов В. Н., Бахарева Н. Ф., Горелов Г. А.** Математическая модель трафика с тяжелохвостным распределением на основе системы массового обслуживания $H_2/M/1$ // Инфокоммуникационные технологии. 2014. № 3. С. 36—41.
10. **URL**: HTTPS://tools.ietf.org/html/rfc3393. RFC 3393 IP Packet Delay Variation Metric for IP Performance Metrics (IPPM) (дата обращения: 26.02.2016).
11. **Myskja A.** An improved heuristic approximation for the GI/GI/1 queue with bursty arrivals // Teletraffic and datatraffic in a Period of Change, ITC-13. Elsevier Science Publishers. 1991. P. 683—688.
12. **Тарасов В. Н., Бахарева Н. Ф.** Обобщенная двумерная диффузионная модель массового обслуживания типа GI/G/1 // Телекоммуникации. 2009. № 7. С. 2—8.

V. N. Tarasov, D. Sc., Professor, Head of Chair, e-mail: veniamin_tarasov@mail.ru,
N. F. Bakhareva, D. Sc., Professor, Head of Chair, e-mail: nadin1956_04@inbox.ru,
Kada Othmane, Postgraduate,
Povolzhsky State University of Telecommunications and Informatics, Samara, 443010, Russian Federation

The Mathematical Model of Teletraffic Based on the $HE_2/H_2/1$ System

The article is devoted to the study of the G/G/1 type $HE_2/H_2/1$ queuing system with a second-order hypererlangian input distribution and a hyperexponential service time law with the aim of obtaining a solution for the average waiting time in queue in the case of stationary mode. For this, the classical method of spectral decomposition of the solution of the Lindley integral equation is used. For practical application of the obtained results, the method of moments is used. It turns out that the hyperelangian distribution law HE_2, like the hyperexponential H_2, which is three-parameter, can be determined by both the first two moments and the first three moments. The choice of such probability distribution laws is due to the fact that they are the most common distributions of non-negative continuous random variables, since the coefficient of variation for the HE_2 $c_\tau \geqslant 1/\sqrt{2}$ distribution covers a wider range than the hyperexponential distribution for which $c_\tau \geqslant 1$. Determination of the principal characteristic of QS G/G/1 — of the average waiting time in queue an important task due to the fact that for such a QS there is no solution in the general case. The method of spectral decomposition of the solution of the Lindley integral equation for the QS $HE_2/H_2/1$ allows one to obtain a solution in closed form.

Keywords: *Hypererlangian and hyperexponential distribution laws, Lindley integral equation, method of spectral decomposition, Laplace transform*

DOI: 10.17587/it.25.531-537

References

1. **Kleinrock L.** Teoriya massovogo obslugivanya, Moscow, Mashinostroenie, 1979, 432 p. (in Russian).
2. **Brannstrom N.** A Queueing Theory analysis of wireless radio systems — Appllied to HS-DSCH, Lulea university of technology, 2004, 79 p.
3. **Whitt W.** Approximating a point process by a renewal process, I: two basic methods, *Operation Research*, 1982, no. 1, pp. 125—147.
4. **Bocharov P. P., Pechinkin A. V.** Teoriya massovogo obsluzhivaniya, Moscow, Publlishing house of RUDN, 1995, 529 p. (in Russian).
5. **Aliev T. I.** *Osnovy modelirovaniya diskretnyh system* (Fundamentals of discrete systems modeling), SPb, Publishing house of SPbGU ITMO, 2009, 363 p. (in Russian).
6. **Aliev T. I.** Approximation of probability distributions in queuing models, *Nauchno-Tekhnicheskij Vestnik Informacionnyh Tekhnologij, Mekhaniki i Optiki,* 2013, no. 2 (84), pp. 88—93 (in Russian).
7. **Tarasov V. N.** Analysis of queues with hyperexponential arrival distributions, *Problemy Peredachi Informacii*, 2016, no. 1, pp. 16—26 (in Russian).
8. **Tarasov V. N., Bakhareva N. F., Lipilina L. V.** Mathematical Model of Teletraffic on the Based G/M/1 System and Results of Computational Experiment, *Informacionnye Tekhnologii*. 2016, no. 2, pp. 121—126 (in Russian).
9. **Tarasov V. N., Bahareva N. F., Gorelov G. A.** Mathematical model of traffic with heavy-tailed distribution based on the queuing system $H_2/M/1$, *Infokommunikacionnye Tekhnologii*, 2014, no. 3, pp. 36—41 (in Russian).
10. **Available** at: Https://tools.ietf.org/html/rfc3393. RFC 3393 IP Packet Delay Variation Metric for IP Performance Metrics (IPPM) (date of access: 26.02.2016).
11. **Myskja A.** An improved heuristic approximation for the GI/GI/1 queue with bursty arrivals, *Tel-etraffic and datatraffic in a Period of Change, ITC-13*, Elsevier Science Publishers, 1991, pp. 683—688.
12. **Tarasov V. N., Bahareva N. F.** A generalized two-dimensional diffusion queuing model of the GI/G/1 type, *Telekommunikacii*, 2009, no. 7, pp. 2—8 (in Russian).

УДК 004.932

DOI: 10.17587/it.25.538-544

И. С. Гречихин, аспирант, ст. преп., e-mail: igrechikhin@hse.ru,
А. В. Савченко, д-р техн. наук, проф., e-mail: avsavchenko@hse.ru,
Национальный исследовательский университет Высшая школа экономики, Нижний Новгород

Метод анализа предпочтений пользователя по фото- и видеоизображениям на мобильном устройстве на основе нейросетевых детекторов объектов на изображениях[1]

Предложен метод извлечения предпочтений пользователей в результате анализа галереи их мобильных устройств. На первом этапе выделяются публичные фото- и видеоизображения, не содержащие лиц из предварительно выделенных кластеров. На втором этапе такие изображения обрабатываются на сервере с помощью высокоточных детекторов объектов. Объекты на остальных (персональных) фото- и видеоизбражениях детектируются непосредственно на устройстве. Представлены экспериментальные результаты сравнительного анализа нескольких предварительно обученных нейросетевых детекторов.

Ключевые слова: *обработка изображений, детектирование объектов, мобильные системы, анализ предпочтений пользователя, кластеризация лиц*

Введение

В настоящее время в связи с одновременным развитием социальных сетей и мобильных устройств [1] наблюдается взрывной рост объема мультимедийных данных, которые создаются пользователями мобильных платформ. При этом такие данные нередко содержат уникальную информацию о пользователе, которая может использоваться, например, для повышения полезности разнообразных рекомендательных систем. Как известно, в последнее время для обработки изображений большинство исследователей и практиков применяют методы, основанные на технологиях глубокого обучения [2]. В контексте задачи анализа предпочтений пользователя по его фотографиям и видеоизображениям наибольший интерес представляют алгоритмы детектирования объектов на изображениях (предметы интерьера, виды еды, транспорт, спортивные принадлежности, музыкальные инструменты и т.п.) [3].

Стоит отметить, что, поскольку указанные пользовательские данные могут содержать персональную информацию, не всегда приемлемой является их передача на удаленный сервер для анализа с помощью современных высокоточных методов [2]. В связи с этим в настоящее время наблюдается заметная тенденция к разработке эффективных архитектур сверточных нейронных сетей (СНС) [3, 4]. В частности, для детектирования объектов на изображениях могут использоваться известные нейросетевые алгоритмы, обеспечивающие баланс между точностью и вычислительной эффективностью [5], такие как SSDLite [6], Faster R-CNN [4], YOLO [7, 8], в которых в качестве базовой СНС используются различные модификации MobileNet [4, 9] и т.п.

К сожалению, точность таких детекторов обычно оказывается намного ниже точности наилучших методов, использующих нейросетевые архитектуры с очень большим числом слоев, такие как ResNet или InceptionResNet [10]. Кроме того, заметим, что далеко не все фотографии и видеоизображения пользователя содержат персональные данные. Например,

[1]Статья подготовлена в ходе проведения исследования (№ 19-04-004) в рамках Программы "Научный фонд Национального исследовательского университета "Высшая школа экономики" (НИУ ВШЭ)" в 2019 г. и в рамках государственной поддержки ведущих университетов Российской Федерации "5-100".

обработка на удаленном сервере вполне приемлема для панорамных снимков достопримечательностей, еды в ресторанах, интерьеров музеев, театров, спортивных сооружений и т.п. Вместе с тем, именно такие изображения содержат наиболее важную информацию о предпочтениях пользователя. Поэтому в настоящей статье предлагается автоматически находить публичные фото- и видеоизображения для их последующей обработки на удаленном сервере с помощью высокоточных детекторов объектов. Предполагается, что персональными являются данные, содержащие лица самого пользователя, его близких друзей и знакомых, выделенных автоматически с помощью известных методов распознавания [11, 12] и кластеризации лиц [13, 14]. При этом объекты во всех остальных данных в галерее можно детектировать с помощью более простых методов непосредственно на мобильном устройстве пользователя. Полученные результаты и сделанные по ним выводы рассчитаны на широкий круг специалистов в области распознавания образов.

1. Анализ предпочтений по изображениям и видеоданным на основе нейросетевых детекторов

Задача анализа предпочтений по фотографиям и видеоданным состоит в том, чтобы по поступившему на вход фотоальбому — множеству фотографий и видеоизображений — выделить наиболее интересные для пользователя категории (виды еды, спортивное оборудование, музыкальные инструменты и т.п.). Предполагается, что для обучения системы для каждой категории задано множество изображений, соответствующих категории объектов, и данные об их местонахождении на изображении (обрамляющие прямоугольники или маска границ). В таком случае результатом анализа предпочтений можно считать частоты встречаемости объектов каждой категории на пользовательских фотографиях и видеоизображениях.

Для детектирования объектов на изображениях и определения их категорий могут использоваться известные высокоточные детекторы, основанные на СНС. В работах [4, 5] предложена архитектура SSDLite и СНС MobileNet v2, которая специально спроектирована для ускорения работы нейронной сети и поэтому удобна для использования в мобильных устройствах. СНС MobileNet извлекает карты признаков входного изображения, используя специальные "разделяемые по глубине" (depthwise-separable) сверточные слои, которые имеют значительно меньшее число параметров и бóльшую скорость обработки данных по сравнению со стандартными сверточными слоями без существенной потери качества. Детектор SSD (Single Shot Detector) использует карту признаков на выходе СНС для предсказания классов и положения объектов за один проход, а его модификация (SSDLite) включает разделяемые по глубине сверточные слои для снижения вычислительной сложности и затрат памяти детектора. В совокупности такая архитектура обнаруживает объекты значительно быстрее, но за счет некоторого уменьшения точности предсказаний.

Faster R-CNN-архитектуры [7] также используют СНС (backbone) для создания карты признаков, но с их помощью определяются несколько (100...200) регионов, в которых могут содержаться потенциально интересные объекты. После этого на основании карты признаков и выделенных регионов предсказывается класс объекта. В качестве СНС в детекторе хорошо зарекомендовали себя архитектуры Inception или InceptionResNet [10], которые считаются одними из самых точных для детектирования объектов на изображениях и их классификации, однако требуют значительных вычислительных ресурсов. Первая СНС (Inception) использует специальные блоки, состоящие из факторизованных, работающих параллельно сверток разного размера, результаты которых соединяются в один слой. СНС состоит из нескольких таких идущих подряд Inception-блоков. InceptionResNet создает более глубокую (и, как следствие, более точную) сеть с помощью добавления к Inception-блокам остаточных (residual) связей.

Таким образом, вычислительная эффективность и сложность по затратам памяти наиболее точных детекторов является недостаточной для их реализации даже на современных мобильных устройствах. При этом использование удаленного сервера для обработки *всех* мультимедийных данных пользователя может оказаться неприемлемым с точки зрения сохранности персональных данных.

2. Предложенный подход

В данной статье предлагается автоматически определять потенциальные публичные изображения на основе известных методов распознавания лиц. Так как в галерее фото- и видео-

файлов отсутствуют идентификаторы (метки) запечатленных на них людей, задача сводится к кластеризации (обучению без учителя). Для ее решения вначале необходимо детектировать лица, например, с помощью описанных в предыдущем разделе методов. Задача группировки состоит в том, чтобы каждому r-му изображению поставить в соответствие одну из $K \geqslant 1$ меток, где общее число различных людей K в общем случае неизвестно. Здесь $r = 1, ..., R$ — номер изображения, а R — общее число обнаруженных в альбоме лиц. Для каждого r-го доступного изображения осуществляется извлечение вектора признаков $\mathbf{x}_r$. В наиболее часто используемых сейчас методах переноса знаний (transfer learning) [2, 15] для предварительного обучения характерных признаков используется внешняя база данных изображений лиц с известными метками классов, с помощью которой происходит обучение глубокой СНС. Далее все изображения лиц приводятся к одному размеру (высота U и ширина V) и подаются на вход СНС [16]. Выходы из $D \gg 1$ значений одного предпоследнего слоя нейронной сети нормируются (в метрике L_2) и формируют вектор признаков $\mathbf{x}_r$ r-го изображения [12]. Для полученных векторов могут использоваться традиционные методы кластеризации, не требующие знания числа кластеров, например иерархическая агломеративная кластеризация [17].

На рис. 1 представлена функциональная схема предлагаемой информационной системы анализа предпочтений пользователей мобильных устройств. На предварительном этапе осуществляется обучение детектированию объектов заданных категорий двух нейросетевых моделей: одной — вычислительно эффективной — для реализации непосредственно на мобильном устройстве и другой — высокоточной — для обработки на удаленном сервере. При этом к обучающему множеству добавляется набор фотографий лиц для их детектирования в дополнение к требуемому списку категорий, характеризующих интересы пользователя.

На первом этапе для обнаружения объектов на всех фотографиях и видеоизображениях непосредственно на мобильном устройстве используют первый детектор, который в дополнение к списку интересов выделяет все лица. Далее с помощью специальной вычислительно эффективной СНС для каждого лица извлекается вектор его характерных признаков и выполняется кластеризация векторов признаков всех лиц. Вначале такая процедура проводится для каждого видеофайла, и векторы признаков центров кластеров лиц, выделенных на каждом видеоизображении, добавляются в общее множество векторов признаков лиц, выделенных на фотографиях, после чего осуществляется итоговая кластеризация и выделяются кластеры с достаточно большим числом лиц. Предполагается, что такие кластеры соответствуют самому пользователю и его друзьям и родственникам, поэтому все содержащие их фото- и видеоизображения объявляются содержащими персональную информацию. Среди остальных данных в галерее пользователь может дополнительно указать их приватность.

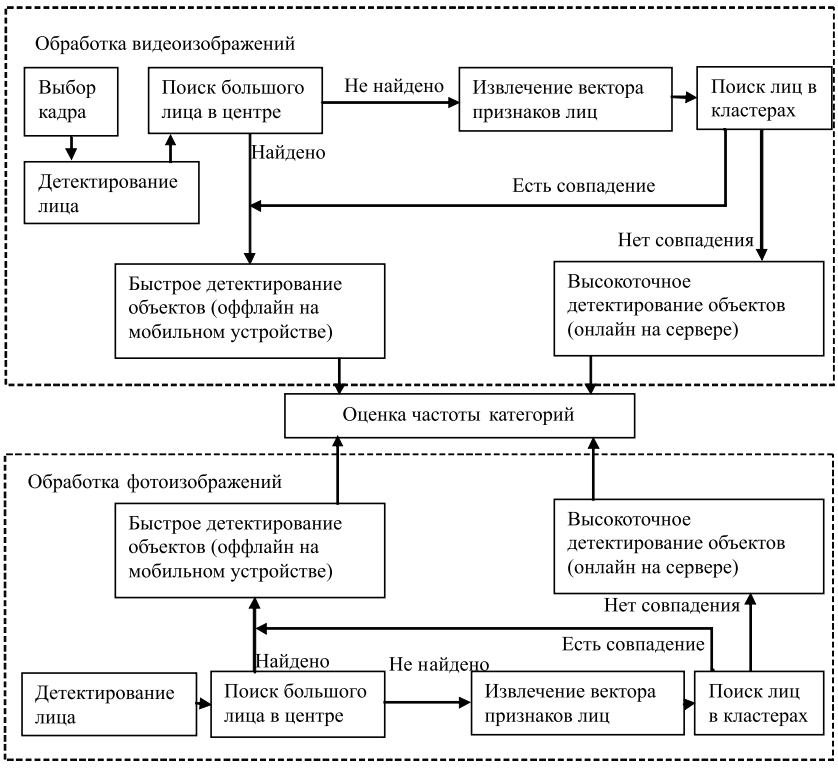


Рис. 1. Схема устройства для анализа предпочтений пользователя

Далее на втором этапе для обработки публичных изображений используется высокоточный нейросетевой детектор, которыей может быть реализован на удаленном сервере. Список обнаруженных объектов возвращается на мобильное устройство и объединяется с результатами первого (эффективного) детектора для подсчета частоты встречаемости каждой категории. Пример экранной формы отображения результатов анализа предпочтений в разработанном Android-приложении приведен на рис. 2.

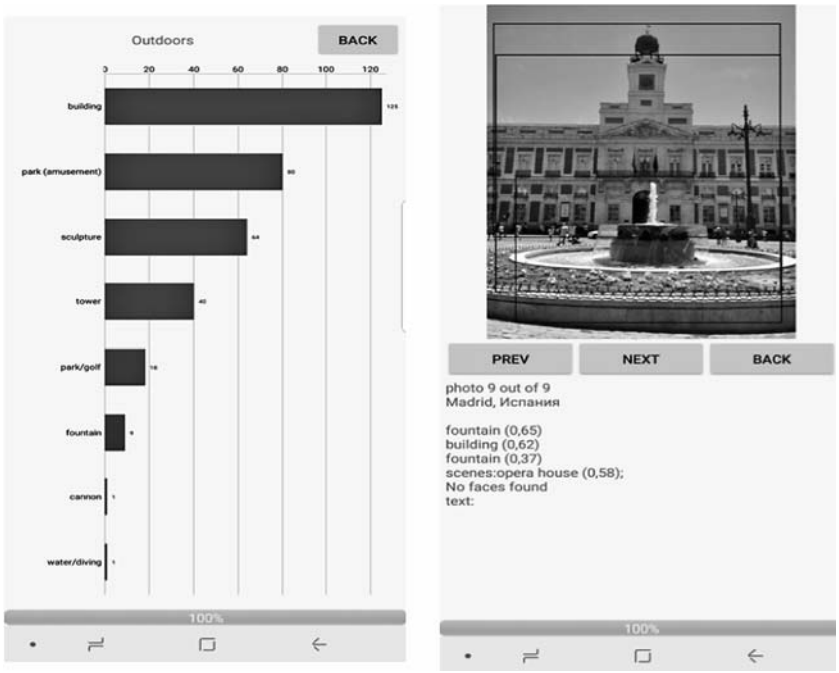


Рис. 2. Экранная форма приложения, реализующего предложенный подход

Таблица 1

Оценки эффективности детекторов категорий

Детектор	Архитектура СНС (backbone)	Размер модели, Мбайт	Время детектирования, с	
			Ноутбук	Смартфон
SSDLite	MobileNet v2, 300 × 300	31,83	0,16	0,30
	MobileNet v2, 512 × 512	31,83	0,21	0,52
Faster R-CNN	Inception v2	64,91	0,4	1,25
	InceptionResNet v2	204,34	1,01	2,39

Таблица 2

Оценки точности и полноты детекторов категорий

Детектор	Архитектура СНС (backbone)	Полнота	mAP	Полнота (родственные категории)	mAP (родственные категории)
Faster R-CNN	InceptionResNet v2	0,425	0,477	0,448	0,534
	InceptionResNet v2 (квантованная)	0,425	0,471	0,448	0,528
	Inception v3	0,393	0,537	0,414	0,593
	ResNet-50	0,332	0,583	0,35	0,636
	ResNet-101	0,465	0,562	0,485	0,618
SSDLite	MobileNet v2, 512 × 512	0,149	0,465	0,166	0,525
	MobileNet v2, 512 × 512 (квантованная)	0,149	0,463	0,166	0,524

3. Вычислительный эксперимент

Для проведения экспериментов и получения детекторов с необходимой точностью создана обучающая выборка из 146 классов (145 категорий интересов пользователя и 1 класс для детектирования лиц) из наборов данных MS COCO, Open Image Dataset и ImageNet. Обучающая выборка была сбалансирована: для каждой категории использовалось не более 5000 изображений. Для обучения применяли библиотеку TensorFlow.

В экспериментах использовали следующие архитектуры нейросетевых детекторов: SSDLite + MobileNet, Faster R-CNN + Inception v2, Faster R-CNN + InceptionResNet v2. Для модели SSDLite исследовались два варианта, соответствующие входным изображениям размера 300 × 300 и 512 × 512 пикселей. Для детекторов Faster R-CNN все изображения масштабировались так, чтобы размер наименьшей стороны был равен 600. Размеры моделей и среднее время предсказаний на ноутбуке (четырехъядерный процессор 4x2.2 ГГц, 16 ГБ ОЗУ) и смартфоне (восьмиядерный процессор: 2x2.2 ГГц, 6x1.6 ГГц, 4 ГБ ОЗУ) представлены в табл. 1.

Как можно увидеть, детекторы SSDLite быстрее и менее затратны по памяти, чем методы на основе Faster R-CNN. Кроме того, модели Faster R-CNN плохо подходят для использования в режиме "оффлайн" на смартфонах из-за времени, затраченного на детектирование.

С использованием тестового набора из других 5000 изображений каждой из 146 категорий оценены показатели полноты (recall, доля верно определенных объектов класса) и точности mAP (mean average precision, доля верных предсказаний). В дополнение составлен список "родственных" категорий, т. е. таких категорий A и B, что детектирование категории A для объекта из категории B нельзя назвать ошибочным (например, категория

Таблица 3

Оценки точности для надежных категорий (полнота более 0,75)

Детектор	Архитектура СНС (backbone)	Число категорий	mAP (родственные категории)
Faster R-CNN	InceptionResNet v2	78	0,662
	InceptionResNet v2 (квантованная)	79	0,663
	Inception v3	44	0,762
	ResNet-50	30	0,838
	ResNet-101	67	0,76
SSDLite	MobileNet v2, 512 × 512	3	0,773
	MobileNet v2, 512 × 512 (квантованная)	3	0,768

"животное" не является ошибкой для объекта "кошка" или "собака", аналогично "строение" — для "небоскреб" или "дом"). Метрики recall и mAP были посчитаны как для исходных категорий, так и с учетом родственных, результаты усреднены по категориям. Результаты эксперимента представлены в табл. 2.

Для каждой модели были отобраны наиболее надежно определяемые категории, значение полноты для которых превышает 0,75. Оценки точности mAP для них приведены в табл. 3.

В табл. 3 можно выделить две архитектуры с лучшими результатами — это Faster R-CNN с СНС InceptionResNet v2 и СНС ResNet-101. Число отобранных категорий характеризует стабильность моделей, т.е. большое число категорий с высоким значением метрик. В среднем у ResNet-101 значения полноты и точности для отобранных категорий выше, чем у InceptionResNet, однако первая архитектура показывает худшие результаты для некоторых важных категорий (лица, строения), которые были включены в отобранные. Например, полнота для категории "небоскреб" у модели ResNet-101 составляет 0,145, однако в среднем ее mAP выше, а число ложноположительных предсказаний меньше. Обе модели показывают низкий mAP (большое число ложноположительных результатов) для категорий "дом", "машина", "животное" и "лицо".

В заключительном эксперименте исследовалось качество кластеризации лиц для набора данных GFW (Grouping Faces in the Wild) [18], содержащего 60 различных фотоальбомов из одной социальной сети. Число лиц в каждом альбоме варьируется в диапазоне от 120 до 3600, при этом альбомы содержат не более $C = 321$ различных людей. Для извлечения признаков лица используются известные СНС: VGGFace (VGGNet-16) [19] и VGGFace2 (ResNet-50) [20] и обученная нами на наборе данных VGGFace-2 СНС MobileNet [14, 21]. Каждая нейронная сеть извлекает вектор признаков лица (1024 для MobileNet, 4096 для VGGNet-16 и 2048 для ResNet-50). Для группировки лиц использовался метод ранговой кластеризации [22], а также иерархическая агломеративная кластеризация со следующими способами определения расстояния между кластерами: single link (одиночная связь), complete link (полная связь), average link (метод невзвешенного попарного среднего), метод взвешенного попарного среднего (в качестве весового коэффициента используется размер кластеров) и медианное расстояние между элементами кластера. Качество кластеризации оценивалось с использованием следующих метрик: отношение числа полученных кластеров K к исходному числу различных людей C, индекс Ранда (Adjusted Rand Index, ARI), индекс взаимной информации (Adjusted Mutual Information, AMI) и бикубическая F-мера (BCubed F-measure). Результаты разных методов кластеризации представлены в табл. 4.

Здесь метод иерархической кластеризации с применением межкластерного расстояния на основе среднего расстояния между точками показывает наилучшие результаты. Ожидаемо, что модель VGGFace2 является несколько точ-

Таблица 4

Результаты кластеризации лиц для набора GFW

Метод кластеризации	СНС	K/C	ARI	AMI	F-мера
Одиночная связь	VGGFace	4,10	0,440	0,419	0,616
	VGGFace2	3,21	0,580	0,544	0,707
	MobileNet	4,19	0,492	0,441	0,636
Метод невзвешенного попарного среднего	VGGFace	1,42	0,565	0,632	0,713
	VGGFace2	1,59	0,603	0,663	0,746
	MobileNet	1,59	0,609	0,658	0,751
Полная связь	VGGFace	0,95	0,376	0,553	0,595
	VGGFace2	1,44	0,392	0,570	0,641
	MobileNet	1,28	0,381	0,564	0,626
Метод взвешенного попарного среднего	VGGFace	1,20	0,464	0,597	0,662
	VGGFace2	1,05	0,536	0,656	0,710
	MobileNet	1,57	0,487	0,612	0,697
Медианное расстояние	VGGFace	5,30	0,309	0,307	0,516
	VGGFace2	4,20	0,412	0,422	0,742
	MobileNet	6,86	0,220	0,222	0,411
Ранговое расстояние	VGGFace	0,82	0,319	0,430	0,630
	VGGFace2	1,53	0,367	0,471	0,641
	MobileNet	1,26	0,379	0,483	0,652

нее остальных, однако СНС MobileNet оказывается в 5...10 раз быстрее и занимает в 2...25 раз меньше памяти по сравнению с остальными моделями. Более того, именно для этой модели с помощью метода невзвешенного попарного среднего (average link) получено наибольшее значение (0,751) F-меры, которое превышает наилучший известный результат (0,74) для этого набора данных [18].

Заключение

Многие задачи построения интеллектуальных мобильных систем зачастую содержат противоречивые требования к реализации высокоточных и одновременно вычислительно эффективных процедур распознавания образов. При этом, как показано в настоящей статье, даже несмотря на наличие некоторых ограничений на обработку персональных данных, часто можно автоматически выделить часть "публичных" изображений, которые для повышения точности системы могут быть отправлены на удаленный сервер. Как показано в проведенном эксперименте, такой подход нередко является наиболее приемлемым за счет использования на сервере наиболее современных нейросетевых моделей, более чем на 10...20 % превосходящих по точности алгоритмы, которые могут быть реализованы на современном мобильном устройстве.

Основным ограничением предлагаемого подхода является использование для выделения публичных изображений только информации о распознанных лицах. В результате многие отсканированные персональные документы могут быть ошибочно отправлены на удаленный сервер. Поэтому потенциальная модификация предложенного метода в будущих исследованиях может состоять в его интеграции с алгоритмами распознавания текста и выявлением текстовых фрагментов, характерных для отсканированных документов.

Список литературы

1. **Harrison G.** Next Generation Databases: NoSQL and Big Data. Berlin, Germany, Springer, 2016. 235 p.

2. **Goodfellow I., Bengio Y., Courville A.** Deep Learning (Adaptive Computation and Machine Learning series). Cambridge, USA, MIT Press, 2016. 800 p.

3. **Kuznetsova A. et al.** The open images dataset V4: Unified image classification, object detection, and visual relationship detection at scale // Cornell University Library, 2018. URL: https://arxiv.org/abs/1811.00982 (date of access 11.02.2019).

4. **Sandler M., Howard A., Zhu M., Zhmoginov A., Chen L. C.** Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation // Cornell University Library, 2018. URL: https://arxiv.org/abs/1801.04381 (date of access 11.02.2019).

5. **Qin Z., Zhang Z., Chen X., Wang C., Peng Y.** Fd-MobileNet: Improved Mobilenet with a fast downsampling strategy // Proceedings of 25th IEEE International Conference on Image Processing (ICIP). 2018. P. 1363—1367.

6. **Huang J. et al.** Speed accuracy trade-offs for modern convolutional object detectors // Cornell University Library. 2016. URL: https://arxiv.org/abs/1611.10012 (date of access 11.02.2019).

7. **Ren S. et al.** Faster R-CNN towards real-time object detection with region proposal networks // Cornell University Library. 2016. URL: https://arxiv.org/abs/1506.01497 (date of access 11.02.2019).

8. **Redmon J., Farhadi A.** YoloV3: An incremental improvement // Cornell University Library. 2018. URL: https://arxiv.org/abs/1804.02767 (date of access 11.02.2019).

9. **Howard A. G.** et al. MobileNets: Efficient convolutional neural networks for mobile vision applications // Cornell University Library. URL: https://arxiv.org/abs/1704.04861 (date of access 11.02.2019).

10. **Szegedy C. et al.** Inception-v4, Inception-ResNet and the impact of residual connections on learning // Proceedings of the International Conference on Artificial Intelligence (AAAI). 2017. Vol. 4. P. 12.

11. **Prince S. J.** Computer vision: Models, learning, and inference. Cambridge, United Kingdom, Cambridge University Press, 2012. 580 p.

12. **Savchenko A. V., Belova N. S**. Unconstrained face identification using maximum likelihood of distances between deep off-the-shelf features // *Expert Systems with Applications,* 2018. Vol. 108. P. 170—182.

13. **Savchenko A. V.** Efficient statistical face recognition using trigonometric series and CNN features // Proceedings of 24th International Conference on Pattern Recognition (ICPR). 2018. P. 3262—3267.

14. **Savchenko A. V.** Efficient facial representations for age, gender and identity recognition in organizing photo albums using multi-output CNN // Cornell University Library. 2018. URL: https://arxiv.org/abs/1807.07718 (date of access 11.02.2019).

15. **Pan S. J.** A survey on transfer learning // IEEE Transactions on Knowledge and Data Engineering. 2010. Vol. 22, N. 10. P. 1345—1359.

16. **Sharif Razavian A., Azizpour H., Sullivan J., Carlsson S.** CNN features off-the-shelf: an astounding baseline for recognition // Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). 2014. P. 806—813.

17. **Sokolova A. D., Kharchevnikova A. S., Savchenko A. V.** Organizing multimedia data in video surveillance systems based on face verification with convolutional neural networks // Proceedings of International Conference on Analysis of Images, Social Networks and Texts (AIST 2017). Cham, Switzerland, Springer. 2017. P. 223—230

18. **He Y., Cao K., Li C., Loy C. C.** Merge or not? Learning to group faces via imitation learning. Cornell University Library, 2018. URL: https://arxiv.org/abs/1707.03986 (date of access 11.02.2019).

19. **Parkhi O. M., Vedaldi A., Zisserman A.** Deep face recognition // Proceedings of the British Conference on Machine Vision (BMVC). 2015. Vol. 1. P. 6.

20. **Cao Q., Shen L., Xie W., Parkhi O. M., Zisserman A.** VGGFace2: A dataset for recognizing faces across pose and age // Proceedings of the International Conference on Automatic Face & Gesture Recognition (FG 2018). 2018. P. 67—74.

21. **Kharchevnikova A. S., Savchenko A. V.** Neural networks in video-based age and gender recognition on mobile platforms // Optical Memory and Neural Networks (Information Optics). 2018. Vol. 27, N. 4. P. 246—259.

22. **Zhu C., Wen F., Sun J.** A rank-order distance based clustering algorithm for face tagging // Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR). 2011. P. 481—488.

I. S. Grechikhin, Postgraduate Student, Senior Lecturer, e-mail: igrechikhin@hse.ru,
A. V. Savchenko, Doctor of Sciences, Professor, e-mail: avsavchenko@hse.ru,
National Research University Higher School of Economics, Nizhny Novgorod

Analysis of User Preferences using Photos and Videos from Mobile Device Based on Object Detection and Neural Networks

In this paper we focus on the problem of user preferences prediction using the gallery of his mobile device. We consider such categories of interests as interior items, food, transport and sport equipment. The novel two-phased method has been proposed. At the first stage, the facial regions are detected on all photos and videos, and the feature vectors are extracted using deep convolutional neural networks. These feature vectors are grouped using known agglomerative clustering techniques. Finally, we select public photos and videos which do not contain faces from the large clusters. At the second stage, these public images are processed on the remote server using high precision Faster R-CNN object detectors. Objects from other images (personal images) are detected on mobile device in offline mode using SSDLite and MobileNet. In the experimental study several neural network-based detectors have been trained using the united training sample from MS Coco, ImageNet and Open Images datasets. Their comparative analysis demonstrated that the Faster R-CNN-based models are characterized with 30 % higher recall when compared to the SSDLite detectors. However, the latter models process each image 3—9-times faster. Finally, we presented the experimental results of facial clustering with GFW (Grouping Faces in the Wild) dataset using either existing feature descriptors (VGGFace, VGGFace2) or the preliminarily trained MobileNet. The latter model with average link hierarchical clustering achieved the highest B-cubed F-measure.

Keywords: *image processing, object detection, mobile systems, visual preferences prediction, face clustering, convolutional neural networks (CNN), Faster R-CNN, SSD*

Acknowledgments. The article was prepared within the framework of the Academic Fund Program at the National Research University Higher School of Economics (HSE University) in 2019 (grant No. 19-04-004) and by the Russian Academic Excellence Project "5-100".

DOI: 10.17587/it.25.538-544

References

1. **Harrison G.** Next Generation Databases: NoSQL and Big Data, Berlin, Germany, Springer, 2016, 235 p.
2. **Goodfellow I., Bengio Y., Courville A.** Deep Learning (Adaptive Computation and Machine Learning series), Cambridge, USA, MIT Press, 2016, 800 p.
3. **Kuznetsova A. et al.** The open images dataset V4: Unified image classification, object detection, and visual relationship detection at scale, Cornell University Library, 2018, available at: https://arxiv.org/abs/1811.00982 (date of access 11.02.2019).
4. **Sandler M., Howard A., Zhu M., Zhmoginov A., Chen L. C.** Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation. Cornell University Library, 2018, available at: https://arxiv.org/abs/1801.04381 (date of access 11.02.2019).
5. **Qin Z., Zhang Z., Chen X., Wang C., Peng Y**. Fd-MobileNet: Improved Mobilenet with a fast downsampling strategy, *Proceedings of 25th IEEE International Conference on Image Processing (ICIP)*, 2018, pp. 1363—1367.
6. **Huang J. et al.** Speed accuracy trade-offs for modern convolutional object detectors. Cornell University Library, 2016, available at: https://arxiv.org/abs/1611.10012 (date of access 11.02.2019).
7. **Ren S. et al.** Faster R-CNN towards real-time object detection with region proposal networks. Cornell University Library, 2016, available at: https://arxiv.org/abs/1506.01497 (date of access 11.02.2019).
8. **Redmon J., Farhadi A.** YoloV3: An incremental improvement. Cornell University Library, 2018, available at: https://arxiv.org/abs/1804.02767 (date of access 11.02.2019).
9. **Howard A. G. et al.** MobileNets: Efficient convolutional neural networks for mobile vision applications. Cornell University Library, available at: https://arxiv.org/abs/1704.04861 (date of access 11.02.2019).
10. **Szegedy C. et al.** Inception-v4, Inception-ResNet and the impact of residual connections on learning, *Proceedings of the International Conference on Artificial Intelligence (AAAI)*, 2017, vol. 4, pp. 12.
11. **Prince S. J.** Computer vision: Models, learning, and inference, Cambridge, United Kingdom, Cambridge University Press, 2012, 580 p.
12. **Savchenko A. V., Belova N. S**. Unconstrained face identification using maximum likelihood of distances between deep off-the-shelf features, *Expert Systems with Applications,* 2018, vol. 108, pp. 170—182.
13. **Savchenko A. V.** Efficient statistical face recognition using trigonometric series and CNN features, *Proceedings of 24th International Conference on Pattern Recognition (ICPR)*, 2018, pp. 3262—3267.
14. **Savchenko A. V.** Efficient facial representations for age, gender and identity recognition in organizing photo albums using multi-output CNN, Cornell University Library, 2018, available at: https://arxiv.org/abs/1807.07718 (date of access 11.02.2019).
15. **Pan S. J.** A survey on transfer learning, *IEEE Transactions on Knowledge and Data Engineering,* 2010, vol. 22, no. 10, pp. 1345—1359.
16. **Sharif Razavian A., Azizpour H., Sullivan J., Carlsson S.** CNN features off-the-shelf: an astounding baseline for recognition, *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2014, pp. 806—813.
17. **Sokolova A. D., Kharchevnikova A. S., Savchenko A. V.** Organizing multimedia data in video surveillance systems based on face verification with convolutional neural networks, *Proceedings of International Conference on Analysis of Images, Social Networks and Texts (AIST 2017),* Cham, Switzerland, Springer, 2017, pp. 223—230.
18. **He Y., Cao K., Li C., Loy C. C.** Merge or not? Learning to group faces via imitation learning, Cornell University Library, 2018, available at: https://arxiv.org/abs/1707.03986 (date of access 11.02.2019).
19. **Parkhi O. M., Vedaldi A., Zisserman A.** Deep face recognition, *Proceedings of the British Conference on Machine Vision (BMVC)*, 2015, vol. 1, pp. 6.
20. **Cao Q., Shen L., Xie W., Parkhi O. M., Zisserman A.** VGGFace2: A dataset for recognizing faces across pose and age, *Proceedings of the International Conference on Automatic Face & Gesture Recognition (FG 2018)*, 2018, pp. 67—74.
21. **Kharchevnikova A. S., Savchenko A. V.** Neural networks in video-based age and gender recognition on mobile platforms. *Optical Memory and Neural Networks (Information Optics),* 2018, vol. 27, no. 4, pp. 246—259.
22. **Zhu C., Wen F., Sun J.** A rank-order distance based clustering algorithm for face tagging, *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR),* 2011, pp. 481—488.

УДК 519.95 DOI: 10.17587/it.25.545-552

Н. А. Игнатьев, д-р физ.-мат. наук, проф., e-mail: n_ignatev@rambler.ru,
Национальный университет Узбекистана им. М. Улугбека

Компактность объектов классов и селекция обучающих выборок

Рассматривается вычисление оценок компактности обучающей выборки и минимальное покрытие ее объектами-эталонами. Для селекции выборок разработан способ удаления шумовых объектов и алгоритм отбора информативных наборов признаков с учетом числа таких объектов. При сравнении различных наборов признаков предложено использовать среднее число объектов выборки без шумовых объектов, притягиваемых одним эталоном минимального покрытия.

Ключевые слова: *шумовые объекты, информативные признаки, селекция обучающей выборки, объекты-эталоны, обобщающая способность, скользящий контроль*

Введение

Гипотеза о компактности является центральным понятием в теории распознавания образов, и единого подхода [1] для оценки компактности не существует. Предложенная в работе [2] мера компактности отдельного класса вычисляется через геометрическую близость входящих в него объектов. Эта мера рассчитана на описание объектов количественными признаками и сильно зависит от масштабов их измерений.

Для вычисления значений компактности необходимо задание меры расстояния между объектами. В качестве такой меры чаще всего выбираются функции со свойствами метрики. Метрики определяют отношения между объектами, которые могут сильно различаться в зависимости от используемых наборов признаков в их описании.

Согласно работе [1] задача выбора информативных признаков состоит в поиске подпространства, в котором характеристика компактности достигает максимального значения. Утверждается, что при выборе набора (подмножества) признаков компактность является критерием информативности более мощным, чем распространенный минимум ошибок при кросс-валидации (скользящем контроле). Для проверки гипотезы о локальной компактности было предложено использовать функцию конкурентного сходства (FRiS-функцию). Число опорных точек или "прецедентов", необходимых для безошибочного распознавания обучающей последовательности, рассматривалось в качестве критерия информативности подсистемы признаков.

Практический интерес характеристики компактности представляют для реализации процесса селекции обучающих выборок [3] в целях повышения обобщающей способности распознающих алгоритмов. Несколько способов вычисления компактности по объектам классов, столпам (объектам-эталонам) выборки, k ближайшим соседям с использованием FRiS-функций описаны в работе [4]. Там же приводится краткий обзор критериев выделения шумовых объектов из выборки. Отмечается, что большая часть этих критериев ориентирована на использование правила ближайшего соседа.

Компактность ассоциируется с непересекающимися областями признакового пространства, каждая из которых представлена группой объектов из одного класса. Так, в методе, описанном в работе [5], конфигурация областей не имеет заранее определенной формы, как правило, гиперсферы или гиперпараллелепипеда. Число групп и их состав не зависят от выбора порядка предъявления объектов для вычисления. Это свойство (устойчивости алгоритма) является следствием использования отношения связанности объектов группы через оболочку (подмножество граничных объектов) класса.

Предобработка данных, реализуемая алгоритмом группировки, гарантирует единственность числа объектов-эталонов минимального покрытия обучающей выборки [5]. Отбор объектов-эталонов проводится по каждой группе отдельно, начиная с самой представительной. Последовательность отбора в группе определяется по упорядоченным значениям отступов до ближайших объектов из противоположных классов.

Цензурирование обучающей выборки путем последовательного удаления шумовых объектов алгоритмом FRiS-Compactor описано в работе [3]. По эвристическим соображениям

устанавливалась доля таких объектов в общей выборке. Доказательство существования зависимости между числом удаляемых шумовых объектов и обобщающей способностью алгоритмов распознавания не приводится.

Наличие ряда закономерностей в данных позволяет уменьшать комбинаторную сложность алгоритмов отбора информативных признаков. Уменьшение сложности происходит за счет исключения просмотра вариантов, не ведущих к оптимальному (согласно выбранному критерию) решению. Для достижения этой цели в работе [6] было предложено использовать предобработку данных путем формирования последовательности признаков, упорядоченных по степени их независимости. Экспериментально было доказано, что значение произведения числа объектов-эталонов локально-оптимального покрытия обучающей выборки на размерность признакового пространства монотонно не возрастает при последовательном удалении малоинформативных признаков из упорядоченной последовательности.

Идея использования меры компактности обучающей выборки со множеством значений в $[a; 1]$, $0 < a < 1$, для отбора информативных латентных признаков описана в работе [7]. Пространство из латентных признаков формируется за счет нелинейного отображения значений исходных признаков в описании объектов на числовую ось с помощью правил иерархической агломеративной группировки. Последовательность объединения исходных признаков на каждом шаге группировки и порядок формирования набора латентных признаков определяются методом динамического программирования. В форме вычислительного эксперимента доказано, что значение меры компактности по первым двум латентным признакам выше, чем на исходном признаковом пространстве. Влияние способа выбора и удаления шумовых объектов на показатели обобщающей способности распознающих алгоритмов и компактности не рассматривалось.

Задача отбора информативных признаков является NP-полной. По этой причине на практике предлагается использовать различные эвристические методы. Типичным недостатком описанных в [8] методов отбора признаков является предположение о том, что признаки независимы.

Реализация многих алгоритмов машинного обучения резко усложняется, когда размерность данных велика. В работе [9] была дана геометрическая интерпретация возникновения эффекта проклятия размерности. Проклятие размерности возникает из-за того, что число возможных наборов признаков гораздо больше, чем число объектов выборки.

Целью данной работы является исследование повышения обобщающей способности алгоритмов распознавания за счет селекции обучающих выборок и снижения размерности признакового пространства. Селекцию обучающих выборок предлагается проводить путем отбора объектов-эталонов минимального покрытия и информативных наборов признаков с учетом удаления шумовых объектов.

Объектом исследования является жадный алгоритм формирования локально-оптимальных наборов признаков. Состав наборов зависит от выбора начального приближения алгоритма. Сравнение точности распознавания проводится по объектам-эталонам минимального покрытия на исходном и информативном наборах признаков.

Предметом исследования является зависимость между точностью распознавания и числом объектов выборки, притягиваемых в среднем одним объектом-эталоном минимального покрытия.

1. О разбиении объектов классов на непересекающиеся группы

Основные идеи приводимого ниже метода изложены в работе [5]. Целями разбиения объектов классов на непересекающиеся группы являются:

- вычисление и анализ значений компактности объектов классов и выборки в целом;
- поиск минимального покрытия обучающей выборки объектами-эталонами.

Рассматривается задача распознавания в стандартной постановке. Считается, что задано множество $E_0 = \{S_1, ..., S_m\}$ объектов, разделенное на $l (l > 2)$ непересекающихся подмножеств (классов) $K_1, ..., K_l$, $E_0 = \bigcup_{i=1}^{l} K_i$. Описание объектов проводится с помощью набора из n разнотипных признаков $X(n) = (x_1, ..., x_n)$, ξ из которых измеряются в интервальных шкалах, $(n - \xi)$ — в номинальной шкале. На множестве объектов E_0 задана метрика $\rho(x, y)$.

Обозначим $L(E_0, \rho)$ — подмножество граничных объектов классов, определяемое на E_0 по метрике $\rho(x, y)$. Объекты $S_i, S_j \in K_t$, $t = 1, ..., l$, считаются связанными между собой $(S_i \leftrightarrow S_j)$, если $\{S \in L(E_0, \rho) | \rho(S, S_i) < r_i$ и $\rho(S, S_j) < r_j\} \neq \varnothing$,

где $r_i(r_j)$ — расстояние до ближайшего от $S_i(S_j)$ объекта из CK_t ($CK_t = E_0 \backslash K_t$) по метрике $\rho(x, y)$.

Множество $G_{tv} = \{S_{v_1}, \ldots, S_{v_c}\}$, $c \geqslant 2$, $G_{tv} \subset K_t$, $v < |K_t|$ представляет область (группу) со связанными объектами в классе K_t, если для любых S_{v_i}, $S_{v_j} \in G_{tv}$ существует путь $S_{v_i} \leftrightarrow S_{v_k} \leftrightarrow \ldots \leftrightarrow S_{v_j}$. Объект $S_i \in K_t$, $t = 1, \ldots, l$, принадлежит группе из одного элемента и считается несвязанным, если не существует пути $S_i \leftrightarrow S_j$ ни для одного объекта $S_j \neq S_i$ и $S_j \in K_t$. Требуется определить минимальное число непересекающихся групп из связанных и несвязанных объектов по каждому классу K_t, $t = 1, \ldots, l$.

Данная задача может рассматриваться и в альтернативной постановке (без задания признаков), если определена квадратная матрица расстояний $\{a_{ij}\}_{m \times m}$ между m объектами и вектор $F = (f_1, \ldots, f_m)$, $f_i \in \{1, \ldots, l\}$ принадлежности объектов к классам $K_1, \ldots, K_l$. Вектор F служит дополнительной информацией для задания условий группировки. Пример использования дополнительной информация в виде частично обученной выборки (ЧОВ) имеется в работе [10]. В выборке указывалось подмножество пар объектов, которые при разбиении не должны попадать в одну группу.

При определении минимального числа групп из связанных и несвязанных объектов классов используется $L(E_0, \rho)$ — подмножество граничных объектов (оболочка) классов по заданной метрике ρ и описание объектов в новом пространстве из бинарных признаков. Для выделения оболочки классов для каждого $S_i \in K_t$, $t = 1, \ldots, l$, строится упорядоченная по $\rho(x, y)$ последовательность

$$S_{i_0}, S_{i_1}, \ldots, S_{i_{m-1}}, S_i = S_{i_0}. \qquad (1)$$

Пусть $S_{i_\beta} \in CK_t$ — ближайший к S_i объект из (1), не входящий в класс K_t. Обозначим $O(S_i)$ окрестность радиуса $r_i = \rho(S_i, S_{i_\beta})$ с центром в S_i, включающую все объекты, для которых $\rho(S_i, S_{i_\tau}) < r_i$, $\tau = 1, \ldots, \beta - 1$. В $O(S_i)$ всегда существует непустое подмножество объектов

$$\Delta_i = \left\{ \begin{matrix} S_{i_\alpha} \in O(S_i) \mid \rho\left(S_{i_\beta}, S_{i_\alpha}\right) = \\ = \min\limits_{S_{i_\tau} \in O(S_i)} \rho\left(S_{i_\beta}, S_{i_\tau}\right) \end{matrix} \right\}. \qquad (2)$$

По множеству (2) принадлежность объектов к оболочке классов определяется как $L(E_0, \rho) = \bigcup\limits_{i=1}^{m} \Delta_i$.

Множество объектов оболочки из $K_t \cap L(E_0, \rho)$ обозначим как $L_t(E_0, \rho) = \{S^1, \ldots, S^\pi\}$, $\pi \geqslant 1$. Значение $\pi = 1$ однозначно определяет вхождение всех объектов класса в одну группу. При $\pi \geqslant 2$ преобразуем описание каждого объекта $S_i \in K_t$ в $S_i = (y_{i1}, \ldots, y_{i\pi})$, где

$$y_{ij} = \begin{cases} 1, \rho(S_i, S^j) < r_i; \\ 0, \rho(S_i, S^j) \geqslant r_i. \end{cases} \qquad (3)$$

Пусть по соотношению (3) получено описание объектов класса K_t в новом (бинарном) признаковом пространстве, $\Omega = K_t$, θ — число непересекающихся между собой групп объектов, $S_\mu \vee S_\eta$, $S_\mu \wedge S_\eta$ — соответственно операции дизъюнкции и конъюнкции по бинарным признакам объектов S_μ, $S_\eta \in K_t$. Пошаговое выполнение алгоритма разбиения объектов K_t на непересекающиеся группы $G_1, \ldots, G_\theta$ таково:

Шаг 1. $\theta = 0$.

Шаг 2. Выделить объект $S \in \Omega$, $\theta = \theta + 1$, $Z = S$, $G_\theta = \varnothing$.

Шаг 3. Выполнять Выбор $S \in \Omega$ и $S \wedge Z = true$, $\Omega = \Omega \backslash S$, $G_\theta = G_\theta \cup S$, $Z = Z \vee S$ **пока** $\{S \in \Omega | S \wedge Z = true\} \neq \varnothing$.

Шаг 4. Если $\Omega \neq \varnothing$, то переход к шагу 2.

Шаг 5. Конец.

2. О селекции обучающей выборки и ее компактности

Одним из способов повышения обобщающей способности распознающих алгоритмов является селекция обучающих выборок через поиск и удаление шумовых объектов. В данной работе множество шумовых объектов рассматривается как подмножество граничных объектов классов по заданной метрике. Множество граничных объектов $B \subset E_0$ определяется как

$$B = \left\{ S \in E_0 \left| \rho(S_i, S) = \min_{S_i \in K_j, S_d \in CK_j} \rho(S_i, S_d) \right. \right\}.$$

Объект $S \in B \cap K_j$, $j = 1, \ldots, l$, принадлежит множеству шумовых объектов D_j класса K_j, если

$$\left| \left\{ S_i \in E_0 \left| \rho(S_i, S) = \min_{S_i \in CK_j, S_d \in K_j} \rho(S_i, S_d) \right. \right\} \right| > \left| \left\{ S_i \in K_j \left| \rho(S_i, S) < \min_{S_i \in K_j, S_d \in CK_j} \rho(S_i, S_d) \right. \right\} \right|. \qquad (4)$$

Для проверки условия (4) нужно определить:

- число объектов из CK_j, для которых $S \in B \cap K_j$ является ближайшим по метрике $\rho(x, y)$;
- число выполненных неравенств вида $\rho(S_i, S) < \min\limits_{S_i \in K_j, S_d \in CK_j} \rho(S_i, S_d)$ для объектов $S_i \in K_j$.

Поиск шумовых объектов по условию (4) имеет смысл, если число граничных объектов $|B|$ больше числа классов l. Считается, что обобщающая способность алгоритма повышается, если дать ему возможность ошибаться на определяемых объектах выборки. В нашем случае в качестве таковых рассматриваются объекты из $\bigcup\limits_{i=1}^{l} D_i$.

Пусть представители класса $K_i \cap \left(E_0 \setminus \bigcup\limits_{j=1}^{l} D_j \right)$, $i = 1, ..., l$, разделены на минимальное число μ непересекающихся групп объектов $G_{i1}, ..., G_{i\mu}$ по алгоритму из п. 1, $m_{ij} = |G_{ij}|$, $j = 1, ..., \mu$, $\sum\limits_{j=1}^{\mu} m_{ij} = m_i$. Для анализа результатов разбиения класса K_i на непересекающиеся группы с учетом их числа, представительности (по числу объектов) и удаления шумовых объектов предлагается использовать такую структурную характеристику, как оценка компактности

$$\Theta_i = \frac{\sum\limits_{j=1}^{\mu} m_{ij}^2}{m_i^2}. \qquad (5)$$

Очевидно, что множество допустимых значений Θ_i по выражению (5) лежит в интервале $\left[\frac{1}{m_i}, 1 \right]$. Если группа G_{i1} содержит все объекты из $K_i \cap \left(E_0 \setminus \bigcup\limits_{j=1}^{l} D_j \right)$, то $\Theta_i = 1$. Усредненная оценка компактности обучающей выборки в целом проводится с учетом доли $\left(\frac{\left| E_0 \setminus \bigcup\limits_{i=1}^{l} D_i \right|}{m} \right)$ исключенных из рассмотрения по условию (4) шумовых объектов как

$$R\left(E_0 \setminus \bigcup_{i=1}^{l} D_i, \rho \right) = \left(\frac{\left| E_0 \setminus \bigcup\limits_{i=1}^{l} D_i \right|}{m} \right) \frac{\sum\limits_{i=1}^{l} m_i \Theta_i}{\left| E_0 \setminus \bigcup\limits_{i=1}^{l} D_i \right|} = \frac{\sum\limits_{i=1}^{l} m_i \Theta_i}{m}. \qquad (6)$$

Значения (5) и (6) косвенно свидетельствуют об однородности (неоднородности) структуры обучающей выборки. Чем ближе сходство групп по числу входящих в них объектов класса, тем ближе значение (5) к $\frac{1}{m_i}$, а (6) — к $\frac{l}{m}$. При $R\left(E_0 \setminus \bigcup\limits_{i=1}^{l} D_i, \rho \right) = 1$ число групп объектов на $E_0 \setminus \bigcup\limits_{j=1}^{l} D_j$ равно числу классов. Множество значений по (5) и (6) соответственно в $\left[\frac{1}{m_i}, 1 \right]$ и $\left[\frac{l}{m}, 1 \right]$ предлагается использовать в качестве меры компактности классов и выборки в целом.

Свойство связанности объектов классов по системе гипершаров, в пересечении которых содержатся объекты оболочек, позволяет получить разбиение выборки на минимальное число непересекающихся групп по алгоритму из п. 1. Такое разбиение в работе [5] предложено использовать для поиска минимального покрытия выборки объектами-эталонами по последовательности локальных областей признакового пространства, представленных объектами групп.

Обозначим $R_S = \rho(\overline{S}, S)$ расстояние от объекта $S \in K_t$ до ближайшего объекта $\overline{S}$ из противоположного к K_t класса ($\overline{S} \in CK_t$); δ — минимальное число непересекающихся групп из связанных и несвязанных объектов классов на E_0.

Последовательность реализации процедуры поиска минимального покрытия объектами-эталонами обучающей выборки в работе [5] следующая. Упорядочим объекты каждой группы $G_u \cap K_t$, $u = 1, ..., \delta$, $t = 1, ..., l$, по множеству значений $\{R_S\}_{S \in G_u}$. В качестве меры расстояния между $S \in G_u$, $u = 1, ..., \delta$, и произвольным допустимым объектом S' используется локальная метрика $d(S, S') = \rho(S, S')/R_S$. Решение о принадлежности S' к одному из классов $K_1, ..., K_l$ или отказе от распознавания принимается по правилу: $S' \in K_t$, если

$$d(S_\mu, S') = \min_{S_j \in E_0} d(S_j, S') \text{ и } S_\mu \in K_t \text{ и } d(S_m, S') \neq \min_{S_j \in CK_t} d(S_j, S'). \qquad (7)$$

Согласно принципу *последовательное исключение,* используемому в процессе поиска покрытия, выборка E_0 делится на два подмножества: множество эталонов E_{ed} и контрольное множество E_k, $E_0 = E_{ed} \cup E_k$. В начале процесса $E_{ed} = E_0$, $E_k = \varnothing$. Упорядочение по значениям отступа $\{R_S\}_{S \in G_u}$, $u = 1, ..., \delta$, используется для определения кандидата на удаление из числа

объектов-эталонов по группе G_u. Идея отбора заключается в поиске минимального числа эталонов, при котором алгоритм распознавания по правилу (7) остается корректным (без ошибок распознающим объекты) на E_0.

Будем считать, что нумерация групп объектов отражает порядок $|G_1| \geqslant ... \geqslant |G_\delta|$, и по группе G_p, $p = 1, ..., \delta$, не проводился отбор объектов-эталонов. Кандидаты на удаление из E_{ed} последовательно выбираются начиная с $S \in G_p$, с минимальным значением R_S. Если включение $S \in E_k$ нарушает корректность решающего правила (7), то S возвращается в множество E_{ed}.

3. Отбор информативных наборов признаков

В работе [1] утверждалось, что при отборе информативных признаков с учетом компактности происходит невозрастание числа объектов-эталонов (столпов), необходимых для корректного распознавания объектов обучающей выборки. Для доказательства утверждения требовалось показать увеличение значения локальной компактности при удалении малоинформативных признаков, выражаемой через число объектов, притягиваемых столпами.

Очевидно, что число и состав шумовых объектов, определяемых условием (4), зависит от наборов признаков в описании объектов. Существует проблема согласования процессов отбора информативных признаков и удаления шумовых объектов. Для решения этой проблемы предлагается использовать алгоритм формирования информативных наборов признаков, состав которых зависит от выбора начальных приближений.

Пусть структура объектов классов на выборке E_0 вычисляется по алгоритму группировки из п. 1, минимальное покрытие объектами-эталонами с учетом удаления шумовых объектов получено по принципу *последовательное исключение* из п. 2. Обозначим $Sh(X(k))$ — число шумовых объектов E_0, определяемых по условию (4) на наборе признаков $X(k) \subset X(n)$, CF — число объектов-эталонов минимального покрытия обучающей выборки, из которой удалены $Sh(X(k))$ шумовых объектов.

Идея поиска информативного набора $X(k) \subset X(n)$ заключается в отказе от перебора тех сочетаний признаков из $X(n)$, которые не ведут к оптимальному или локально-оптимальному решению. Обозначим

$$F(X(k)) = \left(\frac{m - Sh(X(k))}{m}\right)\left(\frac{m - Sh(X(k))}{CF}\right) \quad (8)$$

— мультипликативный показатель притягательной способности CF объектов-эталонов минимального покрытия с учетом числа $Sh(X(k))$ удаляемых из выборки шумовых объектов. Очевидно, что экстремальное значение (8) равно $\frac{m}{l}$, которое достигается в случае отсутствия шумовых объектов и при одном эталоне на каждом из l классов.

Рассмотрим задачу поиска информативного набора $X(k) \subset X(n)$, $1 \leqslant k \leqslant n$, не прибегая к явному определению числа объектов-эталонов минимального покрытия CF. Обозначим P — подмножество индексов признаков из $X(n)$, $D_j(P)$ — множество шумовых объектов класса K_j, $j = 1, ..., l$, по (4) на наборе $\{x_a\}_{a \in P}$.

Считается, что состав множества объектов в гипершаре

$$O(S_i, P) = \left\{ S \in K_j \left| \begin{array}{l} \rho\left(S, \overline{S_i}\right) < \rho\left(S_i, \overline{S_i}\right), \rho\left(S_i, \overline{S_i}\right) = \\ = \min\limits_{S \in CK_i \cap \left(E_0 \setminus \bigcup\limits_{t=1}^{l} D_t(P)\right)} \rho\left(S_i, S\right) \end{array} \right. \right\}$$

с центром в $S_i \in E_0 \cap K_j$, $j = 1, ..., l$, $\overline{S_i} \notin K_j$, и радиусом $\rho\left(S_i, \overline{S_i}\right)$ определяется по набору признаков $\{x_a\}_{a \in P}$. Оценку объекта $S_i \in E_0$ на $\{x_a\}_{a \in P}$ вычислим как

$$Z\left(S_i, P\right) = \begin{cases} 0, S_i \in \bigcup\limits_{j=1}^{l} D_j\left(P\right); \\ \left|O\left(S_i, P\right)\right|. \end{cases} \quad (9)$$

Признак $x_d \in X(n)$ является кандидатом на включение в набор $\{x_a\}_{a \in P}$, если

$$\sum_{S_i \in E_0} Z\left(S_i, P \cup \{d\}\right) > \sum_{S_i \in E_0} Z\left(S_i, P\right).$$

Для отбора информативных наборов признаков предлагается алгоритм, описание пошаговой реализации которого приводится ниже.

Шаг 1. Выбор $i1, j1 \in \{1, ..., n\}$, $i1 \neq j1$. $P = \{i1, j1\}$. $MAXP = 0$;

Шаг 2. Выделить $\bigcup\limits_{j=1}^{l} D_j\left(P\right)$ по (4) на $\left\{x_a\right\}_{a \in P}$.
Вычислить $\left\{Z\left(S_i, P\right)\right\}_1^m$ по (9) на $\{x_a\}_{a \in P}$.
$MAXP = \sum\limits_{i=1}^{m} Z\left(S_i, P\right)$;

Шаг 3. $u = 0$. $C = 0$.
Для всех $v \in \{1, ..., n\} \setminus P$
выделить $\bigcup\limits_{j=1}^{l} D_j(P \cup \{v\})$ по (4)
на $\left\{x_a\right\}_{a \in P \cup \{v\}}$.

Вычислить $\left\{Z\left(S_i, P \cup \{v\}\right)\right\}_1^m$ по (9)

на $\{x_a\}_{a \in P \cup \{v\}}$, $N = \sum_{i=1}^{m} Z\left(S_i, P \cup \{v\}\right)$.

Если $N > C$, то $C = N$, $u = v$.

Шаг 4. Если $C > MAXP$, то $MAXP = C$, $P = P \cup \{u\}$, переход к шагу 2.

Шаг 5. Вывод P.

Шаг 6. Конец.

Состав наборов признаков и их число, определяемое алгоритмом, зависят от выбора значений множества P на первом шаге (индексов признаков $i1$, $j1$). Для выбора единственного решения из нескольких наборов предлагается использовать показатели обобщающей способности распознающих алгоритмов.

4. Вычислительный эксперимент

Методика отбора информативных признаков с учетом удаления шумовых объектов по условию (4) демонстрируется на четырех выборках данных из работы [11]. При вычислении множества граничных объектов классов были использованы метрика Евклида и метрика Журавлева

$$\rho(x, y) = \sum_{i \in I} |x_i - y_i| + \sum_{i \in J} \begin{cases} 1, x_i \neq y_i; \\ 0, x_i = y_i, \end{cases}$$

где I, J — множества индексов соответственно количественных и номинальных признаков. Для унификации масштабов измерений значения количественных признаков были отображены в [0; 1].

Метрики Евклида и Журавлева рассматривали как базовые для формирования локальных метрик каждого объекта выборки. При определении локальной метрики объекта использовали расстояние до ближайшего (граничного) объекта из противоположного класса по базовой метрике. Шумовые объекты в условии (4) рассматриваются как подмножество граничных объектов классов. Удаление шумовых объектов приводит к изменению состава граничных объектов. Следствием из этого является изменение расстояний между объектами, вычисляемыми по локальным метрикам. В табл. 1 представлены оценки (8) компактности минимального покрытия выборок данных из работы [9] объектами-эталонами по заданной метрике при удалении шумовых объектов по условию (4).

Таблица 1

Оценки компактности минимального покрытия объектов по заданной метрике

Выборка данных	Базовая метрика	Число объектов		Значение (8)
		шумовых	эталонов	
Liver	Журавлева	183	51	5,0817
German	Журавлева	260	110	4,9782
Australian	Евклида	117	32	14,8700
Ionosphere	Евклида	23	16	19,1567

Единственность множества граничных объектов классов следует из структуры расстояний между объектами обучающей выборки, определяемой по заданной базовой метрике. Следствием из этого является единственность решения выбора числа и состава шумовых объектов по условию (4) и отбора эталонных объектов по принципу *последовательное исключение* из п. 2. Интегрирующим показателем между числом шумовых и эталонных объектов в табл. 1 является среднее число объектов (8), притягиваемых одним эталоном минимального покрытия выборки.

Решение по отбору информативных признаков алгоритмом из п. 3 не является устойчивым, так как оно зависит от выбора начального приближения. Какому из нескольких наборов отдать предпочтение, может быть определено по показателям обобщающей способности распознающих алгоритмов.

Обобщающая способность распознающих алгоритмов по правилу (7) на данных из работы [9] показана в табл. 2 на исходных и на некоторых из наборов информативных признаков. При вычислении показателей обобщающей

Таблица 2

Обобщающая способность алгоритмов с учетом удаления шумовых объектов

Характеристики	Выборка данных			
	Australian	Ionosphere	Liver	German
Информативный набор признаков	$x_2, x_3, x_5, x_8, x_{10}, x_{13}, x_{14}$	x_1, x_5, x_7, x_8	x_2, x_3, x_4, x_5, x_7	$x_1, x_2, x_3, x_4, x_5, x_8, x_{14}$
Точность на исходном наборе признаков	81,81	80,33	70,51	72,25
Точность на информативном наборе признаков	84,49	85,52	69,32	73,55

способности применяли 60 различных вариантов разделения объектов выборок на обучение и контроль в соотношении 9:1. Для удаления шумовых объектов использовали условие (4).

Набор информативных признаков, полученный алгоритмом из п. 3, не всегда является гарантией повышения обобщающей способности алгоритма распознавания на нем относительно исходного набора. Это видно на примере данных Liver из табл. 2. На исходном наборе обобщающая способность была 70,51, на информативном — 69,32. Изменение показателей компактности (8) на информативных наборах признаков из табл. 2 демонстрирует табл. 3.

Из сравнения результатов вычислительного эксперимента в табл. 1 и табл. 3 можно сделать следующие выводы. Существует различие между числом шумовых объектов и объектов-эталонов на исходном и информативном наборах признаков. Устойчивые изменения прослеживаются по показателю компактности (8). На информативных наборах признаков значения компактности выше, чем на исходных. Такое отношение ("выше") объясняется чувствительностью метрических алгоритмов к размерности пространства. Существуют наборы, добавление новых признаков в которые приводит к размыванию отношений сходства между объектами. Индикатором изменения отношений сходства служат значения показателей (8).

Использование оценок компактности позволяет ранжировать обучающие выборки в следующем порядке:

1) значение (8), равное $\frac{m}{l}$, определяет выборку, в которой нет шумовых объектов по условию (4) и каждый класс представлен одним объектом-эталоном минимального покрытия;

2) множество шумовых объектов, формируемое по условию (4), пусто, а компактность выборки по выражению (6) равна 1. Число объектов-эталонов минимального покрытия выборки больше числа классов;

Таблица 3

Оценки компактности минимального покрытия объектов на информативных наборах признаков из табл. 2

Выборка данных	Базовая метрика	Число объектов		Значение (8)
		шумовых	эталонов	
Liver	Журавлева	170	39	7,1041
German	Журавлева	284	88	5,8256
Australian	Евклида	120	24	19,6196
Ionosphere	Евклида	32	9	32,2130

Таблица 4

Различия между алгоритмами

FRiS-Stolp	Новый алгоритм
Наличие шумовых объектов и их число определяются в процессе вычисления объектов-эталонов (столпов) выборки	Шумовые объекты выбираются из множества граничных объектов классов. Единственность выбора определяется по условию (4)
Выбор объекта $S \in E_0$ в качестве эталона проводится по критерию эффективности $R(S) = \lambda D_s + (1 - \lambda)T_s$, где D_s, T_s — характеристики соответственно обороноспособности и толерантности, параметр $\lambda \in [0,1]$	Выбор объектов-эталонов проводится на основе результатов разбиения объектов классов на непересекающиеся группы. Число и состав множества эталонов для корректного распознавания выборки определяются по каждой группе в отдельности

3) множество шумовых объектов выборки, формируемое по условию (4), не пусто.

Изменение ранга обучающей выборки с 3 на 2 и с 2 на 1 может быть получено путем отбора информативных наборов признаков алгоритмом из п. 3.

Использованный для вычислительного эксперимента алгоритм распознавания является новым, и есть причины для сравнения его с известным алгоритмом построения решающего правила FRiS-Stolp [1,3]. Вычисление локальных метрик в алгоритмах проводится по одному принципу. Различия между алгоритмами представлены в табл. 4.

Как видно из табл. 4, для реализации нового алгоритма не требуется задавать дополнительные параметры. Процедуры отбора шумовых объектов и минимального покрытия выборки объектами-эталонами не пересекаются друг с другом. Сравнительный анализ двух алгоритмов (табл. 4) по показателю обобщающей способности является темой дальнейших исследований.

Заключение

Предложен способ выделения шумовых объектов как подмножества граничных объектов классов обучающей выборки по заданной базовой метрике. Описаны алгоритмы поиска минимального покрытия выборки объектами-эталонами и отбора информативных наборов признаков с учетом удаления шумовых объектов. Для селекции предложено использовать показатели компактности обучающих выборок и среднее число объектов, притягиваемых одним эталоном минимального покрытия выборки.

Список литературы

1. **Загоруйко Н. Г., Кутненко О. А., Зырянов А. О., Леванов Д. А.** Обучение распознаванию образов без переобучения // Машинное обучение и анализ данных. 2014. Т. 1, № 7. С. 891—901.
2. **Субботин С. А.** Комплекс характеристик и критериев сравнения обучающих выборок для решения задач диагностики и распознавания образов // Математичні машини і системи. 2010. № 1. С. 25—39.
3. **Загоруйко Н. Г., Кутненко О. А.** Цензурирование обучающей выборки // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. 2013. № 1(22). С. 66—73.
4. **Борисова И. А., Кутненко О. А.** Цензурирование ошибочно классифицированных объектов выборки // 17-я Всероссийская конференция "Математические методы распознавания образов — 2015". 19—25 сентября 2015, Россия, г. Светлогорск.
5. **Игнатьев Н. А.** Кластерный анализ данных и выбор объектов-эталонов в задачах распознавания с учителем // Вычислительные технологии. 2015. Т. 20, № 6. С. 34—43.
6. **Згуральская Е. Н.** Выбор информативных признаков для решения задач классификации с помощью искусственных нейронных сетей // Нейрокомпьютеры: разработка, применение. 2012. № 2. С. 20—27.
7. **Saidov D. Y.** Data visualization and its proof by compactness criterion of objects of classes //International Journal of Intelligent Systems and Applications (IJISA), 2017. Hang Kong, Vol 9, N. 8. P. 51—58.
8. **Jeffery I., Higgins D., Culhane A.** Comparison and evaluation of methods for generating differentially expressed gene lists from microarray data // BMC Bioinformatics, 2006. Vol. 7, P. 359.
9. **Гудфеллоу Я., Бенджио И., Курвилль А.** Глубокое обучение. Пер. с англ. Москва: ДМК Пресс, 2018. 652 с.
10. **Айвазян С. А., Бухштабер В. М., Енюков И. С., Мешалкин Л. Д.** Прикладная статистика. Классификация и снижение размерности. М.: Финансы и статистика, 1989. 608 с.
11. http://archive.ics.uci.edu/ml/datasets.

N. A. Ignatiev, D. Sc., Professor, e-mail: n_ignatev@rambler.ru,
National University of Uzbekistan named after M. Ulugbek, Tashkent

Compactness of Objects of Classes and Selection of Learning Samples

The relations between the sample objects $E_0 = \{S_1, ..., S_m\}$, divided into disjoint classes $K_1, ..., K_l$, is considered. Compactness determines the quantitative measure of these relations by a given metric $\rho(x, y)$ and various subsets of the set of features X(n). The selection of the sample is based on the results of the sequential execution of three operators:

— search and delete noise objects;

— minimum sample coverage with objects—standards;

— selection informative set of features.

The rule according to which noise objects are considered as a subset of boundary objects of classes is described. A grouping algorithm based on the relationship "↔" of connectedness objects of classes has been developed. For any pair (S_i, S_j), $S_i, S_j \in K_d$, $d = 1, ..., l$, included in one group, there always exists a chain $S_i \forall S_k \forall ... \forall S_j$. The search for the minimum sample coverage with reference objects is performed for each group separately. It is shown that when selecting informative features, the average number of objects attracted by one of the standard increases.

Keywords: *noise object, informative features, selection of training samples, object—standard, generalizing ability, cross validation*

DOI: 10.17587/it.25.545-552

References

1. **Zagoruiko N. G., Kutnenko O. A., Zyryanov A. O., Levanov D. A.** *Mashinnoe Obychenie i Analiz Dannih*, 2014, vol. 1, no. 7, pp. 891—901 (in Russian).
2. **Subbotin S. A.** *Matematichni Mashini i Sistemi*, 2010, no. 1, pp. 25—39 (in Russian).
3. **Zagoruiko N. G., Kutnenko O. A.** *Vestnik Tomskogo Gosudarstvennogo Universiteta. Upravleniye, Vychislitel'naya Tekhnika*, 2013, no. 1(22), pp. 66—73 (in Russian).
4. **Borisova I. A., Kutnenko O. A.** *Tsenzurirovaniye oshibochno klassifitsirovannykh ob"yektov vyborki* (The erroneous classify objects in dataset censoring), *17-ya Vserossiyskaya konferentsiya "Matematicheskiye metody raspoznavaniya obrazov — 2015"*, Svetlogors, 2015 (in Russian).
5. **Ignat'ev N. A.** *Vychislitel'nyye Tekhnologii*, 2015, no. 6, pp. 34—43 (in Russian).
6. **Zguralskya E. N.** *Neyrokomp'yutery: Razrabotka, Primeneniye*, 2012, no. 2, pp. 20—27 (in Russian).
7. **Saidov D. Y.** *International Journal of Intelligent Systems and Applications (IJISA)*, 2017, vol. 9, no. 8, pp. 51—58.
8. **Jeffery I., Higgins D., Culhane A.** *BMC Bioinformatics*, 2006, vol. 7, pp. 359.
9. **Goodfellow I., Bengio Y., Courville A.** Deep Learning, Moscow, DMK Press, 2018, 652 p.
10. **Aivazyan S. A., Buchstaber V. M., Yenyukov I. S., Meshalkin L. D.** Applied statistics. Classification and reduction of dimensionality, Moscow, Finansy i statistika, 1989, 608 p. (in Russian).
11. Available at: http://archive.ics.uci.edu/ml/datasets

УДК 004.9, 004.94, 004.56 DOI: 10.17587/it.25.553-561

А. А. Коляда, д-р физ.-мат. наук, доц., e-mail: razan@tut.by,
П. В. Кучинский, д-р физ.-мат. наук, доц., e-mail: niipfp@bsu.by,
Научно-исследовательское учреждение "Институт прикладных физических проблем имени А. Н. Севченко" Белорусского государственного университета,
Н. И. Червяков, д-р техн. наук, проф., e-mail: Chervyakov@yandex.ru,
Федеральное государственное автономное образовательное учреждение высшего профессионального образования "Северо-Кавказский федеральный университет"

Пороговый метод разделения секрета на базе избыточных модулярных вычислительных структур[1]

Дана формализация порогового метода модулярного разделения секрета с использованием минимально избыточного кодирования. В частности, определены условия, обеспечивающие возможность применения в пороговых криптосхемах разделения секрета в качестве компьютерно-арифметической базы минимально избыточной модулярной арифметики (МИМА). В сравнении с неизбыточными аналогами МИМА позволяет уменьшить трудоемкость операции восстановления секрета-оригинала по долевым секретам, принадлежащим группам абонентов. Получены также условия корректности порогового принципа в рамках модулярного кодирования, и на этой основе разработан метод нейтрализации критичных ситуаций с обеспечением должного уровня криптостойкости.

Ключевые слова: *разделение секрета, минимально избыточная модулярная арифметика, пороговая схема модулярного разделения секрета, псевдослучайная маскирующая функция, распределенные вычисления*

Введение

Неотъемлемой составляющей современного процесса развития распределенных систем обработки данных различного назначения является обеспечение безопасности при хранении, обработке и передаче информации. Для решения обозначенной задачи криптография предоставляет весьма обширный функциональный инструментарий, охватывающий многообразные локальные и распределенные средства [1—12], такие, например, как электронная цифровая подпись, идентификация и аутентификация абонентов, безопасное хранение ключей и т.п. Криптографический ключ фактически представляет собой главный секрет во всем процессе шифрования [3]. Механизм ключей предполагает использование специальной операционной базы, которая обеспечивает генерирование, надежное хранение, маскирующее преобразование ключей в векторные представления и распределение их компонентов как долевых (частичных) секретов между абонентами системы, а также восстановление ключей-оригиналов по наборам долевых секретов. Перспективные технологии защиты данных в распределенных системах, такие как технология активной безопасности, предусматривающая периодическое обновление и разделение ключей участниками сеансов связи, предъявляют к реализационным характеристикам алгоритмов перечисленных операций высокие требования. Поэтому исследования по созданию новых высокопроизводительных технологий выполнения операций над ключами в рамках пороговых схем разделения секретной информации представляют актуальное, активно развиваемое в последние годы направление в криптографии [1, 3, 4, 10—12].

Как известно, компьютерно-арифметической базой средств защиты информации служит арифметика больших целых чисел (ЦЧ)

[1]Исследования выполнены при финансовой поддержке БРФФИ (Договор № Ф18-005 от 30 мая 2018 г.) и ГПНИ "Информатика, космос и безопасность" (2016—2020 гг.).

[1—3, 13—17], вследствие чего эффективность криптопреобразований на практике в решающей мере определяется реализационными свойствами применяемой технологии перевода осуществляемых вычислений из диапазона больших чисел (ДБЧ) в диапазоны ЦЧ стандартной разрядности. Ввиду кодового параллелизма кольцевых операций в модулярной системе счислений (МСС) и благодаря высокому уровню модульности криптографических процедур в свете сказанного в качестве компьютерно-арифметической основы для приложений рассматриваемого класса целесообразно принять модулярную арифметику (МА). Важным дополнительным фактором, указывающим на целесообразность применения МСС для построения пороговых криптосхем разделения секретной информации, является обеспечение возможности использования модулярной системы доступа, которая обладает естественным кодовым параллелизмом.

Обращаясь к существующим методам построения пороговых схем разделения секрета, прежде всего следует отметить предложенный А. Шамиром [18,19] метод, базирующийся на интерполяционных полиномах Лагранжа. Схема Шамира отличается высоким уровнем криптостойкости. Однако ее реализация требует довольно большого числа трудоемких мультипликативых операций и операций инвертирования ЦЧ по большому модулю. Кроме того, схемы данного класса используют сравнительно большой набор псевдослучайных параметров. Еще в большей мере указанные недостатки присущи предложенной в 1979 году векторной схеме Блэкли [20], которая основана на маскирующем преобразовании секрета-оригинала с помощью гиперплоскостей в l-мерном пространстве (l — число сторон, разделяющих секрет). Несмотря на то что схемы Шамира и Блэкли используют только один большой модуль p — модуль рабочего поля GF(p), они допускают применение высокопараллельных вычислительных структур, например, структур, состоящих из l однотипных независимых друг от друга компонентов. Однако в этом отношении более продуктивным представляется подход, предусматривающий использование l различных модулей, что означает переход к МСС. Применение МСС для построения пороговых схем разделения секрета [1, 3, 4, 21, 22] открывает принципиально новые возможности в части существенного уменьшения вычислительной сложности этапов выработки долевых секретов и восстановления по ним секрета-оригинала. Модулярное кодирование дает простой способ формирования частичных секретов как цифр кода МСС, получаемого в результате маскирующего преобразования, причем само это преобразование может иметь минимальную вычислительную сложность. Его роль может, например, выполнять линейное преобразование лишь с одним псевдослучайным коэффициентом.

Наиболее трудоемкой операцией в пороговой МА-криптосистеме разделения секрета является операция восстановления секрета-оригинала по модулярным кодам маскирующего аналога, т.е. по l-компонентным наборам долевых секретов. Ее сложность определяется используемой позиционной формой модулярных чисел и связанными с ней интегральными характеристиками кода МСС [1, 3, 4, 21—23]. Декодирующие процедуры (процедуры восстановления), осуществляющие прямую реализацию китайской теоремы об остатках [3, 4, 22], малоэффективны, так как они требуют вычислений по большому модулю, представляющему собой произведение оснований соответствующей l-модульной МСС. Фундаментальные преимущества МА наиболее полно удается реализовать в рамках так называемого минимально избыточного модулярного кодирования [1, 9, 23] и ассоциированной с ним интервально-модулярной формы чисел. Интегрально характеристическая база минимально избыточной МА (МИМА) позволяет минимизировать временные и аппаратурные затраты на выполнение немодульных операций, в том числе и операций восстановления секрета-оригинала.

Цель представляемого исследования — разработка принципиальных основ пороговых криптосхем разделения секрета с использованием минимально избыточного модулярного кодирования.

Решаемая в статье задача включает определение ограничений на диапазон изменения маскирующего аналога секрета-оригинала, обеспечивающих возможность применения МИМА в пороговой схеме рассматриваемого класса без снижения ее криптостойкости, и получение условий корректности развиваемого метода разделения секретной информации.

1. Постановка задачи и методы ее решения

Введем следующие обозначения:

$\lfloor a \rfloor$ и $\lceil a \rceil$ − наибольшее и наименьшее ЦЧ, соответственно не большее и не меньшее вещественной величины a;

$\mathbf{Z}_m = \{0, 1, ..., m-1\}$ — множество наименьших неотрицательных вычетов по натуральному модулю m;

$|a|_m = A(\mathrm{mod}m)$ — элемент множества $\mathbf{Z}_m$, сравнимый с ЦЧ A по модулю m;

$|A/B|_m$ — элемент χ множества $\mathbf{Z}_m$, удовлетворяющий сравнению $B\chi \equiv A(\mathrm{mod}m)$ (A и B – ЦЧ, $|B|_m \neq 0$);

$(\chi_1, \chi_2, \ldots, \chi_s) = \left(|X|_{m_1}, |X|_{m_2}, \ldots, |X|_{m_s}\right)$ — представление ЦЧ X (модулярный код) в МСС с основаниями m_1, m_2, ..., m_s, составляющими ее базис $\{m_1, m_2, ..., m_s\}$ ($s > 1$).

Пусть p_1, p_2, ..., p_n — упорядоченные по возрастанию попарно простые большие натуральные числа ($n > 1$);

$$P_i = \prod_{l=1}^{i} p_l \, (i = \overline{1,n});$$

$$\_P_j = \prod_{k=1}^{j} p_{n-k+1} = P_n / P_{n-j} \, (j = \overline{1,n});$$

$$\mathbf{P} = \{p_1, p_2, \ldots, p_n\};$$

$$\mathbf{I}\_l = \{\forall (i_1, i_2, \ldots, i_l) \mid 1 \leqslant i_1 < i_2 < \ldots$$
$$\ldots < i_l \leqslant n; 2 \leqslant t \leqslant l \leqslant n\}$$

(t — фиксированное натуральное число);

$$\mathbf{J}\_k = \{\forall (j_1, j_2, \ldots, j_k) \mid 1 \leqslant j_1 < j_2 < \ldots$$
$$\ldots < j_k \leqslant n; 2 \leqslant k < t\};$$

$I\_l = (i_1, i_2, \ldots, i_l)$ и $J\_k = (j_1, j_2, \ldots, j_k)$ — произвольные элементы соответственно множеств $\mathbf{I}\_l$ и $\mathbf{J}\_k$;

$$\mathbf{P}_{I\_l} = \{p_{i_1}, p_{i_2}, \ldots, p_{i_l}\};$$

$$\mathbf{P}_{J\_k} = \{p_{j_1}, p_{j_2}, \ldots, p_{j_k}\};$$

$$\mathbf{P}_{I\_l} = \prod_{j=1}^{l} p_{i_j}; \quad \mathbf{P}_{J\_k} = \prod_{i=1}^{k} p_{j_i}.$$

Концептуальную базу (t, n)-пороговой МИМА-схемы разделения секрета, которая рассчитана на полное число n и пороговое число t абонентов распределенной системы, составляют следующие определяющие ***положения***:

А. Исходный секрет, разделяемый n сторонами, представляет собой целое число $S \in \mathbf{Z}_p$, где p — большой модуль, взаимно простой с p_1, p_2, ..., p_n.

Б. Над S в МСС с базисом **P** выполняется маскирующее преобразование вида

$$\tilde{S} = S + Cp \qquad (1)$$

(C — псевдослучайная целочисленная величина).

Цифры $\tilde{\sigma}_i = \left|\tilde{S}\right|_{p_i} = \left|\sigma_i + |Cp|_{p_i}\right|_{p_i}$ ($\sigma_i = |S|_{p_i}$; $i = \overline{1,n}$) получаемого кода (σ_1, σ_2, ..., σ_n) рассматриваются как долевые (частичные) секреты, принадлежащие одноименным абонентам.

В. Любые t или более абонентов могут восстановить секрет-оригинал S по принадлежащим им маскирующим частичным секретам. Но никакая группа абонентов числом меньше t сделать этого не может.

Г. Область (диапазон) изменения маскирующего секрета $\tilde{S}$ согласуется с принципом минимально избыточного модулярного кодирования, что обеспечивает возможность выполнения декодирующей операции (операции восстановления секрета-оригинала) по упрощенным МИМА-процедурам.

Создание теоретической базы технологии реализации сформулированных основополагающих принципов построения пороговых МИМА-криптосхем разделения секрета является главной задачей представляемых исследований.

Применяемые методологические средства для решения поставленной задачи основаны на идее, которая предусматривает сужение области изменения маскирующего аналога $\tilde{S} = S + Cp$ секрета-оригинала S при выбранных p_1, p_2, ..., p_n до множества $\tilde{\mathbf{S}}$, допускающего определение на нем минимально избыточного модулярного кодирования. Несмотря на появляющуюся при этом незначительную (минимальную) дополнительную избыточность результирующего кода МСС она позволяет свести к теоретическому минимуму сложность операции расчета базовой интегральной характеристики кода (интервального индекса) [1, 23] и, как следствие, существенно упростить процедуру восстановления секрета-оригинала S. Для обеспечения корректности порогового принципа модулярного разделения секрета, т.е. для выполнения условия **В** (с необходимым уровнем криптостойкости), применяется метод нейтрализации порождаемых диапазоном $\tilde{\mathbf{S}}$ значений псевдослучайного параметра C, которые нарушают требование **В**.

2. Согласование диапазона изменения маскирующего секрета с принципом минимально избыточного модулярного кодирования

Рассмотрим два класса МСС, определяемых базисами

$$\mathbf{P}_{I\_l} = \{p_{i_1}, p_{i_2}, \ldots, p_{i_l}\}$$
$$(1 \leqslant i_1 < i_2 < \ldots < i_l \leqslant n; t \leqslant l \leqslant n)$$
и $$\mathbf{P}_{J\_k} = \{p_{j_1}, p_{j_2}, \ldots, p_{j_k}\}$$
$$(1 \leqslant j_1 < j_2 < \ldots < j_k \leqslant n; 2 \leqslant k < t).$$

К первому классу относятся МСС, отвечающие группам абонентов, число l которых не меньше порогового значения t, а ко второму — МСС, используемые группами из $k < t$ абонентов (см. условие **B**).

Ключевым аспектом проблемы выбора рабочего диапазона изменения секрета-маски (1) является поиск условий, обеспечивающих непересекаемость множеств (диапазонов) изменения целых чисел $\tilde{S}_{I\_l} = \tilde{S}(\mathrm{mod}\, P_{I\_l})$ и $\tilde{S}_{J\_k} = \tilde{S}(\mathrm{mod}\, P_{J\_k})$, имеющих в МСС с базисами $\mathbf{P}_{I\_l}$ и $\mathbf{P}_{J\_k}$ соответственно коды $(\tilde{\sigma}_{i_1}, \tilde{\sigma}_{i_2}, \ldots, \tilde{\sigma}_{i_l})$ и $(\tilde{\sigma}_{j_1}, \tilde{\sigma}_{j_2}, \ldots, \tilde{\sigma}_{j_k})$.

Справедливо следующее утверждение.

Теорема 1. Пусть основания базиса $\mathbf{P} = \{p_1, p_2, \ldots, p_n\}$ модулярной (t, n)-схемы разделения секрета $S \in \mathbf{Z}_p$ упорядочены по возрастанию (p — большой модуль). Для того чтобы диапазоны $\mathbf{Z}_{P_{J\_k}} = \{0, 1, \ldots, P_{J\_k} - 1\}$ изменения вычетов $\tilde{S}_{J\_k} = (\tilde{\sigma}_{j_1}, \tilde{\sigma}_{j_2}, \ldots, \tilde{\sigma}_{j_k})$ в МСС с базисами $\mathbf{P}_{J\_k} (2 \leqslant k < t)$ не пересекались с множеством значений секрета-маски $\tilde{S}$, имеющего в МСС с базисами $\mathbf{P}_{I\_l} (t \leqslant l \leqslant n)$ коды $(\tilde{\sigma}_{i_1}, \tilde{\sigma}_{i_2}, \ldots, \tilde{\sigma}_{i_l})$, достаточно выполнения условия

$$\tilde{S} \in \{\_P_{t-1}, \_P_{t-1} + 1, \ldots, P_t - 1\}. \qquad (2)$$

Доказательство. Предположим, что секрет S намерены разделить между собой любые l участников сеанса связи, за которыми закреплены основания базиса $\mathbf{P}_{I\_l}$. Поскольку модули базиса $\mathbf{P}$ криптосхемы упорядочены по возрастанию, то при всех рассматриваемых l выполняется неравенство

$$P_t \leqslant P_{I\_l} \left(l = \overline{t, n}\right). \qquad (3)$$

Маскирующий секрет (1) должен принадлежать диапазонам МСС с базисами $\mathbf{P}_{I\_l}$ — множествам $\mathbf{Z}_{P_{I\_l}}$ при всех $l = \overline{t, n}$. С учетом (3) для этого достаточно потребовать выполнения условия

$$\tilde{S} = S + Cp < P_t. \qquad (4)$$

Что касается модулей $p_{j_1}, p_{j_2}, \ldots, p_{j_k}$ класса $\mathbf{P}_{J\_k}$, отвечающих группам участников сеанса связи, число k которых меньше порогового значения t, то они удовлетворяют неравенству

$$P_{J\_k} \leqslant \_P_{t-1}. \qquad (5)$$

Как следует из соотношений (1)—(5), рабочий диапазон конструируемой пороговой схемы (по переменной $\tilde{S}$) легко может быть ориентирован исключительно только на значения $\tilde{S}$, отвечающие кодам $(\tilde{\sigma}_{i_1}, \tilde{\sigma}_{i_2}, \ldots, \tilde{\sigma}_{i_l})$ МСС с базисами $\mathbf{P}_{I\_l}$ $(t \leqslant l \leqslant n)$. Соответствующее множество значений секрета-маски $\tilde{S}$ описывается неравенством

$$\_P_{t-1} < S + Cp < P_t. \qquad (6)$$

Ввиду (5) диапазоны МСС с базисами $\mathbf{P}_{J\_k}$ находятся вне промежутка $(\_P_{t-1}; P_t)$. Кодам $(\tilde{\sigma}_{j_1}, \tilde{\sigma}_{j_2}, \ldots, \tilde{\sigma}_{j_k})$ указанных МСС отвечают вычеты $\tilde{S}_{J\_k}$ по модулям $P_{J\_k} \leqslant \_P_{t-1}$, т.е. ЦЧ, не принадлежащие диапазону изменения $\tilde{S}$ (см. условие (6)). Таким образом, ограничение (6) для $\tilde{S}$ обеспечивает непересекаемость множества результатов $\tilde{S}$ маскирования секрета S по правилу (1) с диапазонами $\mathbf{Z}_{P_{J\_k}}$ всех МСС, определяемых базисами $\mathbf{P}_{J\_k}$. *Теорема доказана.*

В случае применения маскирующего преобразования (1) областью изменения псевдослучайного параметра C, которая отвечает условию (6), служит множество

$$\mathbf{C} = \left\{\left\lfloor \frac{\_P_{t-1}}{p} \right\rfloor, \left\lfloor \frac{\_P_{t-1}}{p} \right\rfloor + 1, \ldots, \left\lfloor \frac{P_t - 1}{p} \right\rfloor\right\}. \qquad (7)$$

Замечание. Из вышеизложенного следует, что диапазоны $\mathbf{Z}_{P_{I\_l}}$ всех МСС, определяемых базисами $\mathbf{P}_{I\_l}$, включают диапазон $\mathbf{Z}_{P_t} = \{0, 1, \ldots, \_P_{t-1}, \_P_{t-1} + 1, \ldots, P_t - 1\}$ МСС с базисом $\{p_1, p_2, \ldots, p_t\}$. Это позволяет рассматривать множество $\tilde{\mathbf{S}} = \{\_P_{t-1}, \_P_{t-1} + 1, \ldots, P_t - 1\}$ как рабочий диапазон модулярной (t, n)-криптосхемы разделения секрета с основаниями $p_1, p_2, \ldots, p_n$ по переменной $\tilde{S} = S + Cp$. Поскольку восстановление секрета-оригинала S по кодам $(\tilde{\sigma}_{i_1}, \tilde{\sigma}_{i_2}, \ldots, \tilde{\sigma}_{i_l})$ маскирующего секрета $\tilde{S}$ в МСС с базисами $\mathbf{P}_{I\_l}$ является немодульной операцией, которая весьма сложна, особенно на диапазонах больших чисел, то ее целесообразно выполнять в рамках минимально избыточного кодирования [1, 9, 23], обеспечивающего минимизацию реализационных затрат. Предлагаемый подход предусматривает использование для выполнения необходимых немодульных операций МСС с базисами $\mathbf{P}_{I\_l}$ и диапазонами $\{0, 1, \ldots, p_0 P_{I\_(l-1)} - 1\} \subset$

$\subset \{0, 1, ..., P_{I\_l} - 1\}$, где p_0 — вспомогательный модуль, удовлетворяющий условию:

$$p_{i_l} \geqslant p_0 + l - 2 \quad (p_0 \geqslant t - 2) \tag{8}$$

при всех $l = \overline{t,n}$. Такие МСС называются *минимально избыточными* [23].

Переход от неизбыточных к соответствующим минимально избыточным МСС (МИМСС) влечет за собой замену рабочего диапазона $\tilde{\mathbf{S}}$ по переменной $\tilde{S}$ (см. замечание) на множество

$$\tilde{\mathbf{S}} = \{\_P_{t-1}, \_P_{t-1} + 1, \ldots, p_0 P_{t-1} - 1\}, \tag{9}$$

которое является составной частью диапазонов всех введенных МИМСС.

Сущность принципа минимально избыточного модулярного кодирования для пороговой (t, n)-криптосхемы разделения секрета раскрывает следующая теорема.

Теорема 2. Пусть $m_1 = p_{i_1}, m_2 = p_{i_2}, \ldots, m_l = p_{i_l}$; $M_{l-1} = \prod_{j=1}^{l-1} m_j$; $M_{j,l-1} = \dfrac{M_{l-1}}{m_j}$, $\tilde{\sigma}_j = \left|\tilde{S}\right|_{m_j} \left(j = \overline{1,l}\right)$; $I_l\left(\tilde{S}\right)$ — интервальный индекс (ИИ) числа $\tilde{S} = (\tilde{\sigma}_1, \tilde{\sigma}_2, \ldots, \tilde{\sigma}_l)$ в МСС с базисом $\{m_1, m_2, m_l\}$ [23], определяемый равенством

$$\tilde{S} = \sum_{j=1}^{l-1} M_{j,l-1} \left|M_{i,l-1}^{-1} \tilde{\sigma}_j\right|_{m_j} + M_{l-1} I_l\left(\tilde{S}\right). \tag{10}$$

Для того чтобы в МСС с основаниями m_1, m_2, ..., m_l ИИ $I_l(\tilde{S})$ каждого элемента $\tilde{S}$ диапазона $\{\_P_{t-1}, \_P_{t-1} + 1, \ldots, p_0 P_{t-1} - 1\}$ (p_0 — вспомогательный модуль) полностью определялся компьютерным ИИ-вычетом $I_l(\tilde{S}) = \left|I_l(\tilde{S})\right|_{m_l}$, необходимо и достаточно, чтобы l-е основание удовлетворяло условию $m_l \geqslant p_0 + l - 2 \quad (p_0 \geqslant l - 2)$.

При этом для $I_l(\tilde{S})$ верны следующие расчетные соотношения:

$$I_l(\tilde{S}) = \begin{cases} \hat{I}_l(\tilde{S}), & \text{если} \quad \hat{I}_l(\tilde{S}) < p_0; \\ \hat{I}_l(\tilde{S}) - m_l, & \text{если} \quad \hat{I}_l(\tilde{S}) \geqslant p_0; \end{cases} \tag{11}$$

$$\hat{I}_l(\tilde{S}) = \left| \sum_{j=1}^{l} R_{j,l}(\tilde{\sigma}_j) \right|_{m_l}; \tag{12}$$

$$R_{j,l}(\tilde{\sigma}_j) = \left| -m_j^{-1} \left|M_{j,l-1}^{-1} \tilde{\sigma}_j\right|_{m_j} \right|_{m_l} \quad (j \neq l), \qquad R_{l,l}(\tilde{\sigma}_l) = \left|M_{l-1}^{-1} \tilde{\sigma}_l\right|_{m_l}. \tag{13}$$

Позиционная форма (10) числа $\tilde{S}$ называется его интервально-модулярной формой по базису **M**. Применение минимально избыточного модулярного кодирования, сущность которого раскрывает ***теорема 2,*** снижает сложность расчета базовых интервально-индексных характеристик (см. (11)—(13)) практически до теоретического минимума. Это открывает принципиально новые возможности для повышения производительности пороговых МА-криптосхем разделения секрета.

3. Проблема корректности порогового принципа модулярного разделения секрета

Общая формула для восстановления секрета S по $\tilde{S}$ вытекает непосредственно из (1) и имеет вид

$$S = \left|\tilde{S}\right|_p. \tag{14}$$

В МИМСС с базисами $\mathbf{P}_{I\_l}$ $(t \leqslant l \leqslant n)$ и диапазонами $\left\{0, 1, \ldots, p_0 P_{I\_(l-1)} - 1\right\}$, содержащими множество (9) всех маскирующих секретов $\tilde{S}$ (см. теорему 1), преобразование $\tilde{S} \to S$ осуществляется корректно. Найдем ограничение на область изменения $\tilde{S}$, исключающее возможность восстановления S по $\tilde{S}\left(\mathrm{mod}\, P_{J\_k}\right) = \left(\tilde{\sigma}_{j_1}, \tilde{\sigma}_{j_2}, \ldots, \tilde{\sigma}_{j_k}\right)$ любыми k абонентами ($2 \leqslant k < t$), за которыми закреплены модули $p_{j_1}, p_{j_2}, \ldots, p_{j_k}$. Справедлива следующая теорема.

Теорема 3. Маскирующий (модифицированный) секрет $\tilde{S}$ и вычет $\tilde{S}\left(\mathrm{mod}\, P_{J\_k}\right)$ являются равноостаточными по модулю p, т.е. дающими при делении на p один и тот же остаток S, тогда и только тогда, когда целое число

$$Q = Q\left(\tilde{S}; J\_k\right) = \frac{\tilde{S}}{P_{J\_k}} \tag{15}$$

кратно модулю p.

Доказательство. Предположим, что число $\tilde{S}$ и вычет $\tilde{S}\left(\mathrm{mod}\, P_{J\_k}\right)$ по модулю P_{J_k} при делении на p дают один и тот же остаток. Ввиду (14) в этом случае

$$\left|\tilde{S}\left(\mathrm{mod}\, P_{J\_k}\right)\right|_p = \left|\tilde{S}\right|_p = S.$$

При этом

$$\tilde{S} \equiv \tilde{S}\left(\mathrm{mod}\, P_{J\_k}\right)(\mathrm{mod}\, p).$$

Отсюда следует, что разность $\tilde{S} - \tilde{S}\left(\mathrm{mod}\, P_{J\_k}\right)$ нацело делится на p. Согласно лемме Эвклида из теории делимости [1] ЦЧ $\tilde{S}$ с учетом обозначения (15) представимо в виде

$$\tilde{S} = \tilde{S}\left(\mathrm{mod}\, P_{J\_k}\right) + \frac{\tilde{S}}{P_{J\_k}} P_{J\_k} =$$
$$= \tilde{S}\left(\mathrm{mod}\, P_{J\_k}\right) + Q\left(\tilde{S}; J\_k\right) P_{J\_k}.$$

Следовательно

$$\tilde{S} - \tilde{S}\left(\mathrm{mod}\, P_{J\_k}\right) = Q\left(\tilde{S}; J\_k\right) P_{J\_k}. \qquad (16)$$

Так как левая часть равенства (16) при принятом предположении нацело делится на p, а модуль p взаимно прост со всеми основаниями базиса **P** криптосхемы (см. пункт **А**), то ЦЧ $Q\left(\tilde{S}; J\_k\right)$ кратно p.

Пусть теперь ЦЧ $Q\left(\tilde{S}; J\_k\right)$ нацело делится на p, тогда из (16) вытекает делимость разности $\tilde{S} - \tilde{S}\left(\mathrm{mod}\, P_{J\_k}\right)$ на модуль p. Это означает, что $\tilde{S} \equiv \tilde{S}\left(\mathrm{mod}\, P_{J\_k}\right)(\mathrm{mod}\, p)$. Таким образом, из кратности числа $Q\left(\tilde{S}; J\_k\right)$ модулю p следует равноостаточность по данному модулю $\tilde{S}$ и $\tilde{S}\left(\mathrm{mod}\, P_{J\_k}\right)$.

Теорема доказана.

Как показывает теорема 3, непересекаемость диапазона (9) принадлежности результатов $\tilde{S}$ маскирования секрета S с диапазонами $\mathbf{Z}_{P_{J\_k}}$ МСС, определяемых базисами $\mathbf{P}_{J\_k}$, полностью не исключает возможность восстановления k абонентами ($2 \leqslant k < t$) секрета S по соответствующим маскирующим аналогам — по модулярным кодам $(\tilde{\sigma}_{j_1}, \tilde{\sigma}_{j_2}, \ldots, \tilde{\sigma}_{j_k})$ вычетов $\tilde{S}\left(\mathrm{mod}\, P_{J\_k}\right)$. Это обеспечивает нейтрализация элементов диапазона $\mathbf{C} = \{C_{\text{НП}}, C_{\text{НП}} + 1, \ldots, C_{\text{ВП}}\}$ изменения псевдослучайного параметра C ($C_{\text{НП}}$ и $C_{\text{ВП}}$ — нижний и верхний пороги для C), которые порождают ЦЧ $Q\left(\tilde{S}; J\_k\right)$ вида (15), кратные модулю p. Искомые достаточные условия того, чтобы рассматриваемая (t, n)-криптосхема разделения секрета была пороговой, дает следующая теорема.

Теорема 4. Пусть $p_1, p_2, \ldots, p_n$ — упорядоченные по возрастанию попарно простые большие натуральные числа, используемые в качестве оснований модулярной криптосхемы разделения секрета $S \in \mathbf{Z}_p = \{0, 1, \ldots, p-1\}$ (p — большое число, взаимно простое со всеми $p_1, p_2, \ldots, p_n$) между n абонентами путем наделения их маскирующими частичными секретами $\tilde{\sigma}_i = \left|\tilde{S}\right|_{p_i}$ $\left(i = \overline{1, n}\right)$, где $\tilde{S} = S + Cp$; C — псевдослучайный целочисленный параметр. Для того чтобы любые l абонентов ($2 \leqslant t \leqslant l \leqslant n$; t — фиксированное ЦЧ), за которыми закреплены основания $p_{i_1}, p_{i_2}, \ldots, p_{i_l}$ $(1 \leqslant i_1 < i_2 < \ldots < i_l \leqslant n)$, могли восстановить секрет S по набору принадлежащих им частичных секретов — модулярному коду $\left(\tilde{\sigma}_{i_1}, \tilde{\sigma}_{i_2}, \ldots, \tilde{\sigma}_{i_l}\right)$ маскирующего секрета $\tilde{S}$, но никакая группа, включающая $k<t$ абонентов, которым отвечают основания $p_{j_1}, p_{j_2}, \ldots, p_{j_k}$ $(1 \leqslant j_1 < j_2 < \ldots < j_k \leqslant n)$, не имела возможности восстановления S по коду $(\tilde{\sigma}_{j_1}, \tilde{\sigma}_{j_2}, \ldots, \tilde{\sigma}_{j_k})$, достаточно выполнения следующей системы условий:

$$\begin{cases} \tilde{S} \in \left\{\tilde{S}_{\text{НП}}, \tilde{S}_{\text{НП}} + 1, \ldots, \tilde{S}_{\text{ВП}}\right\} \subseteq \\ \subseteq \{\_P_{t-1}, \_P_{t-1} + 1, .., p_0 P_{t-1} - 1\}, \\ C \in \left(\mathbf{C} \backslash \mathbf{C}_p\right), \end{cases} \qquad (17)$$

где $\tilde{S}_{\text{НП}}$ и $\tilde{S}_{\text{ВП}}$ — используемые нижнее и верхнее значения секрета-маски $\tilde{S}$;

$$\_P_{t-1} = \prod_{s=0}^{t-2} p_{n-s};\ P_{t-1} = \prod_{s=1}^{t-1} p_s;$$
$$\mathbf{C} = \{C_{\text{НП}}, C_{\text{НП}} + 1, \ldots, C_{\text{ВП}}\}$$
$$(C_{\text{НП}} = \lfloor \tilde{S}_{\text{НП}} / p \rfloor;\ C_{\text{ВП}} = \lfloor \tilde{S}_{\text{ВП}} / p \rfloor);$$
$$\mathbf{C}_p = \{C \in \mathbf{C} | S + Cp \in (\tilde{S}_{\text{НП}}; \tilde{S}_{\text{ВП}});$$

p — делитель ЦЧ $Q\left(\tilde{S}; J_k\right) = \dfrac{\tilde{S}}{P_{J_k}}$
$(1 \leqslant j_1 < j_2 < \ldots < j_k \leqslant n;\ 2 \leqslant k < t)\}$.

Сформулированная теорема практически является следствием теорем 1—3.

Элементы множества $\mathbf{C}_p$ из (17) могут быть рассчитаны предварительно и записаны в память. Сложность операции проверки принадлежности к $\mathbf{C}_p$ значений псевдослучайного параметра C при их генерировании в процессе маскирования секрета S по правилу (1), как того требует теорема 4, в решающей мере определяется мощностью $|\mathbf{C}_p|$ множества $\mathbf{C}_p$. В свете сказанного важнейшим оптимизационным аспектом проблемы синтеза модулярной пороговой (t, n)-криптосхемы разделения секрета, базирующейся на теореме 4, является минимизация характеристики $|\mathbf{C}_p|$.

4. Обсуждение результатов исследования

Конечной целью представляемого направления исследований является повышение эффективности модулярных пороговых схем разделения секрета за счет использования минимально избыточного кодирования. Разработанная теоретическая база МА-схем рассматриваемого класса позволяет уменьшить (в сравнении с аналогами) реализационные затраты (временные и аппаратурные) на выполнение этапа восстановления секрета-оригинала по наборам долевых секретов, принадлежащих группам абонентов числом не меньше порогового значения t, причем с обеспечением необходимого уровня криптостойкости.

Декодирующие процедуры (процедуры восстановления секрета-оригинала) как в неизбыточных, так и в минимально избыточных МСС базируются на операциях расширения кода. Выполнение этих операций с применением прямых реализаций китайской теоремы об остатках, т.е. выражений типа

$$\left|\tilde{S}\right|_{M_l} = \left|\sum_{j=1}^{l} M_{j,l} \left|M_{j,l}^{-1}\tilde{\sigma}_j\right|_{m_j}\right|_{M_l},$$

где $M_l = \prod_{s=1}^{l} m_s$; $M_{j,l} = \dfrac{M_l}{m_j}$; $m_1, m_2, \ldots, m_l$ — основания МСС (см. теорему 2), неэффективны, так как требуют вычислений по очень большим модулям M_l. Гораздо меньшей сложностью обладают процедуры, которые синтезируются на основе позиционных форм модулярных чисел, использующих интегральные характеристики кода, такие как ранг $\rho_l(\tilde{S})$ или коэффициенты полиадического представления ЦЧ $\tilde{S}$ [1, 3]. В частности, расчетное соотношение операции расширения кода на некоторый модуль m с помощью ранговой формы модулярных чисел имеет вид

$$\left|\tilde{S}\right|_m = \left|\sum_{j=1}^{l} M_{j,l} \left|M_{j,l}^{-1}\tilde{\sigma}_j\right|_{m_j} - M_l \rho_l(\tilde{S})\right|_m. \qquad (18)$$

В неизбыточной МСС вычисление характеристики $\rho_l(\tilde{S})$ занимает время порядка $O(l^2)t_{\text{сл}}$, где $t_{\text{сл}}$ — длительность операции сложения двух ЦЧ. Что касается минимально избыточной МСС, то формирование в ней базовой интегральной характеристики — интервального индекса по расчетным соотношениям (11)—(13) — осуществляется за l сложений. В сравнении с рангом это обеспечивает как минимум l-кратное сокращение временных затрат. Адекватное увеличение быстродействия достигается и для операции расширения кода (см. (10)—(13), (18)), а в конечном счете и для всего процесса восстановления секрета-оригинала S по коду $\left(\tilde{\sigma}_{i_1}, \tilde{\sigma}_{i_2}, \ldots, \tilde{\sigma}_{i_l}\right)$ [1, 9].

Отмеченное повышение производительности пороговой МИМА-схемы разделения секрета в сравнении с неизбыточными МА-аналогами достигается с сохранением предусматриваемого пороговым принципом уровня криптостойкости. Возникающие в рамках применяемого подхода для модулярного разделения секрета критические ситуации, описываемые ***теоремами 3*** и ***4***, устраняются путем нейтрализации соответствующих значений псевдослучайного параметра C в процессе генерирования маскирующего секрета $\tilde{S} = S + Cp$ (см. теорему 4).

Заключение

Основные результаты представленных в статье исследований по модулярным пороговым схемам состоят в нижеследующем.

1. Проведена формализация модели пороговой криптосхемы разделения секрета, компьютерно-арифметической базой которой служит МИМА. Благодаря снижению до теоретического минимума вычислительной сложности расчетных соотношений используемой в МИМА интегральной характеристики кода (интервального индекса) в рамках исследуемой модели достигается более высокий (в сравнении с традиционными решениями) уровень производительности на стадии декодирования секрета-оригинала.

2. Исходя из критерия простоты реализации для маскирования секрета-оригинала выбрана линейная функция с аддитивной вариационной компонентой псевдослучайного типа, которая кратна модулю p кольца принадлежности секрета. Это дает возможность выполнения декодирующей операции с помощью быстродействующих МИМА-процедур Монтгомери [9]. Показано, что адаптивное согласование диапазона изменения псевдослучайного параметра маскирующей функции с областью ее значений позволяет осуществлять минимально избыточную модулярную декомпозицию функции маскирования при любом допустимом базисе оснований схемы. Таким образом, минимально избыточные модулярные вычислительные структуры представляют

собой качественно новый инструментарий для реализации в пороговых схемах разделения секрета всех необходимых операций как при маскировании исходного секрета, так и в процессе его восстановления по кодам маскирующего аналога.

3. Для (t,n)-пороговой МИМА-схемы разделения секрета получены достаточные условия непересекаемости диапазона изменения секрета-маски с диапазонами МСС, определяемых k-компонентными базисами ($2 \leqslant k < t$), а также необходимое и достаточное условия равноостаточности по модулю p маскирующего секрета и отвечающего ему вычета в некоторой k-модульной МСС. Доказанные теоретические положения составляют основу корректной минимально избыточной реализации порогового принципа разделения секретной информации с обеспечением свойственного для схем исследуемого класса уровня криптостойкости.

Список литературы

1. **Червяков Н. И., Коляда А. А., Ляхов П. А.** и др. Модулярная арифметика и ее приложения в инфокоммуникационных технологиях. М.: ФИЗМАТЛИТ, 2017. 400 с.

2. **Ananda Mohan P. V.** Residue number systems: Theory and applications. Basel: Birghauser, Mathematics, 2016. 351 p.

3. **Червяков Н. И.** и др. Применение искусственных нейронных сетей и системы остаточных классов в криптографии. М.: Физматлит, 2012. 280 с.

4. **Galibus T. V., Matveev G. V.** Finite fields Grobner bases and modular secret sharing // J. of discrete mathematical sciences. 2012. Vol. 15, N. 6. P. 339—348.

5. **Schinianakis D., Stouraitis T.** Multifunction recidue architectures for cryptography // IEEE Trans. Circuits and Syst. I. 2014. Vol. 61, N. 4. P. 1156—1169.

6. **Alhasan A., Saeed I. Agbelnab.** The Hoffman's Method of Secured Data Encoding and Error Correction Using Residue Number System (RNS) // Communications and applied electronics (CAE). 2015. Vol. 2, N. 9. P. 14—18.

7. **Zalekian Azin, Mohammad Esmaeildoust, Amer Kaabi.** Efficient implementation of NTRU cryptography using residue number system // Int. Journal of Computer Applications. 2015. Vol. 124, N. 7. P. 33—37.

8. **Чен Ц., Яцкив В., Саченко А., Су Ц.** Беспроводные сенсорные сети на основе модулярной арифметики // Известия высших учебных заведений. Радиоэлектроника. 2017. Т. 60, N. 5. С. 274—285.

9. **Патент** на изобретение № 2652450 РФ.МПК:J06F7/57, H03K19/00. Устройство вычисления модулярного произведения Монтгомери / Н. И. Червяков (RU), А. А. Коляда (BY), В. А. Кучуков (RU), М. Г. Бабенко (RU). Заявка № 2017129526. Приоритеты: дата подачи заявки — 18.08.2017. Опубл. 26.04.2018, бюл. № 12.

10. **Gayathri B. NVSK, Rajendra G.** Efficient access control for security of cloud storage systems using RNS cryptography // Int. J. of Scientific research in computer science, engineering and information technology. 2018. Vol. 3, Iss. 4. P. 403—407.

11. **Bahramian Mojtaba, Khadijeh Eslami.** An efficient threshold verifiable multisecret sharing scheme using generalized jacobian of elliptic curves // Journal of algebraic structures and their applications. 2017. Vol. 4, Iss. 2. P. 45—55.

12. **Jia Xingxing, Daoshun Wang, Daxin Nie, Xiangyang Luo, Jonathan Zheng Sun.** A new threshold changeable secret sharing scheme based on the Chinese remainder theorem // Information sciences. 2019. Vol. 473. P. 13—30.

13. **Харин Ю. С., Берник В. И., Матвеев Г. В.** и др. Математические и компьютерные основы криптологии. Мн.: Новое знание, 2003. 382 с.

14. **Инютин С. А.** Основы модулярной алгоритмики. Ханты-Мансийск: Полиграфист, 2009. 347 с.

15. **Оцоков Ш. А.** Способ организации высокоточных вычислений в модулярной арифметике // Первая Международная конференция "Параллельная компьютерная алгебра и ее приложения в новых инфокоммуникационных системах". Ставрополь, РФ, 20—24 окт., 2014. Сб. науч. тр. Ставрополь: ИИЦ "Фабула", 2014. С. 270—277.

16. **Комарова Ю. А., Талалаев И. А.** Аналитический обзор методов и структур для работы с большими данными // Первая Международная конференция "Параллельная компьютерная алгебра и ее приложения в новых инфокоммуникационных системах". Ставрополь, РФ, 20—24 окт., 2014. Сб. науч. тр. Ставрополь: ИИЦ "Фабула", 2014. С. 477—485.

17. **Афонин М. С.** Способ обработки больших чисел на ПЛИС с малой ресурсной мощностью // Первая Международная конференция "Параллельная компьютерная алгебра и ее приложения в новых инфокоммуникационных системах". Ставрополь, РФ, 20—24 окт., 2014. Ставрополь: ИИЦ "Фабула", 2014. С. 511—520.

18. **Shamir A.** How to share a secret // Communications of the ACM. 1979. Vol. 22, N. 11. P. 612—613.

19. **Шнайер Б.** Алгоритмы разделения секрета. Схема интерполяционных полиномов Лагранжа // Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке Си. Н.: Триумф, 2002. С. 588—589.

20. **Blakley G. R.** Safe guarding cryptographic keys // Proc. Of the 1979 AFIPS national computer conference. Montvale: AFIPS press, 1979. P. 313—317.

21. **Asmuth C. A.,** Bloom J. A modular appoach to key safe guarding // IEEE Tras. On information theory. 1983. Vol. 29, N. 2. P. 208—210.

22. **Shiong, Jian Shyu, Ying- Ru Chen.** Treshold secret image sharing by Chinese remainder theorem // IEEE Asia — Pacific Services Computing conference. 2008. Vol. 1. P. 1332—1337.

23. **Коляда А. А., Пак И. Т.** Модулярные структуры конвейерной обработки цифровой информации. Минск: Университетское, 1992. 256 с.

A. A. Kolyada, Doctor of Physical and Mathematical Sciences, Associate Professor, e-mail: razan@tut.by,
P. V. Kuchynski, Doctor of Physical and Mathematical Sciences, Associate Professor, E-mail: niipfp@bsu.by,
Research establishment "Institute of Applied Physics Problems of A. N. Sevchenko" Belarusian State University,
N. I. Chervyakov, Doctor of Technical Sciences, Professor, e-mail: Chervyakov@yandex.ru, whbear@yandex.ru,
Federal State Autonomous Educational Institution of Higher Professional Education
"North-Caucasus Federal University"

The Threshold Method of Secret's Division Based on Redundant Modular Computing Structures

The article describes the formalization of the threshold method of modular separation of secret using minimally redundant coding. In particular, the conditions were determined to ensure the possibility of using minimum redundant modular arithmetic (MRMA) as the computer-arithmetic basis for the use in threshold cryptographic schemes of secret separation. In comparison with no redundant analogues, the MRMA allows to minimize the complexity of the operation of restoration of the secret-original on the basis of the share secrets belonging to groups of subscribers.

The conditions for the correctness of the threshold principle in the framework of modular coding are also obtained, and on this basis a method has been developed for neutralizing critical situations with ensuring an adequate level of cryptographic resistance.

Keywords: *separation of the secret, minimally redundant modular arithmetic, horny scheme of modular separation of the secret, pseudo-random masking function, distributed computing*

DOI: 10.17587/it.25.553-561

References

1. **Chervjakov N. I., Koljada A. A., Ljahov P. A.** ets. Modular arithmetic and its applications in infocommunication technologies, Moscow, FIZMATLIT Publ., 2017, 400 p. (in Russian).

2. **Ananda Mohan P. V.** Residue number systems: Theory and applications, Basel: Birghauser, Mathematics, 2016. 351 p.

3. **Chervjakov N. I.** The use of artificial neural networks and the residual class system in cryptography, Moscow, FIZMATLIT Publ., 2012, 280 p. (in Russian).

4. **Galibus T. V., Matveev G. V.** Finite fields Grobner bases and modular secret sharing, *J. of discrete mathematical sciences*, 2012, vol. 15, no. 6, pp. 339—348.

5. **Schinianakis D., Stouraitis T.** Multifunction recidue architectures for cryptography, *IEEE Trans. Circuits and Syst. I,* 2014, vol. 61, no. 4, pp. 1156—1169.

6. **Alhasan A., Saeed I. Agbelnab.** The Hoffman's Method of Secured Data Encoding and Error Correction Using Residue Number System (RNS), *Communications and applied electronics (CAE),* 2015, vol. 2, no. 9, pp. 14—18.

7. **Zalekian Azin, Mohammad Esmaeildoust, Amer Kaabi**. Efficient implementation of NTRU cryptography using residue number system*, Int. Journal of Computer Applications*, 2015, vol. 124, no. 7, pp. 33—37.

8. **Chen Ts., Yatskiv V., Sachenko A., Su Ts.** Wireless sensor networks based on modular arithmetic, *Izvestiya vysshikh uchebnykh zavedeniy. Radioelektronika (Proceedings of higher educational institutions. Radio electronics*), 2017, vol. 60, no. 5, pp. 274—285 (in Russian).

9. **Chervyakov N. I.** (RU), **Kolyada A. A.** (BY), **Kuchukov V. A.** (RU), **Babenko M. G.** (RU). The patent for the invention no. 2652450 РФ.МПК:J06F7/57,H03K19/00. Ustroystvo vychisleniya modulyarnogo proizvedeniya Montgomeri (Montgomery Modular Product Computing Device) Application no. 2017129526. Priorities: filing date — 18.08.2017. Posted by 26.04.2018, bulletin no. 12 (in Russian).

10. **Gayathri B. NVSK, Rajendra G.** Efficient access control for security of cloud storage systems using RNS cryptography*, Int. J. of Scientific research in computer science, engineering and information technology*, 2018, vol. 3, iss. 4, pp. 403—407.

11. **Bahramian Mojtaba, Khadijeh Eslami.** An efficient threshold verifiable multisecret sharing scheme using generalized jacobian of elliptic curves, *Journal of algebraic structures and their applications*, 2017, vol. 4, iss. 2, pp. 45—55.

12. **Jia Xingxing, Daoshun Wang, Daxin Nie, Xiangyang Luo, Jonathan Zheng Sun.** A new threshold changeable secret sharing scheme based on the Chinese remainder theorem, *Information sciences*, 2019, vol. 473, pp. 13—30.

13. **Kharin Yu. S., Bernik V. I., Matveyev G. V.** Mathematical and computer fundamentals of cryptology, Minsk, Novoye znaniye, 2003, 382 p. (in Russian).

14. **Inyutin S. A.** Basics of modular algorithms, Khanty-Mansiysk, Poligrafist, 2009, 347 p. (in Russian).

15. **Ocokov Sh. A.** The way to organize high-precision calculations in modular arithmetic, *Pervaja mezhdunarodnaja konferencija "Parallel'naja komp'juternaja algebra i ee prilozhenija v novyh infokommunikacionnyh sistemah",* Stavropol, Russian Federation, 20—24 okt., 2014, Fabula Publ., 2014, pp. 270—277 (in Russian).

16. **Komarova Ju. A., Talalaev I. A.** Analytical review of methods and structures for working with large data, *Pervaja mezhdunarodnaja konferencija "Parallel'naja komp'juternaja algebra i ee prilozhenija v novyh infokommunikacionnyh sistemah",* Stavropol, Russian Federation, 20—24 okt., 2014, Fabula Publ., 2014, pp. 477—485 (in Russian).

17. **Afonin M. S.** The way of processing large numbers on a FPGA with a small resource capacity, *Pervaja mezhdunarodnaja konferencija "Parallel'naja komp'juternaja algebra i ee prilozhenija v novyh infokommunikacionnyh sistemah",* Stavropol, Russian Federation, 20—24 okt., 2014, Fabula Publ., 2014, pp. 511—520 (in Russian).

18. **Shamir A.** How to share a secret, *Communications of the ACM*, 1979, vol. 22, no. 11, pp. 612—613.

19. **Shnajer B.** Secret sharing algorithms. Lagrange interpolation polynomial scheme, *Applied cryptography. Protocols, algorithms and source code in C*, N., Triumf, 2002, pp. 588—589 (in Russian).

20. **Blakley G. R.** Safe guarding cryptographic keys. Of the 1979 AFIPS national computer conference, Montvale, AFIPS press, 1979, pp. 313—317.

21. **Asmuth C. A., Bloom J.** A modular appoach to key safe guarding, *IEEE Tras. On information theory*, 1983, vol. 29, no. 2, pp. 208—210.

22. **Shiong Jian Shyu, Ying- Ru Chen**. Treshold secret image sharing by Chinese remainder theorem, *IEEE Asia — Pacific Services Computing conference*, 2008, vol. 1, pp. 1332—1337.

23. **Koljada A. A., Pak I. T.** Modular structures of conveyor processing of digital information, Minsk, Universitetskoe, 1992, 256 p. (in Russian).

УДК 004.051 DOI: 10.17587/it.25.562-572

Е. В. Кузнецова, канд. экон. наук, доц., доцент кафедры бизнес-аналитики школы бизнес-информатики факультета бизнеса и менеджмента, e-mail: Ev.Kuznetsova@hse.ru, Национальный исследовательский университет "Высшая школа экономики"

Автоматизация проектной деятельности в организациях, выполняющих контрактные ИТ-проекты

Для проектно-ориентированных ИТ-компаний определены функциональные области автоматизации и классы ПО, которое может использоваться для автоматизации процессов в этих областях. Выполнен обзор наиболее популярного ПО для управления проектами, портфелями проектов, управления рисками, систем-трекеров задач. Рассмотрены функциональные возможности ERP-систем SAP S/4 HANA, Microsoft Dynamics 365 for Finance and Operations, решения 1С:ERP + PM Управление проектной организацией 2 для автоматизации управления проектами. Выявлены преимущества использования ERP-систем для создания корпоративной системы управления проектами ИТ-компании. Предложены состав ПО и схема информационного взаимодействия.

Ключевые слова: *автоматизация управления ИТ-проектами, автоматизация управления портфелями проектов, ERP-системы, системы баг-трекеры, автоматизация управления рисками*

Введение

Традиционно автоматизированную корпоративную систему управления проектами (АКСУП) рассматривают как систему из трех составляющих: методологической, организационной и технологической (программное обеспечение (ПО) и ИТ-инфраструктура). Созданию методологической и организационной составляющих АКСУП посвящено подавляющее число научных публикаций в данной области; международные и национальные стандарты управления проектами (УП) и управления портфелями проектов (УПП) являются надежной основой для разработки корпоративной методологии и проведения организационных изменений при создании АКСУП. Складывается обманчивое впечатление, что создание технологической составляющей АКСУП не вызывает на практике каких-либо проблем. Так, в книгах, имеющих ярко выраженную практическую направленность и посвященных непосредственно созданию КСУП [1—3], вопросы автоматизации рассмотрены кратко и крайне поверхностно. Лишь в работе [4] выполнен обзор специализированного ПО. В то же время, по мнению автора, одна из главных проблем, с которой при создании АКСУП сталкиваются российские предприятия и ИТ-компании в частности, является недостаток функциональности отдельно взятых специализированных программных продуктов, используемых при автоматизации УП и УПП, т.е. невозможность реализовать все функциональные требования в рамках одной информационной системы (ИС). Поэтому распространена ситуация, когда в проектно-ориентированных организациях (ПОО) постепенно внедряется и одновременно используется большое число программных продуктов, решающих различные задачи УП и УПП, а интеграционные издержки и неэффективные трудозатраты, связанные с двойным вводом информации, весьма существенны.

В сети Интернет присутствует много информации, посвященной функциональным возможностям специализированного ПО для автоматизации процессов УП и УПП, однако отсутствуют рекомендации по комплексному подходу к автоматизации. Особенно важным комплексный подход к созданию технологической составляющей АКСУП является для предприятий, выполняющих контрактные проекты для внешних заказчиков. Как показано в работе [5], управление проектным бизнесом связано с высокими рисками для всех вовлеченных

в него сторон. Предприятия, выполняющие проекты для заказчиков, должны удовлетворить потребности последних, обеспечить прибыльность данных проектов и ликвидность своих активов. Для таких предприятий эффективность процессов УП, т.е. достижение результата с минимальными затратами, является существенным конкурентным преимуществом, особенно в такой динамично развивающейся отрасли, как информационные технологии.

Поэтому в данной статье автор специально рассматривает вопросы создания технологической составляющей АКСУП ИТ-компаний, т.е. вопросы выбора ПО, соответствующего специфике проектов, и создания адекватной ей функциональной ИТ-архитектуры.

Функциональные области автоматизации в ИТ-компаниях

Рассмотрим специфику деятельности ИТ-компаний, выполняющих контрактные проекты для внешних заказчиков, которая влияет на состав ПО для автоматизации проектной деятельности и выбор инструментов автоматизации.

1. Ведение проектной деятельности как основной операционной деятельности, что требует организации учета практически всех доходов и расходов предприятия в разрезе проектов.

2. Наличие предпроектной стадии, длительной и трудозатратной, связанной с участием в закупках, проводимых в соответствии с законодательством и нормативными документами потенциальных заказчиков, что требует учета и согласования предпроектных активностей.

3. Использование в качестве основных ресурсов проектов трудовых ресурсов, что влечет необходимость планирования и учета рабочего времени, а также контроля загрузки сотрудников проектной деятельностью.

4. Дробление работ проекта на относительно мелкие (трудоемкостью до одного человекочаса и менее) задачи.

5. Использование гибких (Agile) подходов к УП, предъявляющих специфические требования к планированию и выполнению работ проекта.

6. Применение трансфертного ценообразования в расчетах между подразделениями организации или организациями группы компаний, совместно участвующими в выполнении проектов.

С учетом данной специфики обобщенно можно выделить следующие функциональные области проектной деятельности, требующие автоматизации:

1) управление отдельными проектами;

2) УПП, включая управление предпроектной деятельностью;

3) управление рисками проектов;

4) организация производственной деятельности и непосредственно выполнение работ проекта, в том числе:

- детальное планирование работ и учет рабочего времени по детализированным работам;
- организация коммуникации участников проекта;

5) ведение бухгалтерского (финансового и управленческого) учета по проектам;

6) бюджетирование проектной деятельности, формирование план-факт отчетов.

Очевидно, что, даже если не рассматривать вопросы полномасштабной интеграции ИС, автоматизирующих деятельность в перечисленных функциональных областях, а исходить только из потребности подготовки в разных системах сопоставимой отчетности по проектной деятельности, необходимо использование единой кодировки проектов и отдельных работ проектов, синхронизации справочников сотрудников, контрагентов, договоров, поддержания единой статусной схемы проектов (как потенциальных, так и уже реализуемых). Таким образом, при наличии разнородного ПО необходима организации управления нормативно-справочной информации (НСИ) (MDM).

Для решения задачи создания оптимальной функциональной ИТ-архитектуры с одновременной минимизацией затрат на управление НСИ рассмотрим далее наиболее популярные программные продукты различных категорий из представленных на рынке, которые можно использовать для автоматизации перечисленных выше функциональных областей деятельности.

ПО для автоматизации процессов управления отдельными проектами и портфелями проектов

К основному функционалу, требующемуся для УП, можно отнести: структурную декомпозицию работ (СДР), календарное планирование работ, назначение трудовых ресурсов на работы с использованием корпоративного пула ресурсов, учет фактически отработанного времени, отслеживание сроков, формирование отчетности по проекту.

К основному функционалу, требующемуся для УПП, можно отнести: возможность описания и согласования инициатив в предпроектной деятельности, ведение корпоративного пула

ресурсов, планирование, учет и анализ использования трудовых ресурсов, поддержка стадий жизненного цикла и статусной модели проектов, контроль проведения расчетов с заказчиками, поставщиками, субподрядчиками, формирование отчетности по портфелю проектов.

В сети Интернет присутствует большое количество информации, посвященной обзору функционала и сравнению программных продуктов для УП, например, на сайте Московского отделения Project Management Institute [6]. Наиболее распространенными в России программными продуктами для УП являются, по опыту автора, не претендующему, конечно, на полноту, продукты зарубежных вендоров: Microsoft Project [7] и Oracle Primavera [8]. Обе системы полностью покрывают основной функционал, требующийся для управления отдельными контрактными ИТ-проектами в ПОО, однако явным лидером пользовательской аудитории является Microsoft Project. Oracle Primavera в основном используется при реализации крупных и сложных проектов с большим числом участников, особенно в машиностроении и строительстве, имеет отраслевые решения. Представлены на рынке и отечественные разработки. На портале TAdviser [9] приведены данные о статистике проектов внедрений ПО для УП в России. Лидерами по числу проектов в области автоматизации УП являются продукты Visari (вендор Бизнес Автоматика НТЦ), Адванта (вендор Адванта Консалтинг) и Microsoft Project. Третье место Microsoft Project можно объяснить тем, что в связи с простотой освоения и наличием большого количества обучающих материалов по данному продукту организации редко прибегают к услугам сторонних исполнителей в проектах автоматизации УП.

Популярность MS Project также обусловлена тем, что он распространяется в рамках пакета MS Office, имеет привычный и удобный для пользователя MS Office интерфейс. По данным Microsoft, их решения для управления проектами и портфелями насчитывают более 20 млн пользователей [10].

В компаниях, реализующих инвестиционные проекты, достаточно часто процессы УПП остаются неавтоматизированными. В ПОО эффективное управление портфелем контрактных проектов является не только конкурентным преимуществом, но и вопросом выживания на рынке. Поэтому для ПОО целесообразно рассматривать предлагаемые вендорами комплексные решения для управления отдельными проектами и портфелями проектов в целом.

По результатам отчета Gartner 2018 г. [11] в магическом квадрате среди систем управления портфелем проектов (Magic Quadrant for Project Portfolio Management, Worldwide) на мировом рынке лидируют вендоры Planview и CA Technologies. Microsoft занимает лидирующие позиции в разделе "Претенденты" ("Challengers").

В связи с лидирующими позициями Microsoft Project в России необходимо упомянуть комплексное решение для УП и УПП Microsoft Enterprise Project Management (EPM), включающее в себя помимо Project Professional также ПО для УПП Microsoft Project Server, а также Microsoft SharePoint Server — платформу для создания порталов и систем документооборота, совместной работы проектных команд любых размеров. Использование Microsoft Project Server позволяет компаниям [12]:

- централизовать сохранение сведений о проекте и использовать настраиваемые рабочие процессы управления на протяжении всего жизненного цикла проекта;
- отбирать проекты, ориентированные на стратегические бизнес-приоритеты с учетом бюджетных и ресурсных ограничений;
- обеспечить эффективное использование ресурсов и централизованное управление ими.

В Project Server существует модуль "Стратегия" для УПП, поддерживающий следующую функциональность:

- формирование перечней критериев для оценки проектов;
- ранжирование созданных критериев;
- ранжирование проектов путем их оценки относительно критериев;
- отбор проектов в портфель на основании проведенной оценки с учетом имеющихся ограничений, а также с учетом взаимных зависимостей проектов;
- анализ сформированного сценария портфеля;
- сравнение различных сценариев между собой.

Описанная выше функциональность Project Server по формированию портфеля проектов ориентирована в первую очередь на инвестиционные портфели и не используется ПОО для формирования контрактных портфелей проектов. Следует отметить, что наибольшей популярностью в России пользуются совместное применение Microsoft Sharepoint Server и Project. Использование Project Server существенно у́же.

ПО для управления рисками проектов

Основной требуемый функционал: идентификация и оценка рисков проектов, планирование мероприятий реагирования на риски и контроль их выполнения, поддержка статусной схемы рисков и мероприятий, контроль здоровья риска, проекта, задачи, ведение базы данных рисков, формирование отчетности.

На рынке представлено достаточно большое число специализированных решений, позволяющих автоматизировать процессы управления рисками (УР), в том числе RiskGap, Omnimet Risk Management, Palisade Software — @RISK — Industrial, SAS Risk Management. В качестве интересной функциональной возможности решения RiskGap можно отметить возможность проведения оценки рисков вместе с командой, что делает возможным его использование в ходе проведения риск-сессий [13].

При создании автоматизированных систем УР организации могут выбирать как использование специализированного узконаправленного готового решения для УР, так и внедрение платформенного решения по автоматизации всех служб контроля, в том числе и риск-менеджмента. Такое платформенное решение — АВАКОР — представлено российской компанией "Диджитал Дизайн" [14]. Данное решение позволяет автоматизировать процессы УР в комплексе с процессами внутреннего аудита и внутреннего контроля с применением технологий и подходов Big Data. Однако большинство ПОО, выполняющих контрактные ИТ-проекты, относится к сегменту малого и среднего бизнеса и не имеет специализированных служб внутреннего аудита и внутреннего контроля. Решение АВАКОР более привлекательно для крупных компаний с высоким уровнем зрелости системы внутреннего контроля.

Для ПОО в первую очередь интересна возможность привязки рисков к структуре проекта, назначение на роли в процессах УР сотрудников, выполняющих проект и являющихся его стейкхолдерами, построение отчетности по проектам, в которых риск является одним из многих показателей. Средствами Microsoft Project можно создать таблицы и настраиваемые поля для УР, например, как предложено в работе [15]. Определенные возможности в управлении рисками предоставляет использование Microsoft Project Server [16]. Однако описанная в этих источниках функциональность существенно уступает функциональности специализированных систем и не позволяет полностью автоматизировать все процессы УР.

ПО для автоматизации производственной деятельности и выполнения работ проекта

Чаще всего в ИТ-компаниях для детального планирования работ и учета рабочего времени по детализированным работам используются системы отслеживания ошибок (bug tracking system, BTS, баг-трекеры, таск-трекеры) — прикладные системы, призванные помочь разработчикам ПО в части учета и контроля ошибок, возникающих при тестировании или эксплуатации. Процесс функционирования таких систем в большей степени основан на работе с ошибками и/или дефектами, возникающими в процессе разработки. Эти системы позволяют хранить как минимум такую информацию об обнаруженных ошибках:

- кто сообщил о проблеме;
- дата и время, когда была обнаружена проблема;
- серьезность (статус) проблемы;
- описание неправильного поведения программы;
- ответственный исполнитель;
- плановая и фактическая трудоемкость решения проблемы;
- состояние решения проблемы.

Отслеживание этих свойств дефектов и ошибок оказалось возможным использовать для решения задач управления ИТ-проектами: детализации работ календарного плана на отдельные задачи, их описание и комментирование членами команды, определение приоритетных задач, распределение задач между исполнителями, контроль трудозатрат и текущего статуса задач.

Несомненным преимуществом BTS является возможность оперативного детального планирования работ и организации коммуникации членов команды проекта. Особенно существенным последнее преимущество является для территориально-распределенных команд, что часто встречается в ИТ-компаниях, и для проектов, которые реализуются с использованием гибких методологий.

Наиболее распространенными в российской практике BTS являются следующие системы:

- MantisBT — система с открытым исходным кодом. Разработана компанией Mantis на языке PHP и поддерживается операционными системами Linux, Windows и MacOS на стороне сервера. Система совместима с интернет-браузерами Chrome, Firefox, Safari, Opera и Internet Explorer 10+ [17];
- Redmine — гибкое веб-приложение для управления проектами, является систе-

мой с открытым кодом. Разработано Jean-Philippe Lang на языке программирования Ruby и представляет собой приложение на основе веб-фреймворка Ruby on Rails [18];
- Jira — позиционируется вендором Atlassian Software Systems как инструмент для управления проектами с использованием гибких методологий, который также используют для отслеживания ошибок. Имеет закрытый исходный код, является коммерческим продуктом, который разработан на языке Java [19].

Целесообразно провести сравнение этих продуктов в табличной форме по критериям, имеющим наибольшее значение для их настройки и использования (табл. 1). В табл. 2 приведены основные преимущества и недостатки рассматриваемых систем.

Возможности, которые предоставляет Jira для гибкого управления проектами:
- поддержка таких инструментов УП, как диаграмма Ганта, Scrum-доска и стена Kanban;
- наличие удобных инструментов расстановки приоритетов пользовательских историй, задач и багов в бэклоге продукта;
- возможность отслеживания выполнения проекта в сравнении с запланированным графиком;

Таблица 1

Сравнение систем Mantis, Redmine, Jira

Критерий	MantisBT	Redmine	Jira
Совместимость с ОС	Linux, OS X, Windows	Linux, OS X, Unix, Windows	Linux, Solaris, Windows
Пользовательский интерфейс	Web, e-mail, iPhone, Android	Web, E-mail, Atom, iPhone, Windows Phone, Android	Web, e-mail, RSS, iPhone, Android
Серверная часть	MySQL, PostgreSQL, MS SQL	MySQL, PostgreSQL, SQLite	DB2, Firebird, HSQLDB, MaxDB, Mckoi, MySQL, Oracle, PostgreSQL, SQL Server, Sybase ASA
Интеграция с системой управления версионностью	Да	Да	Да
Интеграция с Wiki-системами	Да, с системами MediaWiki, DokuWiki, XWiki	Да, integrated wiki, discussion forums, news blogs, email integration, calendars, Gantt Charts, Экспорт в PDF, Экспорт в Excel/CSV	Да, с системой Atlassian Confluence

Таблица 2

Достоинства и недостатки систем Mantis, Redmine, Jira

Система	Преимущества	Недостатки
MantisBT	Подсветка состояния ошибок, возможность интеграции с системами контроля версий, поддержка многоуровневой иерархии, гибкая система фильтров, поддержка русского языка	Сложности с операциями настройки и невозможность настроить внешний вид. Нельзя управлять правами доступа на уровне отдельных полей задачи. Можно управлять правами доступа на уровне проектов, но нельзя назначить права на какую-то версию проекта или отдельную задачу
Redmine	Поддержка плагинов, миграции с других BTS, гибкая система разграничения прав доступа пользователей, вложенные проекты неограниченной глубины, пакетное редактирование задач, возможность создавать дополнительные поля, настраивать их видимость, просмотр вложений из БТС, возможность создавать подпроекты и подзадачи, помечать задачи для отслеживания	Сложность установки и обновления (требуется язык программирования Ruby), а также отсутствие прав на отдельные типы переходов в Workflow
Jira	Дружелюбный интерфейс, мощная система фильтров, настраиваемая подсветка приоритетов, визуальный Workflow, различные диаграммы метрик, простая установка, удобные миграторы с других BTS, поддержка горячих клавиш, настраиваемый глобальный Dashboard, удобная установка. Подходит для крупных проектов. Наиболее дешевая из проприетарных систем	Платная система. Из-за многофункциональности пользователю-новичку сложнее ее освоить

- возможность каждого члена команды участвовать в обсуждении этапов проекта и поставленных задач;
- наличие системы оценивания, благодаря оценкам команда будет работать качественнее и эффективнее.

Таким образом, функционал данного продукта выходит за рамки BTS и позволяет рассматривать его и как ПО для управления отдельными проектами на основе гибких методологий.

Следует отметить, что хотя Jira является платным продуктом, он широко используется в ИТ-компаниях. Причины популярности не только в наличии функционала для Agile-команд. Jira часто выбирают те компании, которые используют Вики-систему Confluence того же вендора — Atlassian Software Systems.

В ходе непосредственного выполнения работ ИТ-проектов могут использоваться и другие категории ПО, такие как:

1) Вики-системы, например: Битрикс24, Яндекс.Вики, MediaWiki и уже упомянутая выше система Atlassian Software Systems;

2) системы для обеспечения управления процессами контроля качества на всех этапах разработки, например: HP Quality Cente, ALM (Application Lifecycle Management), Polarion ALM Борлас;

3) системы управления версиями кода, например: HG Mercurial, CVS (Concurrent Versions System), GIT, Baazar, Subversion(SVN).

Описание и анализ возможностей перечисленных систем выходят за рамки данной статьи. Однако их наличие на предприятии и использование в производственных процессах, несомненно, должно учитываться при проектировании функциональной ИТ-архитектуры в каждом конкретном случае.

ПО для ведения бухгалтерского (финансового и управленческого) учета проектной деятельности и бюджетирования проектов

Обзор программных продуктов для ведения бухгалтерского учета и бюджетирования выходит за рамки данного исследования. Однако с точки зрения оптимизации функциональной ИТ-архитектуры целесообразно рассмотреть применение ERP-систем, обладающих как развитым функционалом для ведения бухгалтерского учета, бюджетирования, так и наличием возможностей для автоматизации процессов УП и УПП.

Возможности ERP-систем для автоматизации управления проектами

По данным портала TAdviser [20], приведенным в табл. 3, лидерами по выручке от внедрения ERP-систем в России в 2016 г. были компании SAP, 1С, Microsoft. Поэтому целесообразно рассмотреть функциональные возможности для автоматизации процессов УП и УПП программных продуктов SAP HANA, 1С:ERP + PM Управление проектной организацией 2, Microsoft Dynamics 365 for Finance and Operations.

SAP S/4 HANA (SAP Business Suite 4 SAP HANA, ранее SAP ERP)

SAP S/4 HANA — передовая платформа в линейке продуктов SAP, расширяющая возможности ранее существовавших решений. Рассмотрим функциональные возможности, предоставляемые SAP S/4 HANA для организаций сферы R&D и инжиниринга по hf,jnt [21].

SAP S/4HANA включает:

- SAP S/4HANA Enterprise Management — управление предприятием;
- продукты SAP S/4HANA LoB (line of business), расширяющие основные функции SAP S/4HANA Enterprise Management для отдельных направлений деятельности;
- продукты SAP S/4HANA LoB Products для отдельных отраслей. SAP S/4HANA LoB Отраслевые решения расширяют основные

Таблица 3

Выручка от реализации ERP-проектов в России за 2016 г.

№	Компания	Выручка от ERP-проектов в 2016 г., млн руб.	Выручка от ERP-проектов в 2015 г., млн руб.	Динамика 2016/2015, %
1	SAP*	20800	19060	9,1
2	1С*	14000	12750	9,8
3	Microsoft*	3700	3390	9,1
4	Борлас	2459,1	2267,7	8,4
5	IBS	1945	1291	50,7
6	Oracle*	1700	1715	-0,9
7	Maykor-GMCS	1549,2	1231	25,8
8	Галактика	1409	1289	9,3
9	AT Consulting	1408,7	1216,6	16
10	Крок	1355	1294,9	4,6
Сумма		50326,0	45505,2	

* По оценке TAdviser

функции SAP S/4HANA Enterprise Management для отдельных отраслей в рамках направлений деятельности.

Использование модуля "Управления проектами" — PS — предоставляет возможности создания структурного и календарного плана проекта, определения потребности в трудовых, финансовых и материальных ресурсах, контроля выполнения работ и процессов закупки работ, услуг и материалов, формирования аналитической отчетности по проектам. Для этих целей в системе ведется СДР, где работы разделяются на этапы и затем детализируются до уровня отдельных операций путем создания сетевого графика проекта. В течение срока действия проекта сетевые графики используются как основа для планирования, анализа, управления и контроля календарных планов, сроков и ресурсов. Проводится автоматический анализ потребностей в материалах и ресурсах, учет затрат и контроль бюджета. Модуль "Управление проектами" поддерживает полный жизненный цикл проекта: от планирования до реализации и анализа результатов. Гибкая система позволяет настроить индивидуальные параметры для каждого проекта с учетом потребностей и особенностей работы предприятия.

Модуль "Управление проектами" интегрирован с другими модулями системы. Эффективность использования модуля "Управление проектами" достигается за счет его интеграции с другими модулями системы: Финансы (FI), Управленческий учет (CO), Управление материальными потоками (MM), Сбыт (SD).

Богатые возможности планирования и учета материальных затрат, детализации работ, возможности интеграции с системами сметного планирования делают решения на базе SAP для автоматизации проектной деятельности особо привлекательными в таких областях, как выполнение НИОКР и строительство. Для управления ИТ-проектами функционал SAP S/4HANA LoB и SAP S/4HANA LoB Products является, пожалуй, даже избыточным.

1С:ERP + PM Управление проектной организацией 2

Решение "1С:ERP + PM Управление проектной организацией 2" является совместным продуктом компании ITLand и фирмы "1С". Решение разработано на технологической платформе "1С:Предприятие 8.3". Рассмотрим далее функциональные возможности данной системы по источнику [22].

1. Планирование содержания и сроков проекта: СДР, планирование вех проекта, расчет календарного плана, фиксация базового плана проекта. План проекта может быть введен в систему комбинированным способом: из шаблона, файла MS Project, вручную. Поддерживается импорт и экспорт данных из/в MS Project.

2. Создание дерева ключевых показателей проекта и мониторинг их значений.

3. Планирование объемов и поставок проекта — результатов выполнения проектной задачи.

4. Планирование субподрядных работ и материальных затрат проекта.

5. Планирование персонала и трудозатрат проекта:
- назначение руководителей проектных задач;
- назначение трудовых ресурсов на проект и планирование трудозатрат проектных задач;
- моделирование эффективности проекта.

У работы может быть только один исполнитель, поэтому, если для выполнения работы требуется несколько исполнителей, то нужно либо декомпозировать работу до элементарных операций, либо объединить исполнителей в один трудовой ресурс.

6. Управление загрузкой и рабочим временем, включая:
- анализ загрузки трудовых ресурсов и выполнимости моделируемого портфеля проектов;
- планирование оперативной загрузки специалистов на проектах;
- учет рабочего времени по проектам.

7. Регистрация фактических данных о хозяйственных операциях и прохождении вех проекта.

8. Управление финансами проекта, включающее формирование бюджетов и план-фактный анализ.

9. Управление рисками проектов, включающее их идентификацию и оценку, планирование мероприятий по работе с рисками, фиксацию свершения рисков и мониторинг управления рисками проекта. Следует отметить, что наличие в системе функционала управления рисками является ее несомненным преимуществом перед другими ERP-системами.

10. Проведение оценки проекта и расчет цены контракта, в том числе с использованием шаблонов.

11. Управление портфелями и программами проектов.

Также в системе поддерживаются ведение стадий жизненного цикла и статусная модель проектов, возможности актуализации проектов, создания неограниченного числа версий

проектов, ведения базы знаний и использования шаблонов, формирования план-факта отчетности, в том числе расчет показателей освоенного объема, анализ затрат трудовых ресурсов, анализ контрольных событий и др.

К числу преимуществ данного решения нужно отнести распространенность программных продуктов 1С в России, оперативную реакцию вендора на изменения в бухгалтерском и налоговом учете, что позволяет организовать в данной системе ведение бухгалтерского учета в полном соответствии с российским законодательством, используя возможности соответствующего модуля 1С: ERP.

Microsoft Dynamics 365 for Finance and Operations

Компания Microsoft в 2017 г. выпустила на рынок ERP-систему для средних и крупных предприятий Microsoft Dynamics 365 for Finance and Operations, в состав которой входит модуль "Управление и отчетность по проектам". Рассмотрим функциональные возможности УП, реализуемые в данным модуле, опираясь на [23].

1. Создание коммерческого предложения заказчику проекта с оценкой по труду, расходам и материалам.

2. Соотнесение проектов с заключенными контрактами.

3. Планирование содержания и сроков: СДР, формирование графиков работ — оценка времени, которое необходимо для выполнения задачи, настройка взаимозависимостей задач и выбор дат начала и окончания задач. Для создания графиков поддерживается ведение календарей рабочего времени.

4. Оценка стоимости задачи на основе определения стоимости ресурсов: трудовых, материальных, расходов. Обеспечивается ведение стоимости труда и цен номенклатуры (закупочных и цен реализации заказчику).

5. Возможность использования внутрихолдинговых трудовых ресурсов, автоматизированное управление компетенциями трудовых ресурсов.

6. Учет и анализ фактически отработанного ресурсами времени.

7. Формирование прогнозов и бюджетов для отслеживания хода выполнения проекта. При бюджетировании используется система workflow-процессов, которая включает управление изменениями и делает возможным сохранение истории версий.

8. Создание заказов на продажу и заказов на покупку по проекту, учет поступлений покупок и начисления затрат по проекту, выставление накладных для проектов различных типов: "Фиксированная цена" и "Время и расходы".

9. Управление затратами и выручкой при внутригрупповых операциях с использованием трансфертных цен.

Рассмотрим подробнее последнюю из перечисленных возможностей, поскольку она является очень полезной для ПОО и соответствует специфике их деятельности.

В компании может быть несколько подразделений, филиалов, а также и других юридических лиц, которые обмениваются между собой продуктами и услугами при выполнении проектов. Юридическое лицо, которое предоставляет услуги или продукт, называется сдающим в аренду юридическим лицом, а юридическое лицо, которое получает услугу или продукт, называется заимствующим юридическим лицом. На рис. 1 показан сценарий, в котором два юридических лица SI FR (заимствующее юридическое лицо) и SI USA (сдающее в аренду юридическое лицо) совместно используют ресурсы для выполнения проекта для заказчика A. В этом сценарии SI FR обязуется выполнить работу для заказчика A.

Использование Microsoft Dynamics 365 for Finance and Operations обеспечивает:

- создание накладных заказчика для проекта в заимствующем юридическом лице с помощью внутрихолдинговых табелей учета рабочего времени, расходов и накладных поставщика в сдающем в аренду юридическом лице;
- поддержку расчетов налогов и косвенных затрат;
- перенос признания выручки в сдающем в аренду юридическом лице и времени признания затрат заимствующим юридическим лицом;

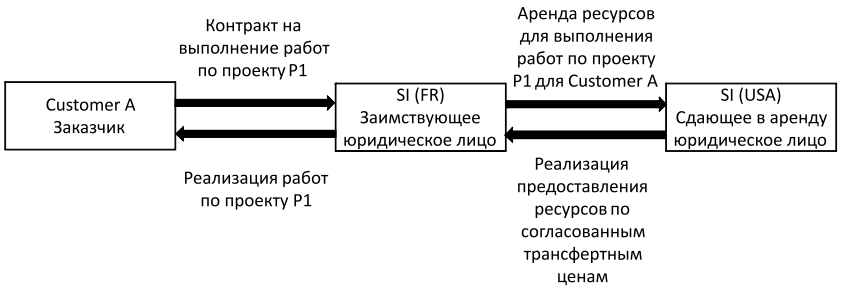


Рис. 1. Схема внутрихолдинговой реализации работ и услуг [23]

- начисление дохода по незавершенному производству в сдающем в аренду юридическом лице;
- настройку трансферной цены, которая может основываться на различных моделях ценообразования.

Предлагаемое решение

Предлагаемая в настоящей работе схема информационного взаимодействия АКСУП проектно-ориентированной группы ИТ-компаний с внешними системами приведена на рис. 2. На данной схеме специфическая функциональность УП, УПП и управления рисками реализуется на базе Microsoft Dynamics 365 for Finance and Operations. Данный выбор обусловливается целесообразностью автоматизировать процессы УП, УПП, управления рисками проектов, ведения управленческого учета и бюджетирования всех юридических лиц, входящих в группу, в единой системе. Функциональность решения условна разделена на пять модулей: Проекты, Бюджетирование, Управленческий учет, Управление персоналом (HR), Управление рисками. Необходимо отметить, что для автоматизации процессов управления рисками требуется выполнить значительные доработки. Выбор Microsoft Dynamics 365 for Finance and Operations в данном случае обусловлен наличием в рассматриваемой ИТ-компании большого и успешного опыта внедрения продуктов Microsoft и специалистов соответствующей квалификации. В общем случае при выборе ERP-системы для создания АКСУП ПОО необходимо учитывать такие факторы, как наличие необходимой функциональности, стоимость, возможность интеграции с существующими на предприятии системами, наличие персонала соответствующей компетенции и др. [24].

Помимо уже рассмотренных выше преимуществ, предлагаемый подход позволяет централизовать управление проектным персоналом группы компаний и организовать в рамках компании "биржу трудовых ресурсов".

Внешними системами по отношению к АКСУП являются системы, исторически существующие в группе компаний:

- система корпоративного казначейства группы компаний, в которую входит ИТ-компания;
- система для детального планирования работ Jira;
- система бухгалтерского (финансового) учета.

На рис. 2 показано, какие системы являются мастер-системами для ведения НСИ определенного рода.

В системе Jira исторически ведется учет фактического рабочего времени сотрудников по

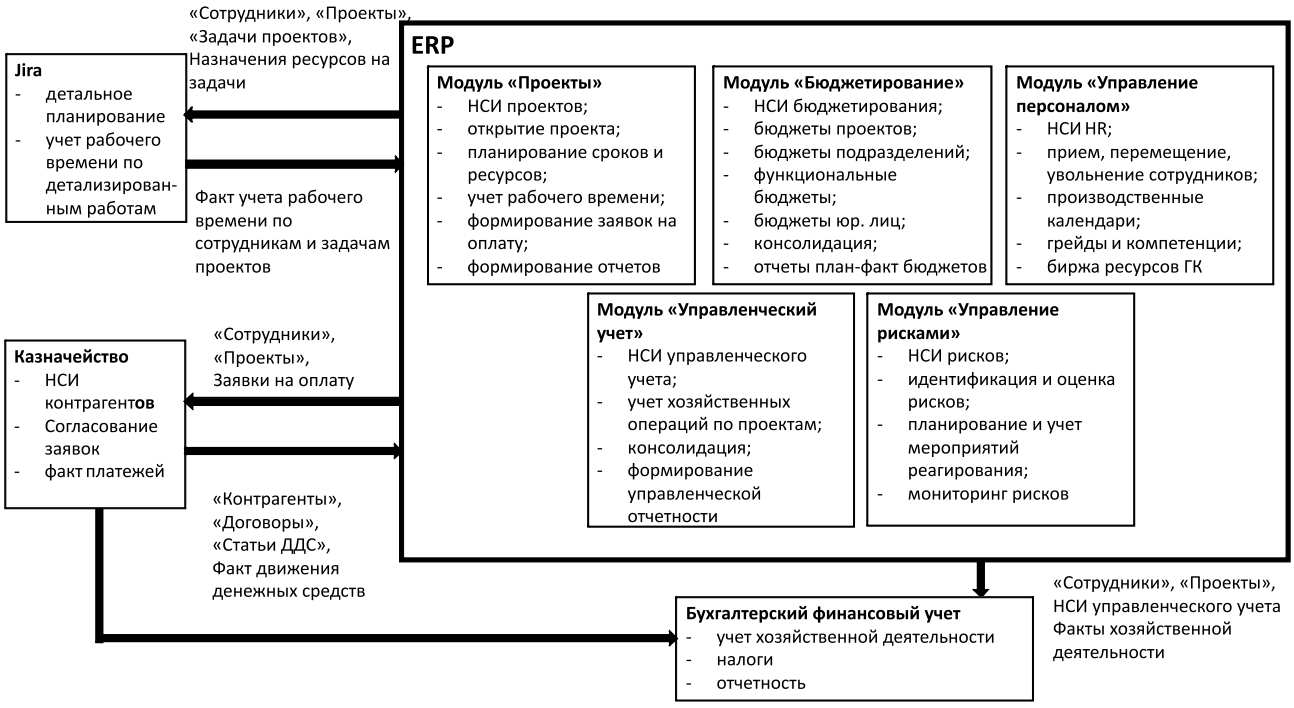


Рис. 2. Схема информационного взаимодействия. Источник: разработано автором (ГК — группа компаний; ДДС — движение денежных средств)

задачам, детализированным до уровня отдельных операций. В системе ERP предлагается вести справочник "Задачи проекта", элементы которого будут использоваться в качестве задач верхнего иерархического уровня в системе Jira. Данные о фактически отработанном времени сотрудников, агрегированные до уровня задач проекта, передаются в ERP-систему.

Использование описанного подхода позволяет не только снизить интеграционные издержки, но и сделать более "прозрачными", контролируемыми и эффективными процессы УП на всем протяжении жизненного цикла проектов, "встроить" систему бюджетирования проектов в общую систему бюджетирования группы компаний, повысить ценность контрактного портфеля ИТ-проектов одновременно со снижением уровня рисков.

Список литературы

1. **Богданов В. В.** Управление проектами. Корпоративная система — шаг за шагом. М.: Манн, Иванов и Фербер, 2012. 248 с.
2. **Береговенко А.** Корпоративная система управления проектами. Пособие для настройки успешного бизнеса, реальные кейсы. Ridero, 2018, 100 с.
3. **Нугайбеков Р. А., Максин Д. Г., Ляшук А. В.** Корпоративная система управления проектами: от методологии к практике. М.: Альпина Паблишер, 2015. 234 с.
4. **Илларионов А. В., Клименко Э. Ю.** Портфель проектов: Инструмент стратегического управления предприятием. М.: Альпина Паблишер, 2013. 312 с.
5. **Леманн Оливер.** Проектный бизнес и неоплачиваемые заказчиком проекты // Управление проектами и программами. 2018. № 4. С. 262—275.
6. **Сайт** Московского отделения PMI. URL: https://pmi.ru/infosystem/
7. **Microsoft** Project. URL: https://products.office.com/ru-ru/project/project-and-portfolio-management-software
8. **Oracle** Primavera. URL: https://www.oracle.com/industries/construction-engineering/index.html
9. **Данные** портала TAdvise о системах управления проектами. URL: http://www.tadviser.ru/index.php/Системы_управления_проектами?cache=no&ptype=system#ttop
10. **Партнерство** с Microsoft. URL: https://partner.microsoft.com/ru-ru/membership/project-portfolio-management-competency
11. **Daniel B. Stang, Matt Light, Teresa Jones.** Magic Quadrant for Project Portfolio Management, Worldwide. URL: www.gartner.com
12. **Microsoft** Project Server. URL: https://technet.microsoft.com/ru-ru/library/fp179724.aspx
13. **RiskGap**. URL: www.riskgap.ru
14. **АВАКОР.** URL: https://digdes.ru/info/avtomatizatsiya-protsessov-upravleniya-riskami
15. **Управление** рисками в Project 2016. URL: https://blogs.technet.microsoft.com/project_ru/2016/01/19/project-2016-1-10/
16. **Microsoft** Project Server. URL: https://support.office.com/ru-ru/article/Добавление риска в проект-7aa1acc9-50cf-4f15-ac3b-fedf41b31c83
17. **Mantis** Bug Tracker. URL: https://www.mantisbt.org/
18. **Redmine.** URL: https://www.redmine.org/
19. **JIRA** Software. URL: https://ru.atlassian.com/software/jira
20. **Данные** портала TAdviser о выручке от внедрения ERP-систем. URL: [http://www.tadviser.ru/index.php/ERP]
21. **SAP** S/4HANA 1809 — Feature Scope Description. URL: https://help.sap.com/doc/e2048712f0ab45e791e6d-15ba5e20c68/1809.000/en-US/FSD_OP1809.pdf
22. **1C:ERP + PM** Управление проектной организацией 2. URL: https://solutions.1c.ru/catalog/erp-pm/features
23. **Microsoft** Dynamics 365 for Finance and Operations. URL: https://docs.microsoft.com/ru-ru/dynamics365/unified-operations/financials/project-management/overview-project-management-accounting
24. **Кряжев С. А., Кузнецова Е. В., Макаров Е. Н.** Управление портфелем ИТ-проектов как инструмент реализации ИТ-стратегии // Информационные технологии. 2017. Т. 23. № 11. С. 833—840.

E. V. Kuznetsova, Candidate of Economic Science, Docent, Associate Professor, e-mail: Ev.Kuznetsova@hse.ru, National Research University Higher School of Economics, Moscow, Russian Federation

Project Activity Automation for Organizations Performing IT Projects

For project-oriented IT companies the functional areas of process automation and software classes for such automation in these areas are defined in the paper. The overview of most popular software for project and project portfolio management, risk management and task tracker systems is presented. The functional capabilities for project management automation are discussed for such software as ERP-systems SAP S/4 HANA, Microsoft Dynamics 365 for Finance and Operations, 1C:ERP + PM Управление проектной организацией 2 solution. The benefits of ERP-systems for IT company corporate project management system design are revealed. The software structure and corresponding data interaction are proposed.

Keywords: *project Management automation, Portfolio Management automation, ERP-systems, bug tracking system, Risk Management automation*

DOI: 10.17587/it.25.562-572

References

1. **Bogdanov V. V.** Project Management. Corporate system — step by step, Moscow, Mann, Ivanov and Ferber, 2012, 248 p. (in Russian).
2. **Berehovenko A.** Corporate project management system. A guide for setting up a successful business, real cases, Ridero, 2018, 100 p. (in Russian).
3. **Nugaybekov R. A., Maksin D. G., Lyashuk A. V.** Corporate project management system: from methodology to practice, Moscow, Alpina Publisher, 2015, 234 p. (in Russian).
4. **Illarionov A. V., Klimenko E. Yu.** Project portfolio: A tool for strategic enterprise management, Moscow, Alpina Publisher, 2013, 312 p. (in Russian).
5. **Lehmann O.** Project business and projects unpaid by the customer, *Project and Program Management*, 2018, no. 4, pp. 262—275 (in Russian).
6. **The site** of the Moscow branch of PMI, available at: https://pmi.ru/infosystem/ (in Russian).
7. **Microsoft** Project, available at: https://products.office.com/ru-ru/project/project-and-portfolio-management-software
8. **Oracle** Primavera, available at: https://www.oracle.com/industries/construction-engineering/index.html
9. **TAdvise** portal data on project management systems, available at: http://www.tadviser.ru/index.php/Project_control_systems?Cache=no&ptype=system#ttop (in Russian).
10. **Partnership** with Microsoft, available at: https://partner.microsoft.com/ru-ru/membership/project-portfolio-management-competency (in Russian).
11. **Daniel B., Stang, Matt Light, Teresa Jones.** Magic Quadrant for Project Portfolio Management, Worldwide, available at: www.gartner.com
12. **Microsoft** Project Server, available at: https://technet.microsoft.com/ru-ru/library/fp179724.aspx
13. **vRiskGap**, available at: www.riskgap.ru
14. **AVAKOR,** available at: https://digdes.ru/info/avtomatizatsiya-protsessov-upravleniya-riskami
15. **Risk** management in Project 2016, available at: https://blogs.technet.microsoft.com/project_ru/2016/01/19/project-2016-1-10/ (in Russian).
16. **Microsoft** Project Server, available at: https://support.office.com/ru-ru/article/Adding risk to the project-7aa1acc9-50cf-4f15-ac3b-fedf41b31c83
17. **Mantis** Bug Tracker, available at: https://www.mantisbt.org/
18. **Redmine,** available at: https://www.redmine.org/
19. **JIRA** Software, available at: https://ru.atlassian.com/software/jira
20. **Data** portal TAdviser on revenue from the introduction of ERP-systems, available at: [http://www.tadviser.ru/index.php/ERP] (in Russian).
21. **SAP** S / 4HANA 1809 — Feature Scope Description, available at: https://help.sap.com/doc/e2048712f0ab45e791e6d-15ba5e20c68/1809.000/en-US/FSD_OP1809.pdf
22. **1C:** ERP + PM Management of the project organization 2, available at: https://solutions.1c.ru/catalog/erp-pm/features
23. **Microsoft** Dynamics 365 for Finance and Operations, available at: https://docs.microsoft.com/ru-ru/dynamics365/unified-operations/financials/project-management/overview-project-management-accounting
24. **Kryazhev S. A., Kuznetsova Ye. V., Makarov E. N.** Portfolio Management of IT Projects as a Tool for Implementing an IT Strategy, *Informacionnye texnologii,* 2017, vol. 23, no. 11, pp. 833—840 (in Russian).

УДК 004.032.26 DOI: 10.17587/it.25.572-576

С. Е. Левин, первый зам. гендиректора, **Я. Н. Окрент,** гл. науч. сотр. e-mail: info@pkcc-ps,
ООО "Российская корпорация средств связи — Программные системы",
С. Я. Нагибин, д-р техн. наук, проф., зав. кафедрой, **Н. Е. Балакирев,** канд. техн. наук, доц.,
Московский авиационный институт (национальный исследовательский университет)

Математическая модель технологического процесса производства стирола

Приведена математическая модель технологического процесса производства стирола с использованием нейросетевых технологий. Рассмотрено применение искусственной нейронной сети прямого распространения с одним скрытым слоем, обученной на экспериментальной выборке. Приведен алгоритм формирования нейронной сети. Модель реализована в виде программного модуля. Приведены результаты прогнозирования процесса производства стирола на реальных данных и рекомендации по использованию разработанной модели в процессе оценки промышленной безопасности особо опасных производственных процессов.

Ключевые слова: *промышленная безопасность, искусственные нейронные сети, математическая модель производства стирола, нормализация параметров, обучение нейронной сети, метод обратного распространения ошибки, алгоритм формирования нейронной сет, оценка результатов прогнозирования*

Введение

При оценке рисков промышленной безопасности особо опасных производственных объектов определяющими факторами не всегда являются характеристики надежности оборудования. На промышленную безопасность сложного технологического процесса влияют множество факторов как внешнего, так и внутреннего характера.

Качество исходного сырья, погодные условия, влияние автоматизированных систем и

эксплуатирующего персонала на технологический процесс, профессиональная подготовка персонала и многие другие факторы могут внести решающий вклад в развитие негативных явлений, вплоть до потери контроля над технологическим процессом.

Раннее обнаружение этих факторов важно для своевременного принятия предупредительных мер, которые позволят предотвратить их негативное влияние [1].

Для изучения аномального поведения сложных технологических систем применяют, как правило, методы математического моделирования, которые позволяют свести задачу изучения различных процессов к задаче изучения свойств математической модели, представляющей собой систему уравнений, описывающих процессы.

Несколько более сложной задачей является оценка надежностных характеристик производства, в которых протекают химические процессы. Химия — наука экспериментальная. Все результаты исследований строения и реакций веществ проверяются на опыте с последующими рекомендациями к практическому использованию. Моделирование свойств и реакционной способности химических соединений является составной частью общей стратегии исследований. Математическая модель с помощью определенного алгоритма позволяет прогнозировать течение химических процессов. При этом в данном случае при моделировании необходимо, кроме теории надежности и математической статистики, использовать аппарат теории катастроф, теории хаоса и теории нелинейных динамических систем [2].

Стирол — один из важнейших продуктов нефтехимии, сырье для получения полимеров (полистирол, синтетический каучук) и сополимеров (ударопрочный полистирол на основе акрилонитрила и бутадиена), относится ко второму классу опасности. Производство стирола — крупнотоннажное, единичная мощность современных агрегатов составляет 150—300 тысяч тонн стирола в год. Основным промышленным способом производства стирола в настоящее время является дегидрирование этилбензола.

Постановка задачи

Необходимо построить математическую модель процесса производства стирола в целях прогнозирования его поведения при изменяющихся входных условиях.

Основные этапы моделирования

Для связи входных параметров и выходной характеристики предложено использовать сложную математическую модель искусственной нейронной сети (ИНС). Нейросети являются эффективным методом имитации процессов, который позволяет выявлять сложные зависимости между входными и выходными характеристиками [3, 4].

Процесс математического моделирования включает в себя следующие этапы:

1) подготовка данных;
2) формирование нейронной сети;
3) обучение нейронной сети;
4) тестирование нейронной сети.

Первый этап: подготовка данных

На первом этапе был проведен анализ данных, рассмотрены зависимости значений параметров установки производства стирола от времени, давления и температуры.

Для перехода к общим размерностям необходимо провести нормализацию двнных:

$$\tilde{x} = \frac{2\left(x - x_{\min}\right)}{x_{\max} - x_{\min}} - 1,\ x \in \left[-1;1\right].$$

Нормализация выполняется, когда на различные входы нейронов подаются данные разной размерности. При отсутствии нормирования значения на втором входе будут всегда оказывать существенно большее влияние на выход сети, чем значения на первом входе. При нормировании размерности всех входных и выходных данных сводятся воедино.

Для успешного прогнозирования поведения системы необходимо оставить в обучающей выборке только те значения, которые характеризуют общую тенденцию развития системы, в противном случае многочисленные выбросы, которые могут быть ошибками снятия показаний, будут "отвлекать" модель от общего направления развития, увеличивая разброс выходных данных (рис. 1).

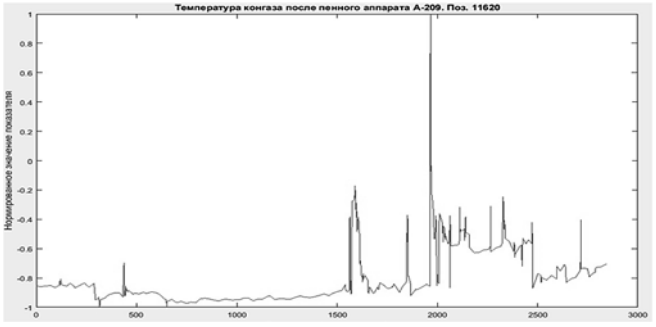


Рис. 1. Выброс на примере показателя температуры

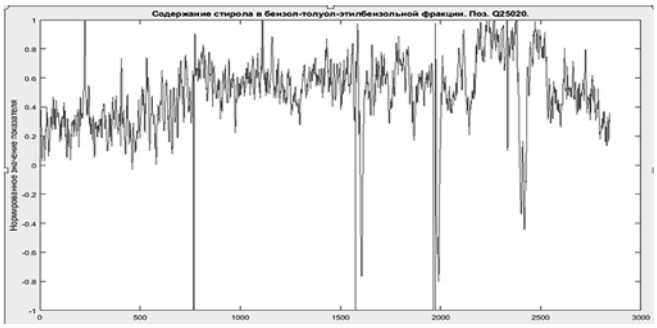


Рис. 2. Выбросы на примере показателя содержания стирола

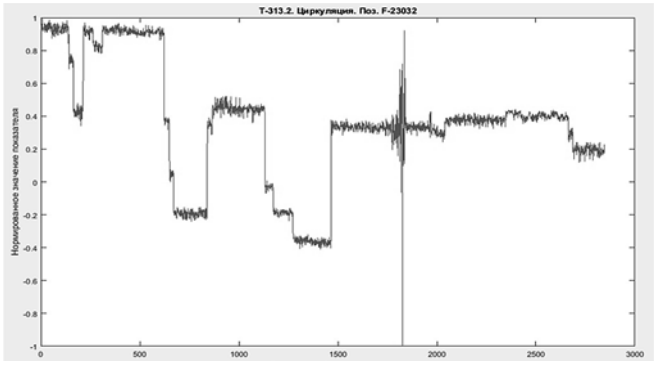


Рис. 3. Выбросы на примере показателя циркуляции

В то же время причиной некоторых выбросов или даже целых их серий могут быть определенные физические явления, которые система должна уметь выявлять и предсказывать (рис. 2, 3).

Для этого перед фильтрацией данных необходимо провести анализ по выявлению причин выбросов.

Конечным этапом подготовки выявления зависимостей между параметрами является сглаживание их значений. Для этого задается диапазон допустимых значений параметра, а для точек, которые не принадлежат данной области, ищется новое значение путем интерполяции. При вариации параметра разброса σ получаются разные модели кривых. Заметим, что чем больше σ, тем ближе построенная кривая к экспериментальной, но в то же время в модель включается большее число выбросов (рис. 4, см. четвертую сторону обложки).

Второй этап: формирование нейронной сети

Для решения поставленной задачи была выбрана модель нейронной сети прямого распространения (*feed-forward neural network*). В построенной модели используется трехслойная нейронная сеть прямого распространения с одним скрытым слоем. В ИНС выбранной структуры элементы (нейроны) между собой никак не связаны, но связаны с нейронами предыдущего и следующего слоев (рис. 5, см. четвертую сторону обложки). Текущее состояние нейрона определяется как взвешенная сумма его входов. Выход нейрона — это функция его состояния (*функция активации*), вычисляющая его выходное значение.

Текущее состояние нейрона определяется как взвешенная сумма его входов. Выход нейрона — это функция его состояния (функция активации), вычисляющая его выходное значение. Методом подбора в качестве функции активации была выбрана сигмоидальная функция — гиперболический тангенс (*tansig*):

$$f_a(x) = \frac{\mathbf{e}^x - \mathbf{e}^{-x}}{\mathbf{e}^x + \mathbf{e}^{-x}}.$$

В результате должна минимизироваться сумма квадратов отклонений значений, полученных в результате работы нейросети, от экспериментальных данных. Таким образом, минимизируется следующий функционал:

$$F(\overline{u},\overline{w}) = \sum_{l=1}^{L}\sum_{k=1}^{M}\left[Y - f\left(\overline{x},\overline{a}\right)\right]^2, \qquad (1)$$

где $f\left(\overline{x},\overline{a}\right)$ — вычисляемая функция, которая получается в результате работы нейронов всех слоев; $\overline{a} \in R^{NN} = [v, w]$ — вектор оцениваемых параметров; M — число экспериментов.

Алгоритм формирования нейронной сети приведен ниже.

Шаг 1. На первом этапе осуществляется ввод входных данных.

Шаг 2. Инициализация:

$$v_{i,\,j} = w_{j,\,l} \in [-1,\ 1];$$

Шаг 3. Нормализация x, y:

$$\tilde{x}_{k,i} = \frac{2\left(x_{k,i} - x_{\min i}\right)}{x_{\max i} - x_{\min i}} - 1;\ \tilde{y}_{k,l} = \frac{2\left(y_{k,l} - y_{\min l}\right)}{y_{\max l} - y_{\min l}} - 1.$$

После работы нейронной сети параметры элементов матриц X, Y восстанавливаются:

$$y_{k,i} = y_{\min i} + \frac{\left(\tilde{y}_{k,i} + 1\right)\left(y_{\max i} - y_{\min i}\right)}{2};$$

$$x_{k,i} = x_{\min i} + \frac{\left(\tilde{x}_{k,i} + 1\right)\left(x_{\max i} - x_{\min i}\right)}{2}.$$

Шаг 4. Вычисление функции $f\left(\overline{x},\overline{a}\right)$:

независимые входные переменные $\tilde{x}_{k,i}$ суммируются с весами и передаются вышестоя-

щему слою (скрытый слой). Каждая единица скрытого слоя принимает эту сумму, обрабатывает ее функцией активации и передает всем единицам в слой выше:

$$z_i = u_{0,j} + \sum_{i=1}^{N} \tilde{x}_i u_{ij};$$

$$z_j = f(z_i, a).$$

Каждая выходная единица суммирует эти взвешенные входные сигналы и обрабатывает их функцией активации:

$$y_l = w_{0,l} + \sum_{j=1}^{P} z_j w_{j,l};$$

$$y_l = f(y_l, a).$$

Третий этап: обучение нейронной сети

Главное свойство нейросетей — способность к обучению. В процессе обучения нейронная сеть способна выявлять сложные зависимости между входными и выходными данными, а также выполнять обобщение. Это значит, что в случае успешного обучения сеть вернет верный результат на основании данных, которые отсутствовали в обучающей выборке, а также на основании неполных и/или "зашумленных", частично искаженных данных. Из способности к обобщению и выделению скрытых зависимостей между входными и выходными данными следует способность ИНС к прогнозированию. После обучения сеть способна предсказать будущее значение некоторой последовательности на основе нескольких предыдущих значений.

Обучение сети прямого распространения проводится методом обратного распространения ошибки таким образом, чтобы минимизировать среднеквадратическую ошибку отклика сети на обучающей выборке. В данном алгоритме выходные значения сравниваются с точными результатами для вычисления значения предопределенной функции ошибки. Используя эту информацию, алгоритм корректирует вес каждого соединения, чтобы уменьшить значение функции ошибки на некоторую величину.

При обучении набор исходных данных делят на две части — обучающую выборку и тестовые данные. Обучающие данные подаются сети для обучения, а проверочные используются для расчета ошибки сети. Для данной модели на вход подавались исходные значения по 43 параметрам. При этом все данные делятся в соотношении 4:1, где 4 части (80 %) отходят на обучение и 1 часть (20 %) отводится на тестирование. Цель обучения состоит в том, чтобы при уменьшении ошибки на проверочных данных сетью выполнялось обобщение.

Четвертый этап: апробация нейронной сети

Тестирование качества обучения ИНС проводилось на 20 % реальных исходных данных, которые не участвовали в ее обучении. Результаты моделирования представлены (рис. 6, см. четвертую сторону обложки).

График демонстрирует результаты прогнозирования нейронной сети (красные) в сравнении с экспериментальными данными (синие). Для оценки эффективности прогнозирования анализировались значения следующих параметров:

- Mean Relative Error (MRE) — относительная ошибка, которая показывает, насколько велика абсолютная ошибка по сравнению с общим размером тестируемых данных. Полученный результат MRE = 2,7931 %;
- KL — расстояние Кульбака — Лейблера, которое показывает потерю информации при замене истинного распределения на расчетное. Чем меньше значение KL, тем лучше совпадение распределений исходного и расчетного векторов. Полученный результат KL = 0,16758;
- R^2 — коэффициент детерминации для модели принимает значения от 0 до 1. Чем ближе значение коэффициента к 1, тем сильнее зависимость. Для приемлемых моделей предполагается, что коэффициент детерминации должен быть не меньше 0,5. Полученный результат $R^2 = 0{,}64174$.

Выводы

Проверка технологического процесса производства стирола с использованием нейросетевых технологий осуществлялась на примере известных входных и выходных данных и показала, что нейронная сеть дает аналогичные имеющиеся результаты с наперед заданной точностью. Для более точного прогнозирования вышеописанные этапы построения математической модели могут потребовать корректировки и экспериментального подбора характеристик сети (число скрытых слоев и нейронов, выбора функции активации). В связи с этим полученные на данной стадии результаты моделирования процесса производства стирола при известных входных параметрах, несмотря на вполне удовлетворительные результаты, необходимо рассматривать как предварительные.

Список литературы

1. **Сенаторов М. Ю., Левин С. Е., Нагибин С. Я.** Дистанционный контроль производственной безопасности топливно-энергетического комплекса // XVI Междунар. науч.-практ. конф. "Технические науки — от теории к практике". Санкт Петербург, 23 января 2017 г.

2. **Об Основах** государственной политики Российской Федерации в области промышленной безопасности на период до 2025 года и дальнейшую перспективу. Указ Президента Российской Федерации от 06.05.2018 № 198.

3. **Николенко С., Кадурин А., Архангельский Е.** Погружение в мир нейронных сетей. СПб.: Питер, 2018. 480 с.

4. **Anil K. Jain, Jianchang Mao, Mohiuddin K. M.** Artificial Neural Networks: A Tutorial, IEEE Computer, 1996, vol. 29, N. 3, pp. 31—44.

S. E. Levin, First Deputy General Director, e-mail: levin@pkcc-ps.ru,
Ya. N. Okrent, Chief Researcher, e-mail: namaste2003@gmail.com,
Russian Corporation of Means of Communication — Software Systems, Moscow, 105005, Russian Federation
S. Ya. Nagibin, Ph. D., Professor, e-mail: nsy7@rambler.ru,
N. E. Balakirev, Ph. D., Associate Professor, e-mail: balakirev1949@yandex.ru
Moscow Aviation Institute (National Research University), Moscow, 121552, Russian Federation

Mathematical Model of the Technological Process of Styrene Production

The article presents a mathematical model of the functioning of the technological process of styrene production using neural network technologies. The use of a direct propagation neural network with one hidden layer trained on an experimental sample is considered. The algorithm for the formation of a neural network is proposed. The model is implemented as a software module. The results of forecasting the process of styrene production on real data and recommendations on the use of the developed model in the process of assessing the industrial safety of highly hazardous production processes are presented.

Keywords: *industrial safety, artificial neural networks, mathematical model of styrene production, normalization of parameters, neural network training, method of back propagation of error, neural network formation algorithm, evaluation of prediction results*

DOI: 10.17587/it.25.572-576

References

1. **Senatorov M., Levin S., Nagibin S.** Distance control of process safety of fuel and energy complex, *XVI International research and practical conference on "Engineering sciences — from theory to practice"*, 2017, pp. 16—24 (in Russian).

2. **On fundamentals** of the Russian Federation state policy in the field of industrial safety for the period up to 2025 and further prospects, Russian Federation Presidential Decree as of 06.05.2018 No. 198 (in Russian).

3. **Nikolenko S., Kadurin A., Arkhangelsky E.** Diving into the world of neural networks, St. Petersburg, Piter, 2018, 480 p. (in Russian).

4. **Anil K. Jain, Jianchang Mao, Mohiuddin K. M.** *Artificial Neural Networks: A Tutorial*, IEEE Computer, 1996, vol. 29, no. 3, pp. 31—44.

Адрес редакции:

107076, Москва, Стромынский пер., 4

Телефон редакции журнала **(499) 269-5510**
E-mail: it@novtex.ru

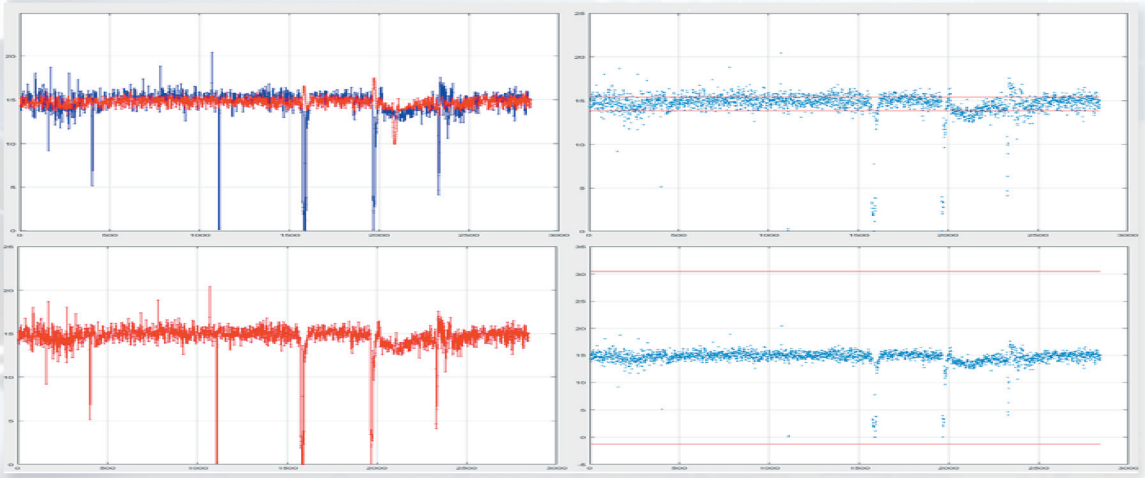
Технический редактор *Е. В. Конова*.
Корректор *З. В. Наумова*.
Сдано в набор 09.07.2019. Подписано в печать 27.08.2019. Формат 60×88 1/8. Бумага офсетная.
Усл. печ. л. 8,86. Заказ IT919. Цена договорная.

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

Рисунок к статье С. Е. Левина, Я. Н. Окрента, С. Я. Нагибина, Н. Е. Балакирева

«МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ТЕХНОЛОГИЧЕСКОГО ПРОЦЕССА ПРОИЗВОДСТВА СТИРОЛА»



а) *б)*

Рис. 4. Демонстрация эффекта сглаживания при различных значениях разброса на примере показателя «Расход стирола-ректификата от Н-327/1,2 в парк. Поз 23140»: *а* – σ = 0,5; *б* – σ =10

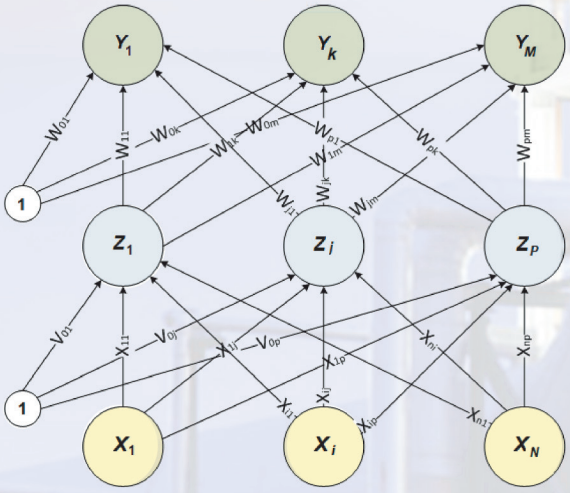


Рис. 5. ИНС прямого распространения с одним скрытым слоем ($Y = [y_{(k,l)}] \in R^{(M \times L)}$ – матрица значений зависимых переменных; $\overline{Z} = \left[z_j \right] \in R^P$ – вектор нейронов скрытого слоя; $X = \left[x_{k,i} \right] \in R^{M \times N}$ – матрица независимых переменных; $W = \left[w_{j,l} \right] \in R^{(P+1) \times L}$ – матрица оцениваемых параметров зависимых переменных; N – число независимых переменных; L – число зависимых переменных; P – число нейронов в первом скрытом слое; $V = \left[v_{i,j} \right] \in R^{(N+1) \times P}$ – матрица оцениваемых параметров первого скрытого слоя; $N_p = (N+1) \times P + (P+1) \times L$ – число оцениваемых параметров)

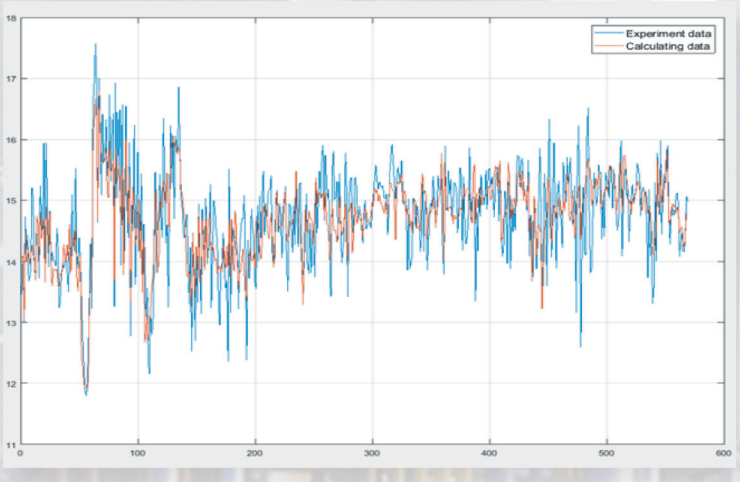


Рис. 6. Результаты работы обученной ИНС

ISSN 1684-6400. Информационные технологии. 2019. Том 25. № 9. 513–576. Индекс: 72656 (Объединенный каталог «Пресса России»)