DOI 10.17587/issn.1684-6400

ISSN 1684-6400

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Том 26

2020

№ 3

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 И. В. Лобова, В. Г. Готмана

«АДАПТИВНАЯ БЕСШОВНАЯ ПОТОКОВАЯ ТРАНСЛЯЦИЯ В РЕАЛЬНОМ ВРЕМЕНИ НАД ПРОТОКОЛОМ HTTP МЕТОДОМ ОПЕРЕЖАЮЩЕЙ ЗАГРУЗКИ»

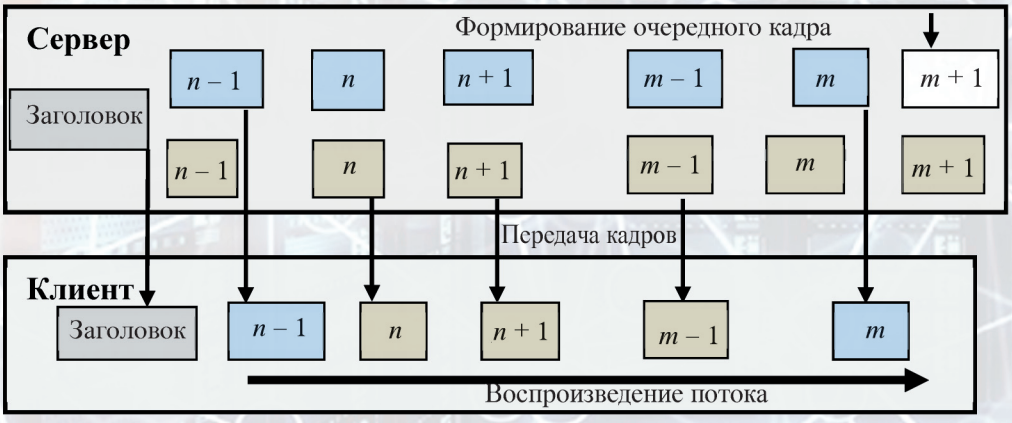


Рис. 1. Схема формирования клиенту видеопотока переменного битрейта

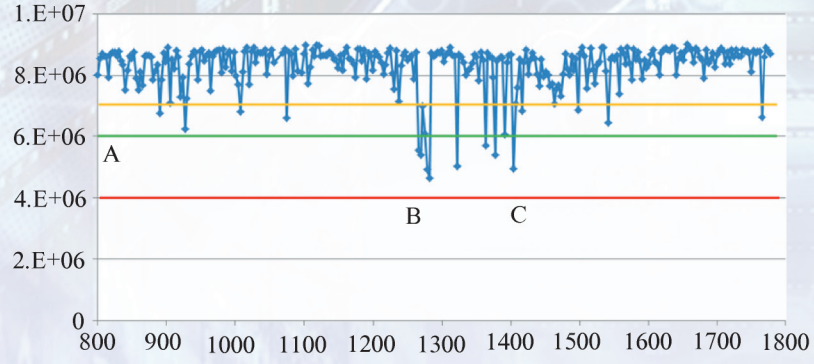


Рис. 2. Типичная картина временного тренда эффективной пропускной способности канала передачи данных (байт/с), $\Delta t_{эпс}$ = 500 мс, измерения проводились каждые 3,5 с

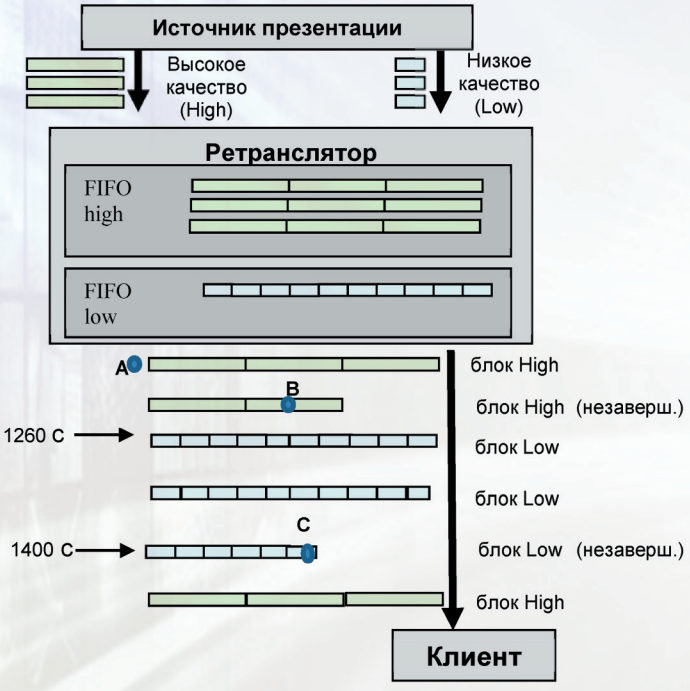


Рис. 3. Схема переключения качества потока, иллюстрирующая ситуацию рис. 2

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Том 26
2020
№ 3

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. 26
2020
No. 3

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"

МОДЕЛИРОВАНИЕ
MODELING

УДК 004.051 DOI: 10.17587/it.26.131-137

В. Н. Гридин[1], науч. руководитель, д-р техн. наук, проф., e-mail: info@ditc.ras.ru,
В. И. Анисимов[1, 2], гл. науч. сотр., д-р техн. наук, проф., e-mail: info@ditc.ras.ru,
[1] Центр информационных технологий в проектировании РАН,
[2] Санкт-Петербургский государственный электротехнический университет

Моделирование нелинейных систем на основе методов декомпозиции[1]

Рассматриваются методы повышения эффективности процессов моделирования нелинейных систем на базе их математического описания в виде блочно-диагональной окаймленной структуры. Устанавливается методика организации вычислительного процесса на основе технологии расчета слабосвязанных схем по частям. Проводится сравнительная оценка возможных подходов к организации итерационных вычислительных процессов при моделировании больших слабосвязанных систем на основе методов декомпозиции. Показывается, что путем перехода от сквозного итерационного процесса к автономным итерационным процессам и последующего уточнения переменных связи можно в несколько раз сократить общее число выполняемых вычислительных операций.

Ключевые слова: *декомпозиция, моделирование, нелинейная система, окаймленная структура, итерационные процессы, сквозной и автономный итерационные процессы*

Введение

Если моделируемая система имеет слабосвязанную иерархическую структуру, то для решения задачи моделирования нелинейных систем наиболее эффективным способом организации вычислительных процессов является декомпозиция исходной системы на ряд подсистем с использованием методики расчета слабосвязанной системы по частям. Такой подход может существенно повысить эффективность вычислительных процессов и привести требуемые для моделирования больших систем вычислительные ресурсы в соответствие с реальными возможностями. Для решения задачи декомпозиции моделируемая система по определенным правилам расчленяется на некоторое число малых подсистем, и на основе этого строятся топологические модели исходной системы. Для каждой из подсистем в отдельности проводится анализ и преобразование описания, общее решение получается путем соединения полученных частных решений для подсистем. При таком подходе не требуется составления полной системы уравнений, достаточно последовательно формировать и обрабатывать уравнения для ее подсистем, которые могут быть сделаны настолько малыми, насколько это практически целесообразно. Достоинства методов декомпозиции проявляются в наибольшей степени при анализе сложных систем и сводятся к сокращению затрат машинного времени, что имеет большое значение при построении распределенных сервис-ориентированных систем автоматизации схемотехнического проектирования [1—8].

При использовании декомпозиционного подхода минимизируется число обменов между оперативной и внешней памятью, если такие обмены приходится выполнять из-за нехватки требуемого объема оперативной памяти. При решении задачи моделирования возможна организация как последовательного вычислительного процесса на автономном компьютере, так и организация параллельных вычислений в локальной или глобальной сети, где компьютер каждого узла сети осуществляет формирование и обработку данных, связанных только с отдельной подсистемой. При этом для решения задач моделирования нелинейных систем на основе методов декомпозиции возможна организация как сквозных, так и автономных итерационных вычислительных процессов. Выбор способа организации итерационного процесса определяется специфическими особенностями моделируемой системы, зависящими от степени концентрации нелинейных элементов в ее отдельных подсистемах.

[1] Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта 18-07-00082а.

Математическое описание нелинейных систем

Компонентные уравнения многополюсных компонентов нелинейной системы в общем случае могут быть записаны в виде $p_i = f(..., q_i, ...)$. С учетом характера зависимых p_i и независимых q_i переменных все полюса и порты многополюсного компонента могут быть разделены на две группы — группу y и группу z. Если в качестве зависимой переменной p_i выбирается токовая переменная i_i, то i-й полюс входит в группу y-полюсов, если же выбирается потенциальная переменная u_i, то i-й полюс входит в совокупность z-полюсов [9].

В общем случае моделируемая система может содержать как линейные, так и нелинейные многополюсники.

Если многополюсный компонент является линейным, то уравнения y-полюсов и z-полюсов имеют вид

$$I_{y1} = Y_{m1}U_{y1} + B_{m1}I_{z1} + J_{m1}; \quad (1)$$

$$U_{z1} = M_{m1}U_{y1} + Z_{m1}I_{z1} + E_{m1}, \quad (2)$$

где I_{y1}, U_{y1}, I_{z1}, U_{z1} — векторы полюсных переменных линейного многополюсника; Y_{m1}, B_{m1}, M_{m1}, Z_{m1} — матрицы линейных параметров компонента; J_{m1}, E_{m1} — векторы задающих источников многополюсника.

Объединяя уравнения (1) и (2) в одно матричное уравнение, получим матричное уравнение линейных многополюсников

$$P_1 = W_{m1}Q_1 + S_{m1}, \quad (3)$$

где $P_1 = [I_{y1}^{\mathrm{T}}, U_{z1}^{\mathrm{T}}]^{\mathrm{T}}$, $Q_1 = [U_{y1}^{\mathrm{T}}, I_{z1}^{\mathrm{T}}]^{\mathrm{T}}$ — векторы зависимых и независимых переменных линейного многополюсного компонента;

$W_{m1} = \begin{pmatrix} Y_{m1} & B_{m1} \\ M_{m1} & Z_{m1} \end{pmatrix}$; $S_{m1} = \begin{pmatrix} J_{m1} \\ E_{m1} \end{pmatrix}$ — матрица параметров и задающий вектор линейных многополюсных компонентов.

Компонентные уравнения нелинейных многополюсников можно записать в виде $i_i = f_y(..., u_k, ..., i_l, ...)$ для y-полюсов и $u_j = f_z(..., u_k, ..., i_l, ...)$ — для z-полюсов, где u_k и i_l — независимые переменные y-полюсов и z-полюсов, или в матричной форме

$$\begin{pmatrix} I_{y2} \\ U_{z2} \end{pmatrix} = \begin{pmatrix} F_y(U_{y2}, I_{z2}) \\ F_z(U_{y2}, I_{z2}) \end{pmatrix}.$$

Здесь I_{y2}, I_{z2} — векторы токовых переменных y- и z-полюсов; U_{y2}, U_{z2} — векторы потенциальных переменных y- и z-полюсов; $F_y(U_{y2}, I_{z2})$ и $F_z(U_{y2}, I_{z2})$ — вектор-функции нелинейных многополюсников.

Для описания нелинейных компонентов отождествим их вектор зависимых токовых переменных I_{y2} с вектором некоторых фиктивных источников тока $J_{m2} = F_y(U_{y2}, I_{z2})$, а вектор их зависимых потенциальных переменных U_{z2} — с вектором некоторых фиктивных источников напряжения $E_{m2} = F_z(U_{y2}, I_{z2})$. Тогда все нелинейные компоненты можно описать уравнением вида

$$P_2 = F(Q_2) = S_{m2}, \quad (4)$$

где

$$F(Q_2) = \begin{pmatrix} F_y(U_{y2}, I_{z2}) \\ F_z(U_{y2}, I_{z2}) \end{pmatrix};$$

$$S_{m2} = \begin{pmatrix} J_{m2} \\ E_{m2} \end{pmatrix}; \; P_2 = \begin{pmatrix} I_{y2} \\ U_{z2} \end{pmatrix}; Q_2 = \begin{pmatrix} U_{y2} \\ I_{z2} \end{pmatrix}.$$

Объединяя уравнения (3) и (4) в одно уравнение, получим объединенное компонентное уравнение многополюсных компонентов:

$$P = W_m Q + S_m, \quad (5)$$

где

$$P = \begin{pmatrix} P_1 \\ P_2 \end{pmatrix}; Q = \begin{pmatrix} Q_1 \\ Q_2 \end{pmatrix}; W_m = W_{m1}; S_m = \begin{pmatrix} S_{m1} \\ S_{m2} \end{pmatrix}.$$

Топологические уравнения моделируемой системы в расширенном узловом базисе [9] можно записать в виде

$$T_P P + T_X X = 0; \quad (6)$$

$$Q = T_P^{\mathrm{T}} X, \quad (7)$$

где

$$X = \begin{pmatrix} V \\ I_z \end{pmatrix}; T_P = \begin{pmatrix} A_y \\ 1 \end{pmatrix}; T_X = \begin{pmatrix} A_z \\ -A_z^{\mathrm{T}} \end{pmatrix}.$$

Здесь A_y, A_z — блочные матрицы инциденций A для y- и z-полюсов.

Исключая из уравнений (5), (6) и (7) векторы P и Q, получим уравнение нелинейной системы в виде

$$WX + S = 0, \quad (8)$$

где $W = T_P W_m T_P^{\mathrm{T}} + T_X$, $S = T_P S_m$.

Разделим далее общий задающий вектор S на линейную S_1 и нелинейную $S_2 = S_2(X)$ составляющие. С этой целью представим матрицу T_P в блочной форме, выделив в ней блоки линейной T_{P1} и нелинейной T_{P2} частей системы:

$$T_P = (T_{P1} \quad T_{P2}).$$

С учетом блочной структуры матрицы T_P окончательно получим уравнение нелинейных систем в виде

$$\Phi(X) = W_1 X + S_1 + S_2(X) = 0, \qquad (9)$$

где $W_1 = T_{P1} W_{m1} T_{P1}^{\mathrm{T}} + T_X$, $S_1 = T_{P1} S_{m1}$, $S_2(X) = T_{P2} S_{m2} = T_{P2} F(Q_2)$.

Таким образом, нелинейные свойства моделируемой системы в нелинейном уравнении (9) определяются видом функциональных зависимостей $P_2 = F(Q_2)$ нелинейных компонентов.

В общем случае задача расчета нелинейной системы сводится к решению уравнения (9), которое выполняется методом Ньютона—Рафсона согласно соотношению [10—13]

$$M^i \Delta X^{i+1} + \Phi(X^i) = 0, \qquad (10)$$

где $M^i = \dfrac{\partial \Phi(X)}{\partial X}\Big| X = X^i$, $\Delta X^{i+1} = X^{i+1} - X^i$.

Непосредственное использование уравнения (10) связано с известными техническими трудностями формирования в общем виде функционала $\Phi(X^i)$ и вычисления матрицы производных M^i.

Поэтому целесообразно использовать схемотехническую интерпретацию метода Ньютона—Рафсона. С этой целью продифференцируем нелинейное уравнение (9), в результате чего получим выражение для вычисления матрицы производных

$$M^i = \frac{\partial \Phi(X)}{\partial X}\Big|_{(X = X^i)} = W_1 + \frac{\partial S_2(X)}{\partial X}\Big|_{(X = X^i)}.$$

Учитывая, что $S_2(X) = T_{P2} S_{m2}$, $S_{m2} = P_2 = F(Q_2)$ и $Q_2 = T_{P2}^{\mathrm{T}} X$, можно записать:

$$\frac{\partial S_2(X)}{\partial X}\Big|_{(X = X^i)} = T_{P2} \frac{\partial P_2}{\partial Q_2}\Big|_{(Q_2 = Q_2^i)} \frac{\partial Q_2}{\partial X} = T_{P2} \frac{\partial P_2}{\partial Q_2}\Big|_{(Q_2 = Q_2^i)} T_{P2}^{\mathrm{T}}.$$

Введем обозначение:

$$W_{m2}^i = \frac{\partial P_2}{\partial Q_2}\Big|_{(Q_2 = Q_2^i)},$$

где W_{m2}^i — матрица параметров нелинейных компонентов, линеаризованных в точке $Q_2 = Q_2^i$.

Тогда можно записать:

$$M^i = W_1 + T_{P2} W_{m2}^i T_{P2}^{\mathrm{T}} = W_1 + W_2^i, \qquad (11)$$

где

$$W_2^i = T_{P2} W_{m2}^i T_{P2}^{\mathrm{T}}.$$

Таким образом, матрица M^i может быть найдена без вычисления матрицы производных $\dfrac{\partial \Phi(X)}{\partial X}$.

Подстановка значения M^i в уравнение Ньютона—Рафсона (10) дает

$$(W_1 + W_2^i)\Delta X^{i+1} + \Phi(X^i) = 0.$$

Подставляя сюда значение $\Delta X^{i+1} = X^{i+1} - X^i$, а также выражение для функционала $\Phi(X^i) = W_1 X^i + S_1 + S_2(X^i)$ и учитывая очевидные соотношения $S_2(X^i) = T_{P2} P_2^i$, $Q_2^i = T_{P2}^{\mathrm{T}} X^i$, $W_2^i = T_{P2} W_{m2}^i T_{P2}^{\mathrm{T}}$, после несложных преобразований получим уравнение для схемотехнической интерпретации метода Ньютона—Рафсона в виде

$$(W_1 + W_2^i) X^{i+1} + S_1 + S_2^i = 0, \qquad (12)$$

где

$$W_1 = T_{P1} W_{m1} T_{P2}^{\mathrm{T}} + T_x, \quad W_2^i = T_{P2} W_{m2}^i T_{P2}^{\mathrm{T}}; \qquad (13)$$

$$S_1 = T_{P1} S_{m1}, \quad S_2^i = T_{P2} S_{m2}^i. \qquad (14)$$

Входящие в эти выражения матрицы $W_{m2}^i = \dfrac{\partial P_2}{\partial Q_2}\Big|_{(Q_2 = Q_2^i)}$ и $S_{m2}^i = P_2^i - W_{m2}^i Q_2^i$ являются линеаризованными в точке i матрицами нелинейных многополюсников и могут быть легко вычислены на основании выполнения линеаризации характеристик используемых нелинейных компонентов.

Расчет нелинейных систем на основе декомпозиции и расчета по частям

Пусть имеется нелинейная система, описываемая уравнением (9), расчет которой выполняется на основе схемотехнической интерпретации метода Ньютона—Рафсона согласно уравнению

$$W^i X^{i+1} + S^i = 0, \qquad (15)$$

где $W^i = W_1 + W_2^i$, $S^i = S_1 + S_2^i$, при этом в соответствии с методом схемотехнической интерпретации матрицы W_1, W_2^i, S_1, S_2^i определяются соотношениями (13) и (14).

Если моделируемая система имеет слабосвязанную структуру, то наиболее эффективным способом организации вычислительных процессов является декомпозиция исходной системы на ряд составляющих подсистем с использованием декомпозиционного подхода расчета по частям [14—18]. При этом сложная система по определенным правилам расчленяется на некоторое число малых подсистем, и на основе этого строятся топологические модели исходной системы. Для каждой из подсистем в отдельности проводится анализ и отыскивается решение, общее

решение получается путем соединения полученных частных решений для подсистем. Такой подход может существенно повысить эффективность вычислительных процессов, реализующих моделирование системы, и привести требуемые для решения задачи ресурсы в соответствие с имеющимися реальными возможностями.

Единственным ограничением для реализации декомпозиционного подхода к моделированию больших систем является требование выполнения условий слабосвязанности отдельных подсистем исходной системы. Условия слабосвязанности требуют, во-первых, отсутствия индуктивных связей между отдельными подсистемами. Во-вторых, для каждого управляемого источника должно выполняться требование концентрации внутри каждой подсистемы его управляющих и управляемых переменных.

Разбиение исходной системы на отдельные подсистемы может быть выполнено произвольным образом с учетом выполнения сформулированных выше условий слабосвязанности. С практической точки зрения при этом обычно удобно включать в каждую подсистему компоненты, реализующие некоторую определенную функциональную задачу. Число подсистем выбирается исходя из общей размерности моделируемой системы.

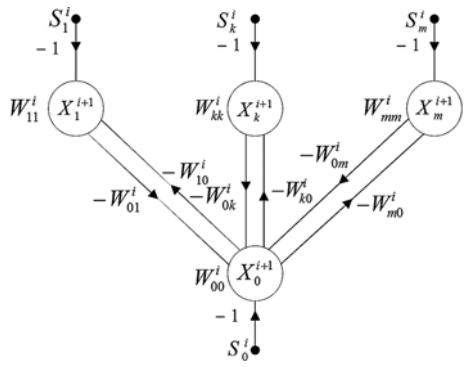
Все переменные разделенной на подсистемы исходной структуры могут быть разделены на две группы. В первую группу входят внутренние переменные подсистем, во вторую группу — переменные связи в узлах соединения подсистем. Соответствующим выбором нумерации внутренних переменных отдельных подсистем и переменных связи в узлах соединения подсхем можно обеспечить представление исходного матричного уравнения моделируемой системы в виде, содержащем матрицу с окаймленной блочно-диагональной структурой. Такая матрица будет содержать окаймляющую блочную строку и окаймляющий блочный столбец, а все остальные ненулевые элементы этой матрицы будут сконцентрированы в диагональных матричных блоках. При этом каждый такой блок общей матрицы определяется только структурой и значениями параметров одной из подсистем.

Для каждого узла схемы можно ввести узловую переменную, определяющую состояние этого узла, и из всех введенных узловых переменных v_p можно образовать вектор узловых потенциалов $X_0 = V_0 = [..., v_p, ...]^{\mathrm{T}}$. В каждой подсистеме можно выделить внутренние переменные x_i и составить для каждой подсистемы вектор $X_k = [..., x_i, ...]^{\mathrm{T}}$. При этом общий вектор всех переменных системы будет иметь вид $X = [X_1^{\mathrm{T}}, \ldots, X_k^{\mathrm{T}}, \ldots, X_m^{\mathrm{T}}, X_0^{\mathrm{T}}]^{\mathrm{T}}$, где вектор $X_0 = V_0$ будет включать в себя узловые переменные узлов, а векторы X_k — переменные k-й подсистемы.

С учетом принятой структуры вектора X матричное уравнение всей системы (15) в расширенном базисе узловых потенциалов [9, 19] может быть записано в виде

$$\begin{pmatrix} W_{11}^i & & & & W_{10}^i \\ & \ldots & & & \ldots \\ & & W_{kk}^i & & W_{k0}^i \\ & & & \ldots & \ldots \\ & & & W_{mm}^i & W_{m0}^i \\ W_{01}^i & \ldots & W_{0k}^i & \ldots & W_{00}^i \end{pmatrix} \begin{pmatrix} X_1^{i+1} \\ \ldots \\ X_k^{i+1} \\ \ldots \\ X_m^{i+1} \\ X_0^{i+1} \end{pmatrix} + \begin{pmatrix} S_1^i \\ \ldots \\ S_k^i \\ \ldots \\ S_m^i \\ S_0^i \end{pmatrix} = 0. \quad (16)$$

Для построения и преобразования описания моделируемой схемы на основании декомпозиционного подхода особенно удобно использовать топологическое описание схемы с помощью обобщенных сигнальных графов [20]. Обобщенный сигнальный граф отображает систему уравнений, записанных в обобщенной причинно-следственной форме, и содержит взвешенные вершины, среди которых могут быть и вершины с нулевым весом. Такое описание помимо наглядного представления о структуре связей между подсистемами позволяет снять проблему выбора последовательности нумерации внутренних переменных подсистем и переменных узлов связи. Сопоставленный матричному уравнению (16) обобщенный сигнальный граф приведен на рисунке. Показанная на рисунке структура обобщенного сигнального графа наглядно демонстрирует соответствующую структуру математического описания слабосвязанной системы, в которой взаимодействие между отдельными подсистемами реализуется только через узловые переменные связи. Вследствие слабосвязанности моделируемой системы оказывается возможной независимая обработка математического описания отдельных подсистем и уравнений связи.



Обобщенный сигнальный граф слабосвязанной нелинейной системы

Топологическое представление описания моделируемой схемы в виде обобщенного сигнального графа предоставляет удобные возможности для анализа структуры этого графа и вычисления определителя графа Δ на основании рассмотрения всех его элементарных графов g_i, каждый из которых содержит совокупность некасающихся контуров и взвешенных вершин (включая взвешенные вершины с нулевым весом).

При этом определитель графа Δ может быть вычислен на основании топологической формулы $\Delta = \Sigma\delta_i$, где δ_i — определитель элементарного графа, равный произведению весов входящих в него некасающихся контуров и взвешенных вершин, и суммирование осуществляется по всем возможным элементарным графам [20].

Итерационные процессы решения уравнения (16) на основе декомпозиции могут быть реализованы двумя способами — путем организации сквозного вычислительного процесса для всей системы или автономных вычислительных процессов для отдельных подсистем с последующим процессом расчета переменных связи системы.

Сквозной вычислительный процесс является наиболее общим типом процесса и может быть реализован в соответствии со следующим алгоритмом:

1. Ввод начальных значений $X_k^i = X_k^0$, $X_0^i = X_0^0$.
2. Начало цикла $k = \overline{1, m}$.
3. Линеаризация нелинейных компонентов в точке X_k^i.
4. Формирование линеаризованных матриц подсистем W_{kk}^i, W_{k0}^i, W_{0k}^i, W_{00}^i, S_k^i, S_0^i.
5. Преобразование W_{kk}^i к единичной матрице методом Гаусса—Жордано и расчет новых значений k-й блочной строки $\bar{W}_{k0}^i$, $\bar{S}_k^i$.
6. Расчет поправок $\Delta_k W_{00} = \bar{W}_{k0} W_{0k}$ и $\Delta_k S_0 = W_{0k}\bar{S}_k$ и коррекция матриц уравнения узлов связи с учетом поправок $\overline{W_{00}} = W_{00} - \sum_{k=1}^{m} \Delta_k W_{00}$, $\overline{S_0} = S_0 - \sum_{k=1}^{m} \Delta_k S_0$.
7. Окончание цикла $k = \overline{1, m}$.
8. Решение уравнения $\bar{W}_{00}^i X_0^{i+1} + \bar{S}_0^i = 0$ для переменных связи.
9. Начало цикла $k = \overline{1, m}$.
10. Расчет вектора внутренних переменных подсистем $X_k^{i+1} = -\bar{W}_{k0}^i X_0^{i+1} - \bar{S}_k^i$.
11. Окончание цикла $k = \overline{1, m}$.
12. Вычисление нормы $N = \|X^{i+1} - X^i\|$.
13. Если $N > \varepsilon$, то $X_k^i = X_k^{i+1}$, $X_0^i = X_0^{i+1}$ и возврат к шагу 2, иначе — вывод векторов внутренних переменных X_k^{i+1} и переменных связи X_0^{i+1}.

Организация автономных вычислительных процессов целесообразна в тех случаях, когда степень нелинейности отдельных подсистем значительно отличается, особенно когда подавляющее число подсистем являются линейными, а нелинейные элементы могут быть сконцентрированы в незначительном числе подсистем. В частном случае все нелинейные свойства системы можно сконцентрировать в одной из подсистем, при этом все остальные подсистемы будут содержать только линейные элементы.

Автономные вычислительные процессы могут быть организованы согласно следующему алгоритму:

1. Ввод начальных значений $X_k^i = X_k^0$, $X_0^i = X_0^0$.
2. Начало цикла $k = \overline{1, m}$.
3. Линеаризация нелинейных компонентов в точке X_k^i.
4. Формирование линеаризованных матриц отдельных подсистем W_{kk}^i, W_{k0}^i, W_{0k}^i, W_{00}^i, S_k^i, S_0^i.
5. Преобразование W_{kk}^i к единичной матрице методом Гаусса—Жордано и расчет новых значений k-й блочной строки $\bar{W}_{k0}^i$, $\bar{S}_k^i$.
6. Расчет $X_k^{i+1} = -\bar{W}_{k0}^i X_0^{i+1} - \bar{S}_k^i$.
7. Вычисление нормы $N = \|X_k^{i+1} - X_k^i\|$.
8. Если $N > \varepsilon$, то $X_k^i = X_k^{i+1}$ и возврат к шагу 3, иначе переход к шагу 9.
9. Расчет поправок $\Delta_k W_{00} = \bar{W}_{k0} W_{0k}$ и $\Delta_k S_0 = W_{0k}\bar{S}_k$ и коррекция матриц уравнения узлов связи с учетом поправок $\overline{W_{00}} = W_{00} - \sum_{k=1}^{m} \Delta_k W_{00}$, $\overline{S_0} = S_0 - \sum_{k=1}^{m} \Delta_k S_0$.
10. Окончание цикла $k = \overline{1, m}$.
11. Решение уравнения $\bar{W}_{00}^i X_0^{i+1} + \bar{S}_0^i = 0$ для переменных узлов связи.
12. Начало цикла $k = \overline{1, m}$.
13. Уточнение вектора внутренних переменных подсистем $X_k^{i+1} = -\bar{W}_{k0}^i X_0^{i+1} - \bar{S}_k^i$.
14. Окончание цикла $k = \overline{1, m}$.
15. Вычисление нормы $N = \|X^{i+1} - X^i\|$.
16. Если $N > \varepsilon$, то, $X_0^i = X_0^{i+1}$ и возврат к шагу 2, иначе — вывод X_k^{i+1} и X_0^{i+1}.

Для оценки выигрыша от перехода к автономным вычислительным процессам рассмотрим случай, когда система разделена на 10 подсистем ($m = 10$). Пусть одна из подсистем требует для расчета 10 итераций, а остальные подсистемы — по 2 итерации, и расчет вектора переменных связи X_0^{i+1} также выполняется за две итерации. Тогда, если на одну итерацию требуется $N_{\text{оп}}$ операций, то для автономного вычислительного процесса получим общее число операций $M_{\text{авт}} = 30N_{\text{оп}}$, для сквозного — $M_{\text{скв}} = 110N_{\text{оп}}$. Таким образом, за счет перехода от сквозного итерационного процесса к автономным итерационным процессам с последующей организацией расчета переменных связи, оказывается возможным в несколько раз сократить общее число выполняемых операций.

Заключение

Рассмотренные методы повышения эффективности моделирования нелинейных систем на базе математического описания в виде блочно-диагональной окаймленной структуры существенно уменьшают время, необходимое для решения задач моделирования больших слабосвязанных систем. Сравнительная оценка возможных подходов к организации итерационных вычислительных процессов при моделировании больших слабосвязанных систем на основе декомпозиции позволяет сделать вывод о целесообразности перехода к организации автономных итерационных процессов. Такой переход особенно эффективен в случаях, когда степень нелинейности отдельных подсистем значительно отличается, особенно если подавляющее число подсистем являются линейными, а нелинейные элементы могут быть сконцентрированы в незначительном числе подсистем. При этом путем перехода от сквозного итерационного процесса к автономным итерационным процессам и последующей организации расчета переменных связи можно в несколько раз сократить общее число выполняемых вычислительных операций.

Однако в общем случае при произвольном характере распределения нелинейных компонентов по отдельным подсистемам моделируемой системы более целесообразным является использование сквозного итерационного процесса.

Список литературы

1. **Анисимов В. И., Гридин В. Н.** Методы построения схем автоматизированного проектирования на основе Интернет-технологий и компактной обработки разреженных матриц // Информационные технологии в проектировании и производстве. 2009. № 1. С. 3—7.

2. **Зеленухина В. А.** Разработка Интернет-ориентированных виртуальных лабораторий математического моделирования посредством разделения вычислительных и визуализационных задач // Информационные технологии. 2010. № 10. С. 22—29.

3. **Коваленко О. С., Курейчик В. М.** Обзор проблем и состояний облачных вычислений и сервисов // Известия ЮФУ. Технические науки. 2012. № 7. С. 146—153.

4. **Гридин В. Н., Дмитревич Г. Д., Анисимов Д. А.** Построение систем автоматизированного проектирования на основе Web-технологий // Информационные технологии. 2011. № 5. С. 23—27.

5. **Гридин В. Н., Дмитревич Г. Д., Анисимов Д. А.** Построение веб-сервисов систем автоматизации схемотехнического проектирования // Информационные технологии и вычислительные системы. 2012. № 4.

6. **Анисимов Д. А.** Методы построения систем автоматизации схемотехнического проектирования на основе вебсервисов // Известия СПбГЭТУ "ЛЭТИ" 2012. № 10. С. 56—61.

7. **Морган С.** Разработка распределенных приложений на платформе Microsoft.Net Framework. М.: Русская Редакция, 2008. 608 с.

8. **Цимбал А. А., Аншина М. Л.** Технология создания распределенных систем. СПб.: Питер, 2003. 576 с.

9. **Влах И., Сингхал К.** Машинные методы анализа и проектирования электронных схем: Пер. с англ. М.: Радио и связь, 1988. 560 с.

10. **Норенков И. П., Маничев В. Б.** Основы теории и проектирования САПР. М.: Высшая школа. 1990. 334 с.

11. **Тарасик В. П.** Математическое моделирование технических систем. Минск: Дизайн ПРО, 2004. 639 с.

12. **Гридин В. Н., Михайлов В. Б., Шустерман Л. Б.** Численно-аналитическое моделирование радиоэлектронных схем. М.: Наука, 2008. 339 с.

13. **Калабеков Б. А., Лапидус И. Ю., Малафеев В. М.** Методы автоматизированного расчета электронных схем в технике связи. М.: Радио и связь, 1990. 272 с.

14. **Крон Г.** Исследование сложных схем по частям — диакоптика: Пер с англ. М.: Наука, Главная редакция физико-математической литературы, 1972. 542 с.

15. **Баталов Б. В., Егоров Ю. Б., Русаков С. Г.** Основы математического моделирования больших интегральных схем на ЭВМ. М.: Радио и связь, 1982. 168 с.

16. **Анисимов В. И., Тарасова О. Б., Алмаасали С. А.** Организация вычислительных процессов при моделировании схем на основе методов диакоптики // Информационные технологии в проектировании и производстве. 2013. № 4. С. 14—17.

17. **Гридин В. Н., Анисимов В. И., Абухазим М. М.** Моделирование больших систем на основе методов декомпозиции и сжатия данных // Системы высокой доступности. 2015. № 4. С. 77—82.

18. **Гридин В. Н., Анисимов В. И., Абухазим М. М.** Методы моделирования систем на основе методов декомпозиции и компактной обработки разреженных матриц // Информационные технологии в проектировании и производстве. 2016. № 1. С. 3—8.

19. **Хайнеман Р.** PSPICE Моделирование работы электронных схем. М.: Издательство ДМК, 2005. 327 с.

20. **Анисимов В. И.** Топологический расчет электронных схем. Л.: Энергия, 1977. 238 с.

V. N. Gridin[1], Scientific Director, D. Sc., Professor, e-mail: info@ditc.ras.ru,
V. I. Anisimov[1, 2], Chief Researcher, D. Sc., Professor, e-mail: vianisimov@inbox.ru,
[1] Design information technologies Center Russian Academy of Sciences, Odintsovo, Russian Federation,
[2] Saint-Petersburg Electrotechnical University, Saint-Petersburg, Russian Federation

Modeling Non-Linear Systems Based on Decomposition Methods

Methods of increasing the efficiency of nonlinear systems modeling on the basis of their mathematical description in the form of a block-diagonal bordered structure are considered. The method of organization of computational process based on the technology of calculation of loosely coupled circuits by parts is established. The comparative estimation of possible approaches to the organization of iterative computational processes at modelling of the big weakly connected systems on the basis of methods of decomposition is carried out. It is shown that by moving from a through iterative process to autonomous iterative processes and further refinement of communication variables, it is possible to reduce the total number of performed computational operations several times.

Keywords: *decomposition, modeling, nonlinear system, bordered structure, iterative processes, through and autonomous iteration processes*

DOI: 10.17587/it.26.131-137

References

1. **Anisimov V. I., Gridin V. N.** *Informatsionnye Tekhnologii v Proyektirovanii i Proizvodstve* 2009, no. 1, pp. 3—7 (in Russian).
2. **Zelenukhina V. A.** *Informacionnye Tekhnologii*, 2010, no. 10, pp. 22—29 (in Russian).
3. **Kovalenko O. S., Kureichik V. M.** *Izvestiya YUFU. Tekhnicheskiye nauki,* 2012, no. 7, pp. 146—153 (in Russian).
4. **Gridin V. N., Dmitrevich G. D., Anisimov D. A.** *Informacionnye Tekhnologii,* 2011, no. 5, pp. 23—27 (in Russian).
5. **Gridin V. N., Dmitrevich G. D., Anisimov D. A.** *Informatsionnyye Tekhnologii i Vychislitel'nyye Sistemy,* 2012, no. 4 (in Russian).
6. **Anisimov D. A.** *Izvestiya SPbGETU "LETI",* 2012, no. 10, pp. 56—61 (in Russian).
7. **Morgan S.** Development of distributed applications on the platform Microsoft.Net Framework, Moscow, Russian Edition, 2008, 608 p. (in Russian).
8. **Tsimbal A. A., Anshina M. L.** Technology for creating distributed systems, St. Petersburg, Peter, 2003, 576 p. (in Russian).
9. **Vlach I., Singhal K.** Machine methods of analysis and design of electronic circuits, Moscow, Radio and communications, 1988, 560 p. (in Russian).
10. **Norenkov I. P., Manichev V. B.** Fundamentals of CAD theory and design, Moscow, Higher school, 1990, 334 p. (in Russian).
11. **Tarasik V. P.** Mathematical modeling of technical systems, Minsk, Design missile defense, 2004, 639 p. (in Russian).
12. **Gridin V. N., Mikhailov V. B., Shusterman L. B.** Numerical and analytical modeling of electronic circuits, Moscow, Nauka, 2008, 339 p. (in Russian).
13. **Kalabekov B. A., Lapidus I. Yu., Malafeev V. M.** Methods of automated calculation of electronic circuits in communication technology, Moscow, Radio and communications, 1990, 272 p. (in Russian).
14. **Cron G.** The study of complex schemes in parts — diakoptika, Moscow, Nauka, Main edition of the physics and mathematics literature, 1972, 542 p. (in Russian).
15. **Batalov B. V., Egorov Yu. B., Rusakov S. G.** Fundamentals of mathematical modeling of large computer integrated circuits, Moscow, Radio and communications, 1982, 168 p. (in Russian).
16. **Anisimov V. I., Tarasova O. B., Almaasali S. A.** *Informatsionnyye Tekhnologii v Proyektirovanii i Proizvodstve,* 2013, no. 4, pp. 14—17 (in Russian).
17. **Gridin V. N., Anisimov V. I., Abukhazim M. M.** *Sistemy Vysokoy Dostupnosti,* 2015, no. 4, pp. 77—82 (in Russian).
18. **Gridin V. N., Anisimov V. I., Abukhazim M. M.** *Sistemy Vysokoy Dostupnosti,* 2016, no. 1, pp. 3—8 (in Russian).
19. **Heineman R.** PSPICE Modeling the operation of electronic circuits, Moscow, Publishing house DMK, 2005, 327 p. (in Russian).
20. **Anisimov V. I.** Topological calculation of electronic circuits, Leningrad, Energy, 1977, 238 p. (in Russian).

УДК 004.02 DOI:10.17587/it.26.137-144

Ю. А. Зак, д-р техн. наук, e-mail: yuriy_zack@hotmail.com,
Аахен, Германия

Нечеткий линейный регрессионный анализ, учитывающий характер влияния входных факторов

Рассмотрено решение задач нечеткого регрессионного анализа в условиях, когда входные и выходная переменная представлены нормализованными нечеткими множествами с LR-представлением функции принадлежности самого общего вида, а коэффициенты регрессии — отрицательные или положительные действительные числа. Свободный член уравнения регрессии — нечеткое множество самого общего вида. Предусмотрены ограничения на установленную экспертами степень влияния некоторых входных факторов. Критерий аппроксимации — минимальное абсолютное значение средневзвешенной суммы абсолютных значений координат минимальных и максимальных значений абсцисс λ-сечений функций принадлежности нечетких множеств выходной переменной и ее оценки по fuzzy-регрессионной модели. Коэффициенты регрессии рассчитываются в результате решения некоторого подмножества задач линейного программирования с последующим выбором среди них решения с наилучшим значением критерия оптимальности.

Ключевые слова: *нечеткие регрессионные модели, fuzzy-множества, LR-представление функции принадлежности, сечения функций принадлежности, линейное программирование*

Введение

Классические методы регрессионного анализа работают только со статистическими данными, представленными действительными числами и находят широкое применение при построении математических моделей производственных, технических и экономических систем, а также в макроэкономике, социологии, политологии и медицине (см., например, [1, 2, 4]). Однако в ряде случаев отдельные показатели и параметры могут быть описаны только лингвистическими (например, плохой, удовлетворительный, хороший, превосходный и т. п.) либо булевыми, либо fuzzy-переменными, либо интервальными значениями. В работах [3—5] показано, как такие переменные могут быть представлены нечеткими множествами. В усло-

виях, когда в статистической выборке некоторые или все входные и (или) выходная переменная представлены нечисловой информацией, использование fuzzy-регрессионных моделей является эффективной альтернативой получения количественных зависимостей в случае установленных экспертами качественных закономерностей изучаемых явлений. Результатом расчета на основе таких моделей, как и параметрами модели (т. е. значениями входных и выходных переменных, либо коэффициентов уравнений модели), являются нечеткие множества с функцией принадлежности непрерывного вида. Эти результаты расчета определяют диапазон возможных значений выходной переменной и оценку (некоторый аналог вероятности) получения определяемого значения в пределах данного диапазона. Построение модели представляет собой в данном случае определение оптимальных в некотором смысле коэффициентов модели с учетом нечеткой информации об объекте и субъективных представлений исследователя об оценках адекватности построенной регрессионной модели.

1. Краткий обзор публикаций по построению нечетких регрессионных моделей

Известны три метода fuzzy-регрессионного анализа [4, 6, 7]:

а) нечеткая регрессия, основанная на критерии минимизации нечеткости [6, 7];

б) метод, получивший название FLSRA (fuzzy least-square regression analysis) [6—8], который, в свою очередь, имеет две разновидности, в одной из которых используется критерий максимальной совместимости, а в другой — критерий минимизации нечеткости;

в) регрессия интервала [4].

Все три метода могут в качестве исходной информации использовать как нечеткие множества, представленные функциями принадлежности, так и детерминированные данные.

В расчетах нечетких коэффициентов модели используются два критерия:

а) для всех расчетных данных принадлежность фактического значения выходной переменной к его нечеткой оценке должна быть не ниже некоторого значения, определяемого как уровень доверия;

б) общая нечеткость расчетного значения выходной переменной $\tilde{Y}$ должна быть минимизирована. При этом во многих работах используется дефаззифицированное значение выходной переменной или ее оценки по fuzzy-регрессионной модели, либо критерии вида $\max\{0;\ \mu_{\bar{Y}_i}[F(\bar{Y}_i)] - \mu_{\bar{Y}_i}(\bar{Y}_i)\}$ — ограниченная разность нечетких чисел, где $\mu_{\bar{Y}_i}(\bar{Y}_i)$ и $\mu_{\bar{Y}_i}[F(\bar{Y}_i)]$ — соответственно функции принадлежности нечеткого множества выходной переменной и ее расчетного значения на основе fuzzy-регрессионной модели. Здесь и в дальнейшем, в отличие от действительных чисел, символами с чертой сверху обозначаются нечеткие множества. Верхний индекс переменных и нечетких множеств определяет индекс соответствующей переменной или коэффициента модели, а нижний — номер комплекта информации. Большинство публикаций по данной тематике либо рассматривали некоторые частные случаи одной из этих общих постановок задачи, либо давали интересные новые приложения ее применения, либо описывали алгоритмы решения известных постановок этой задачи. П. Даймонд [9, 10] ввел новое понятие расстояния на множестве нечетких чисел между прогнозируемыми и экспериментальными данными. В работах Х. Танака (1982 г.) [11, 12], в работе автора [4], в статьях [9, 10, 13] и во многих других публикациях рассмотрена модель линейной регрессии с нечеткими коэффициентами в виде треугольных fuzzy-чисел. В работах [4, 11, 12] для определения значений коэффициентов модели, минимизирующих суммарную средневзвешенную размытость параметров функции принадлежности, рассматриваемую в различных метриках, предложены методы линейного программирования. В 1987 г. А. Селминс [13] и П. Даймонд [9, 10], а также P. Rousseeuw [14], Янг и Лиу в 2003 г. [15] также предложили методику построения моделей нечеткой регрессии методом наименьших квадратов. Эти подходы, комбинированные с методом наименьших квадратов и получившие название FLSRA (fuzzy least-square regression analysis), были предложены Diamond в 1988 г. и Celmiņš в 1987 г. Предложенные методы, в свою очередь, имеют две разновидности, в одной из которых используется критерий максимальной совместимости, а в другой — критерий минимизации квадратичного отклонения.

Для построения критериев аппроксимации использовались различные метрики, среди которых наибольшее распространение получили показатели λ-сечений нечетких множеств (см., например, [14]). В ряде случаев сформулированная оптимизационная задача становится нелинейной и многоэкстремальной. Для решения ее применялись градиентные, поисковые методы и генетические алгоритмы (см., например, [16]).

2. Постановки задачи

Необходимо на основе представительной выборки, заданной матрицей fuzzy-чисел: $|\bar{X}\bar{Y}|$, где $\bar{X}_i^j$, $j = 1, ..., n$, $i = 1, ..., N$, и все выходные переменные $\bar{Y}_i$, $i = 1, ..., N$, — в самом общем случае нечеткие множества, найти нечеткую линейную модель в виде

$$\bar{Y} = \bar{A}_1 \otimes \bar{X}_1 + \bar{A}_2 \otimes \bar{X}_2 + \ldots + \\ + \bar{A}_i \otimes \bar{X}_i + \ldots + \bar{A}_n \otimes \bar{X}_n + \bar{A}_0. \quad (1)$$

Здесь $\bar{A}_1, \bar{A}_2, \ldots, \bar{A}_j, \ldots, \bar{A}_n$ и $\bar{A}_0$ — некоторые fuzzy-множества с заданными (с точностью до неизвестных параметров) функциями принадлежности. Операции "⊗" и "+" — соответственно операции умножения и сложения fuzzy-множеств. Результат вычисления по формуле (1) — также некоторое fuzzy-множество.

Представляет определенный интерес также некоторые частного вида постановки задачи fuzzy-регрессионного анализа:

1) матрица наблюдений $(\bar{X}\bar{Y})$ представлена fuzzy-числами $\bar{X}_i^1, \bar{X}_i^2, \ldots, \bar{X}_i^j, \ldots, \bar{X}_i^n$ и $\bar{Y}_i$, $i = 1, ..., N$, в каждом i-м комплекте информации. Необходимо найти нечеткую линейную регрессионную модель вида

$$\bar{Y} = b^1 \times \bar{X}^1 + b^2 \times \bar{X}^2 + \ldots + b^j \times \bar{X}^j + \\ + \ldots + b^n \times \bar{X}^n + \bar{B}, \quad (2)$$

где $\bar{X}_i^1, \bar{X}_i^2, \ldots, \bar{X}_i^j, \ldots, \bar{X}_i^n$ и $\bar{Y}_i$, $j = 1, ..., n$, $i = 1, ..., N$, — некоторые нечеткие множества с заданными функциями принадлежности; коэффициенты линейной модели $b^1, b^2, ..., b^j, ..., b^n$ — некоторые действительные положительные числа, а свободный член уравнения $\bar{B}$ — нечеткое множество;

2) входные переменные $x_i^1, x_i^2, \ldots, x_i^j, \ldots, x_i^n$ — действительные числа, а выходные переменные $\bar{Y}_i$, $i = 1, ..., N$, коэффициенты линейной регрессионной модели и свободный член $\bar{B}^j$, $j = 0, ..., n$ — нечеткие множества.

Отметим, что в частном случае некоторые входные переменные $\bar{X}_{ij}$ и выходная переменная в комплектах информации могут быть представлены действительными числами.

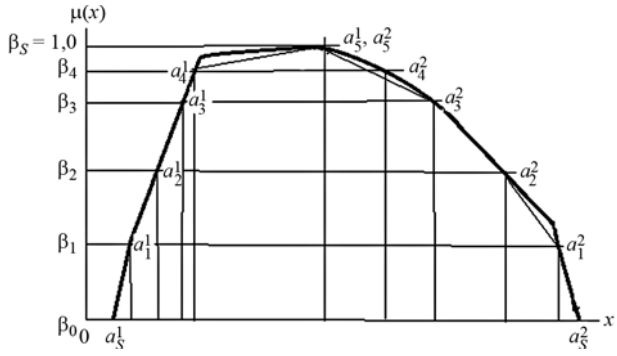
Ниже будет представлено решение задач нечеткого регрессионного анализа в условиях, когда входные и выходная переменная представлены fuzzy-множествами самого общего вида, а коэффициенты регрессии — отрицательные или положительные действительные числа. Свободный член уравнения регрессии — нечеткое множество самого общего вида. Рассмотрены некоторые новые критерии аппроксимации, основанные на сравнении средневзвешенных координат минимальных и максимальных значений абсцисс λ-сечений функций принадлежности нечетких множеств выходной переменной и ее оценки по fuzzy-регрессионной модели, а также координат центров тяжести функций принадлежности fuzzy-множеств. Число сечений, значения функции принадлежности каждого из выбираемых сечений и весовые коэффициенты соответствующих отклонений, которые учитываются в критерии аппроксимации, определяются экспертами и лицом, принимающим решение. Результат расчета fuzzy-регрессионной модели — нечеткое множество, функция принадлежности которого аппроксимируется многоугольником, соединяющим координаты расчетных точек.

В публикациях автора [18, 19] сформулированная задача была решена методом наименьших квадратов в условиях ограничений, что значения коэффициентов регрессии — положительные числа. В данной работе эти ограничения отсутствуют. Кроме того, существенной особенностью рассматриваемой в данной работе постановки задачи является необходимость учета и сохранения в модели качественного влияния отдельных входных факторов (положительного или отрицательного) на выходную переменную, т. е. наличия определенных ограничений. Это не позволяет решить данную задачу методом наименьших квадратов. В этих условиях в данной статье получены детерминированные эквиваленты сформулированных задач и алгоритмы расчета параметров критериев аппроксимации, детерминированных значений коэффициентов уравнения регрессии, а также свободного члена, представленного fuzzy-множеством, определенным с точностью до неизвестных параметров, основанные на решении задач линейного программирования.

Полученные результаты позволяют разработать эффективные алгоритмы построения моделей для многих частных видов функций принадлежности исходных данных задачи (в частности, прямоугольного, треугольного и трапецеидального видов), а также решать многие прикладные проблемы в экономике, логистике, социологии и маркетинге.

3. Критерии аппроксимации, математическая модель и алгоритм решения задачи

Рассматриваются нормализованные нечеткие множества (см. рисунок) с LR-представлением функции принадлежности, т. е. функции, значения функции принадлежности которых $\mu_{\bar{A}}(\bar{A})$ начиная с некоторого значения абс-



Нечеткое множество произвольного вида

цисс $x = a(\bar{A})$, $\mu_{\bar{A}}[a(\bar{A})] = 0$ растут до значения $x = m_1(\bar{A})$, $\mu_{\bar{A}}[m_1(\bar{A})] = 1,0$, на некотором отрезке $x \in [m_1(\bar{A}), m_2(\bar{A})]$ (в частном случае, только в одной точке) $x = [m_1(\bar{A}) = m_2(\bar{A}) = m(\bar{A})]$ имеют постоянное значение $\mu_{\bar{A}}[m_1(\bar{A})] = \mu_{\bar{A}}[m_2(\bar{A})] = 1,0$, а на отрезке $x \in [m_2(\bar{A}), b(\bar{A})]$ убывают до значения $\mu_{\bar{A}}[b(\bar{A})] = 0$. Среди достаточно большого числа функций этого класса наибольший интерес представляют fuzzy-множества с функцией принадлежности прямоугольного, треугольного и трапециевидного типа.

Обозначим координаты абсцисс крайних точек каждого из этих сечений соответственно $a_k^1[A(\beta_k)]$ и $a_k^2[A(\beta_k)]$, $k = 0, 1, ..., K$, координаты абсцисс крайних точек функций принадлежности соответствующих fuzzy-множеств при $\beta_0 = \beta_K = 0$ — $R^1(\bar{A})$ и $R^2(\bar{A})$, а крайних точек функций принадлежности соответствующих fuzzy-множеств при $\beta_s = 1,0$, где $s = \dfrac{K+1}{2}$, — соответственно $T^1(\bar{A})$ и $T^2(\bar{A})$. Здесь, если $T^1(\bar{A}) = T^2(\bar{A})$, т. е. $\mu_{\bar{A}}(A)$ имеет только одну координату абсцисс, для которой $\mu_{\bar{A}}(A) = 1,0$, то K — четное число.

Координаты абсцисс соответствующих сечений функций принадлежности определяются по формулам:

$$\begin{aligned} a_k^1[\bar{A}(\beta_k)] &= R^1(\bar{A}) + \beta_k[T^1(\bar{A}) - R^1(\bar{A})]; \\ a_k^2[\bar{A}(\beta_k)] &= R^2(\bar{A}) - \beta_k[R^2(\bar{A}) - T^2(\bar{A})], \\ k &= 0,\ 1, \ldots, K. \end{aligned} \qquad (3)$$

В результате умножения нечеткого множества на некоторое действительное число b координаты абсцисс соответствующих сечений функций принадлежности fuzzy-множества $\bar{D}$ вычисляются по формулам:

$$a_k^1[\bar{D}(\beta_k)] = ba_k^1[\bar{A}(\beta_k)] = \begin{cases} b\{R^1(\bar{A}) + \beta_k[T^1(\bar{A}) - R^1(\bar{A})]\}, \text{ если } b \geqslant 0; \\ b\{R^2(\bar{A}) - \beta_k[R^2(\bar{A}) - T^2(\bar{A})]\}, \text{ если } b < 0; \end{cases} \qquad (4)$$

$$a_k^2[\bar{D}(\beta_k)] = ba_k^2[\bar{A}(\beta_k)] = \begin{cases} b\{R^2(\bar{A}) - \beta_k[R^2(\bar{A}) - T^2(\bar{A})]\}, \text{ если } b \geqslant 0; \\ b\{R^1(\bar{A}) + \beta_k[T^1(\bar{A}) - R^1(\bar{A})]\}, \text{ если } b < 0. \end{cases} \qquad (5)$$

В результате сложения fuzzy-множеств координаты соответствующих сечений вычисляются по формулам:

$$a_k^1[\bar{P}(\beta_k)] = \sum_{i=1}^{n} a_k^1[\bar{A}_i(\beta_k)] = \left\{\sum_{i=1}^{n} R^1(\bar{A}_i) + \sum_{i=1}^{n} \beta_k[T^1(\bar{A}_i) - R^1(\bar{A}_i)]\right\}; \qquad (6)$$

$$a_k^2[\bar{P}(\beta_k)] = \sum_{i=1}^{n} a_k^2[\bar{A}_i(\beta_k)] = \left\{\sum_{i=1}^{n} R^2(\bar{A}_i) - \sum_{i=1}^{n} \beta_k[R^2(\bar{A}_i) - T^2(\bar{A}_i)]\right\}. \qquad (7)$$

В задаче fuzzy-регрессионного анализа в каждом из N комплектов информации $i = 1, ..., N$, нечеткие множества входной $\bar{X}_{ik}$ и выходной переменной $\bar{Y}_i$ представляются $2(K + 1)$ детерминированными параметрами, соответственно:

$$a_0^1(\bar{X}_i^j), ..., a_k^1(\bar{X}_i^j), ..., a_K^1(\bar{X}_i^j), a_0^2(\bar{X}_i^j), ..., a_k^2(\bar{X}_i^j), ..., a_K^2(\bar{X}_i^j)$$

и $a_0^1(\bar{Y}_i), ..., a_k^1(\bar{Y}_i), ..., a_K^1(\bar{Y}_i), a_0^2(\bar{Y}_i), ..., a_k^2(\bar{Y}_i), ..., a_K^2(\bar{Y}_i)$.

В качестве неизвестных детерминированных параметров (коэффициентов модели) определены детерминированные значения b^1, b^2, ..., b^j, ..., b^n, и уравнение регрессии будем искать в виде

$$a_k^r(\bar{Y}) = \sum_{j=1}^{n} b^j a_k^r(\bar{X}_k^{jr}) + B_k^r, \quad r = 1, 2, \ k = 0, 1, \ \ldots, K. \qquad (8)$$

Здесь $\bar{B}$ — нечеткое множество с функцией принадлежности, определенной с точностью до неизвестных значений детерминированных параметров — $(B_0^r, B_1^r, ..., B_k^r, ..., B_K^r)$, $r = 1, 2$, т. е. каждого из двух детерминированных значений во всех сечениях.

В качестве критериев оптимальности аппроксимации может быть принято минимальное абсолютное значение суммарной величины отклонений расчетных и фактических значений выходной переменной в каждом из сечений соответствующих нечетких множеств, т. е. величины $\min \left| a_k^r(\bar{Y}_i) - \sum_{j=1}^{n} b^j a_k^r(\bar{X}_k^j) - B_k^r \right|$, $r = 1, 2$, $k = 0, 1, ..., K$. Здесь $r = 1, 2$ — индексы крайних точек соответствующих сечений.

Задача выбора коэффициентов уравнения регрессии может быть представлена в виде задачи многокритериальной оптимизации вида

$$a^r_{k,1}(F) = \min_{\substack{b^1,\ldots,b^j,\ldots,b^n,\\ B^1_0,B^2_0,\ldots,B^1_K,B^2_K}} \sum_{i=1}^{N}\left[a^r_k(\bar{Y}_i) - \sum_{j=1}^{n} b^j a^r_k(\bar{X}^j_i) - B^r_k\right], \quad (9)$$

$$r = 1, 2,\ k = 0, 1, \ldots, K;$$

$$a^r_{k,2}(F) = \min_{\substack{b^1,\ldots,b^j,\ldots,b^n,\\ B^1_0,B^2_0,\ldots,B^1_K,B^2_K}} \sum_{i=1}^{N}\left[\sum_{j=1}^{n} b^j a^r_k(\bar{X}^j_i) + B^r_k - a^r_k(\bar{Y}_i)\right], \quad (10)$$

$$r = 1, 2,\ k = 0, 1, \ldots, K.$$

Аддитивная свертка критериев [17, 18] может быть представлена в виде

$$\bar{\Phi} = \min_{\substack{b^1,b^2,\ldots,b^k,\ldots,b^K,\\ B^1_0,B^2_0,\ldots,B^1_K,B^2_K}} \sum_{r=1}^{2}\left\{\delta^r_k \sum_{k=0}^{K}\sum_{i=1}^{N}\left[a^r_k(\bar{Y}_i) - \sum_{j=1}^{n} b^j a^r_k(\bar{X}^j_i) - B^r_k\right] + \mu^r_k \sum_{k=0}^{K}\sum_{i=1}^{n}\left[\sum_{j=1}^{n} b^j a^k_r(\bar{X}^j_i) + B^r_k - a^k_r(\bar{Y}_i)\right]\right\}. \quad (11)$$

Выражение (11) позволяет свести сформулированную многокритериальную задачу к минимизации одного критерия, линейного относительно вектора переменных задачи $(b^0, b^1, \ldots, b^j, \ldots, b^n, B^1_0, B^2_0, \ldots, B^1_K, B^2_K)$ в условиях выполнения ограничений на знаки соответствующих значений коэффициентов.

В выражении (11) приняты следующие обозначения: $0 \leqslant \delta^1_k \leqslant 1$, $0 \leqslant \delta^2_k \leqslant 1$; $0 \leqslant \mu^1_k \leqslant 1$, $0 \leqslant \mu^2_k \leqslant 1$ — весовые коэффициенты, которые удовлетворяют соотношению

$$\sum_{k=0}^{K}(\delta^1_k + \delta^2_k + \mu^1_k + \mu^2_k) = 1; \quad (12)$$

$a^1_k(\bar{Y}_i)$, $a^2_k(\bar{Y}_i)$; $a^1_k(\bar{X}^1_k)$, $a^2_k(\bar{X}^2_k)$, $k = 0, 1, ..., K$ — соответственно минимальные и максимальные значения координаты абсцисс fuzzy-множеств выходной переменной и j-й входной переменной в k-м сечении i-го комплекта информации (исходных данных);

B^1_k, B^2_k — соответственно значения левой и правой координат абсциссы fuzzy-множества свободного члена уравнения регрессии в k-м сечении.

Из свойств LR-представления функции принадлежности нечеткого множества уравнения регрессии должны быть предусмотрены следующие ограничения на коэффициенты этого уравнения:

$$\sum_{j=1}^{n} b^j a^1_{k-1}(\bar{X}^j) + B^1_{k-1} \geqslant \sum_{j=1}^{n} b^j a^1_k(\bar{X}^j) + B^1_k; \quad (13)$$

$$\sum_{j=1}^{n} b^j a^2_{k-1}(\bar{X}^j) + B^2_{k-1} \leqslant \sum_{j=1}^{n} b^j a^2_k(\bar{X}^j) + B^2_k. \quad (14)$$

Обозначим коэффициенты уравнения регрессии:

$$Q^j = \sum_{i=1}^{N}\sum_{k=0}^{K}[\delta^2_k a^2_k(\bar{X}^j_i) - \mu^1_k a^1_k(\bar{X}^j_i)];$$
$$D^j = \sum_{i=1}^{N}\sum_{k=0}^{K}[\mu^1_k a^1_k(\bar{Y}_i) - \delta^2_k a^2_k(\bar{Y}_i)],\ j = 1, \ldots, n. \quad (15)$$

Если все входные переменные имеют положительное влияние на выходную переменную задачи, т. е. с увеличением значения $\bar{X}^j$ значение $\bar{Y}$ должно увеличиваться, то задача аппроксимации сводится к решению только одной задачи линейного программирования — минимизации линейного критерия вида

$$\min_{\substack{b^1,\ldots,b^j,\ldots,b^n,\\ B^1_0,B^2_0,\ldots,B^1_K,B^2_K}} \sum_{j=1}^{n} Q^j b^j + \sum_{k=0}^{K}(\mu^1_k B^1_k - \delta^2_k B^2_k); \quad (16)$$

$$b^j \geqslant 0, j = 1, \ldots, n;\ B^r_k \geqslant 0, k = 0, 1, \ldots, K,\ r = 1, 2. \quad (17)$$

В ряде случаев экспертами определены качественные влияния (положительные или отрицательные) входных параметров на выходную переменную, которые должны быть отражены также в уравнении регрессии. Пусть $\tilde{J}_1, \tilde{J}_2, \tilde{J}_3$ — соответственно подмножества входных переменных, для которых должно быть отражено положительное, отрицательное влияние входного фактора на выходную переменную и факторов, для которых характер влияния не установлен:

$$\tilde{J}_1 \cup \tilde{J}_2 \cup \tilde{J}_3 = \tilde{J} = \{j = 1, \ldots, n\},$$
$$J_1 \cap \tilde{J}_2 = \tilde{J}_1 \cap \tilde{J}_3 = \tilde{J}_2 \cap \tilde{J}_3 = \varnothing. \quad (18)$$

Пусть подмножество $\tilde{J}_3$ состоит из m_3 переменных, тогда оптимальное решение задачи может быть получено в результате решения 2^{m_3} задач вида

$$\tilde{\Phi}(\tilde{H}^3) = \min_{\substack{b^1,\ldots,b^n,\\ B^1_0,B^2_0,\ldots,B^1_K,B^2_K}} \sum_{j\in\tilde{J}^1\cup\tilde{H}^3}\left\{Q^j b^j + \sum_{\tilde{J}^2\cup(\tilde{J}^3/\tilde{H}^3)} D^j b^j + \sum_{k=0}^{K}(\mu^1_k B^1_k - \delta^2_k B^2_k)\right\} \quad (19)$$

в условиях ограничений (13), (14), 17).

Здесь $\tilde{H}^3$ — различные подмножества переменных из подмножества $\tilde{J}^3$, число которых равно 2^{m_3}. Среди этих всех полученных реше-

ний выбирается решение с наименьшим значением критерия оптимальности. Рассчитанные значения детерминированных коэффициентов этой математической модели b^j, $j = 1, ..., n$, и крайние точки соответствующих сечений нечеткого множества свободного члена B_k^r, $k = 0, 1, ..., K$, $r = 1, 2$, принимаются в виде нечеткой регрессионной модели, построенной на основе данного комплекта исходных данных.

4. Оценка адекватности Fuzzy-регрессионой модели

Результат расчета выходных показателей на основе нечеткой регрессионной модели — это fuzzy-множество $F(\bar{Y}_i)$, функция принадлежности которого $\mu_{\bar{Y}_i}[F(\bar{Y}_i)]$ представлена также многоугольником заданного вида (см. рисунок). В качестве детерминированного аналога прогнозируемой величины в каждом комплекте информации может использоваться координата абсциссы центра тяжести расчетного нечеткого множества — $G[F(\bar{Y}_i)]$.

В качестве оценки адекватности построенной Fuzzy-регрессионой модели могут быть приняты среднеквадратические значения суммы квадратов расчетного и фактического отклонений:

- координата абсцисс центра тяжести нечеткого множества, полученного в результате расчета по fuzzy-регрессионной модели $\{G[\bar{Y}_i] - G[F(\bar{Y}_i)]\}^2$, где $G[\bar{Y}_i]$ — фактическое значение абсциссы центра тяжести нечеткого множества значения выходной переменной, которое для $i = 1, ..., N$ вычисляется по формулам

$$G(\bar{Y}_i) = \frac{\int_{-\infty}^{\infty} \bar{Y}_i \mu_{\bar{Y}_i}(\bar{Y}_i) d\bar{Y}_i}{\int_{-\infty}^{\infty} \mu_{\bar{Y}_i}(\bar{Y}_i) d(\bar{Y}_i)};$$

- некоторая средневзвешенная величина абсолютных значений разности координат функции принадлежности фактического и расчетного значений функций принадлежности нечеткого множества $[Q[\bar{Y}_i] - Q[F(\bar{Y}_i)]]$. Здесь

$$Q[\bar{Y}_i] = \sum_{r=1}^{2} \sum_{k=0}^{K} \rho_k^r a_k^r [\bar{Y}_i];$$

$$Q[F(\bar{Y})_i] = \sum_{r=1}^{2} \sum_{k=0}^{K} \rho_k^r a_k^r F[(\bar{Y}_i)],$$

где $a_k^r[\bar{Y}_i]$ и $a_k^r[F(\bar{Y}_i)]$ — соответственно левые и правые крайние точки координат оси абсцисс функции принадлежности нечеткого множества, построенного по уравнениям регрессионной модели для i-го комплекта информации; $Q[\bar{Y}_i]$, $Q[F(\bar{Y})_i]$ — соответственно фактическое и расчетное значения функций принадлежности нечеткого множества в i-м комплекте информации; ρ_k^r, $r = 1, 2$, — весовые коэффициенты, значения которых в частном случае могут быть приняты такими же, как при расчете параметров регрессионной модели;

- абсолютное значение суммы отклонений расчетного и фактического значений функций принадлежности одних и тех же контрольных точек выходной переменной $E[\mu_{\bar{Y}_i}(\bar{Y}_i)] - E[\mu_{F(\bar{Y}_i)} F(\bar{Y}_i)]$, где

$$E[\mu_{\bar{Y}_i}(\bar{Y}_i)] = \sum_{r=1}^{2} \sum_{k=0}^{K} \delta_k^r \mu_{a_k^r(\bar{Y}_i)}[a_k^r(\bar{Y}_i)];$$

$$E[\mu_{\bar{Y}_i}\{F(\bar{Y}_i)\}] = \sum_{r=1}^{2} \sum_{k=0}^{K} \delta_k^r \mu_{a_k^r[F(\bar{Y}_i)]}\{[a_k^r[F(\bar{Y}_i)]\}.$$

Здесь $E[\mu_{\bar{Y}_i}(\bar{Y}_i)]$, $E[\mu_{\bar{Y}_i}\{F(\bar{Y}_i)\}]$ — соответственно значения фактической и расчетной сумм значений функций принадлежности одних и тех же контрольных точек выходной переменной.

В качестве оценки качества прогнозирования на основе нечеткой регрессионной модели может рассматриваться следующий показатель:

$$D = \frac{1}{N-1} \sum_{i=1}^{N} \{W(\bar{Y}_i) - W[F(\bar{Y}_i)]\}^2,$$

где в качестве значения $W(\bar{Y}_i)$ могут использоваться определенные выше показатели $G(\bar{Y}_i)$, $Q(\bar{Y}_i)$ или $E(\bar{Y}_i)$, а в качестве значения $W[F(\bar{Y}_i)]$ — показатели $G[F(\bar{Y}_i)]$, $Q[F(\bar{Y}_i)]$ или $E[F(\bar{Y}_i)]$.

С достаточной для практических приложений точностью в большинстве случаев могут использоваться построенные Fuzzy-регрессионные модели для прогнозирования значения выходной переменной, если справедливы следующие показатели их адекватности:

$$D \leqslant \omega \frac{1}{N-1} \{W(\bar{Y}_i^r) - M[W(\bar{Y}^r)]\}^2,$$

где $M[W(\bar{Y}^r)] = \frac{1}{N} \sum_{i=1}^{N} W(\bar{Y}_i^r)$, а значение весового коэффициента выбирается из условий $\omega \leqslant 0{,}1$.

Заключение

В условиях, когда в статистической выборке некоторые или все входные и выходная переменная представлены нечеткими множествами, а также из экономических соображений, технических характеристик или технологических

условий установлены качественные влияния отдельных входных факторов, использование предлагаемых в работе fuzzy-регрессионных моделей является эффективной альтернативой получения количественных зависимостей установленных экспертами качественных закономерностей изучаемых явлений.

В работе предложены методы решения задачи в условиях, когда входные и выходная переменная и свободный член уравнения регрессии представлены fuzzy-множествами с *LP*-представлением функции принадлежности самого общего вида, а коэффициенты регрессии — действительные числа. В качестве критериев аппроксимации использованы сумма квадратов средневзвешенных координат минимальных и максимальных значений абсцисс сечений функций принадлежности нечетких множеств выходной переменной и их оценок по fuzzy-регрессионной модели. Построен детерминированный эквивалент и приведена вычислительная схема алгоритма решения задачи.

Ограничения, связанные с учетом необходимости учета в уравнении регрессионной модели положительного или отрицательного влияния на выходную переменную отдельных входных факторов, а также свойства арифметических операторов fuzzy-множеств, обусловили необходимость решения 2^{m_3} задач линейного программирования с последующим выбором среди них решения с наилучшим значением критерия оптимальности.

Полученные в работе результаты расширят область приложения fuzzy-регрессионных моделей в экономике, технических системах, социологии, маркетинге и других приложениях.

Список литературы

1. **Дрейпер Н., Смит Г.** Прикладной регрессионный анализ. Множественная регрессия. М.: Диалектика, 2007. 912 с.

2. **Орлова И. В., Половников В. А.** Экономико-математические методы и модели. М.: Вузовский учебник: ИНФРА-М, 2010. 366 с.

3. **Орлов А. И.** Нечисловая статистика. М.: МЗ-Пресс, 2004. 513 с.

4. **Зак Ю. А.** Принятие эффективных решений в экономике и менеджменте в условиях наличия нечисловой информации и размытых данных. М.: Экономика, 2018. 239 с.

5. **Зак Ю. А.** Методы обработки нечисловой информации в маркетинговых исследованиях // Маркетинг и маркетинговые исследования, Grebennikov. 2013. № 1. С. 20—33.

6. **Chang Yun-Hsi O.** Fuzzy regression methods — a comparative assessment // Fuzzy Sets und Systems. 2001. Vol. 119 (2). P. 187—203.

7. **Chang Yun-Hsi O.** Hybrid fuzzy least squares regression analysis and its reliability measures // Fuzzy Sets und Systems. 2001. Vol. 119 (2). P. 225—246.

8. **Штовба С. Д.** Нечеткая идентификация на основе регрессионных моделей параметрической функцией принадлежности // Проблемы управления и информатики. 2006. № 6. С. 1—8.

9. **Diamond P.** Fuzzy least squares // Information Sciences. 1988. Vol. 46. P. 141—157.

10. **Diamond P.** Least Squares Fitting of Several Fuzzy Variables // Proceedings of Secon IFSA Congress. Tokio, 1987. P. 20—25.

11. **Tanaka H., Uejima S., Asai K.** Linear regression analysis with fuzzy model // IEEE Trans. Systems Man Cybernet. 1982. Vol. 12, N. 6. P. 903—907.

12. **Tanaka H., Warada J.** Possiblistic linear system and their application tot he linear regression model // Fuzzy Sets und Systems. 1988. Vol. 27. P. 275—289.

13. **Celmins A.** Least Squares Model Fitting to Fuzzy Vektor Data // Fuzzy Sets und Systems. 1987. Vol. 22. P. 260—269.

14. **Rousseeuw P.** Applyng robust regression to insurance // Insurance: Mathematics and Economics. 1984. Vol. 3, N. 1. P. 67—72.

15. **Yang M.-S., Lee H. H.** Fuzzy Least Squares Alqorithmus for interactive Fuzzy Linear Regressions Models // Fuzzy Sets und Systems. 2003. Vol. 135, N. 2. P. 305—316.

16. **Aliev R., Fazlollahi B., Vahidov R.** Genetic algorithms-based fuzzy regression analysis // Soft Computing. 2002. N. 6. P. 470—475.

17. **Куни Р. Л., Райфа Х.** Принятие решений при многих критериях: предпочтения и замещения. М.: Радио и связь, 1981. 560 с.

18. **Зак Ю. А.** Прикладные задачи многокритериальной оптимизации. М.: Экономика, 2014. 455 с.

19. **Зак Ю. А.** Fuzzy-регрессионные модели в условиях наличия в статистической выборке нечисловой информации // Системні дослідження та інфомаційні технології. 2017. № 1. С. 88—96.

Yu. A. Zack, D. Sc., e-mail: yuriy_zack@hotmail.com

Fuzzy Linear Regression Analysis, Taking into Account the Influence of Input Factors

A solution is presented for fuzzy regression analysis problems in conditions where the input and output variables are represented by normalized fuzzy sets with an LR representation of the most general form membership function, and the regression coefficients are negative or positive real numbers. The free term of the regression equation is a fuzzy set of the most general form. There are restrictions on the degree of influence of some input factors established by experts. The approximation criterion is the minimum absolute value of the average unweighted sum of the absolute values of the coordinates of the minimum and maximum abscissa values of the cross sections for the membership functions of fuzzy sets of the output variable and its evaluation by the fuzzy regression model. Regression coefficients are calculated as a result of solving a certain subset of linear programming problems with the subsequent choice among them of the solution with the best value of the optimality criterion.

Keywords: *fuzzy regression models, fuzzy-sets, LR-representation of the membership function, cross sections of membership functions, linear programming*

DOI: 10.17587/it.26.137-144

References

1. **Drajper N., Smit G.** Applied Regression Analysis, Moscow, Dialektika, 2007, 912 p. (in Russian).
2. **Orlova I. V., Polovnikov V. A.** Economic and mathematical methods and models, Moscow, Busovskij uchebnik, INFRA-M, 2010, 366 p. (in Russian).
3. **Orlov A. I.** Non-numeric statistics, Moscow, M3-Press, 2004, 513 p. (in Russian).
4. **Zack Yu. A.** Making effective decisions in economics and management in the presence of non-numerical information and blurry data, Moscow, Ekonomika, 2018, 239 p. (in Russian).
5. **Zack Yu. A.** Non-numerical information processing methods in marketing research, *Marketing i marketingovije issledovanija, Grebennikov*, 2013, no. 1, pp. 20—33 (in Russian).
6. **Chang Yun-Hsi O.** Fuzzy regression methods — a comparative assessment, *Fuzzy Sets und Systems*, 2001, vol. 119 (2), pp. 187—203.
7. **Chang Yun-Hsi O.** Hybrid fuzzy least squares regression analysis and its reliability measures, *Fuzzy Sets und Systems*, 2001, vol. 119 (2), pp. 225—246.
8. **Shtovba S. D.** Fuzzy identification based on regression models by a parametric membership function, *Problemi Upravlenija i Informatiki*, 2006, no. 6, pp. 1—8 (in Russian).
9. **Diamond P.** Fuzzy least squares, *Information Sciences*, 1988, vol. 46, pp. 141—157.
10. **Diamond P.** Least Squares Fitting of Several Fuzzy Variables, Proceedings of Secon IFSA Congress, Tokio, 1987, pp. 20—25.
11. **Tanaka H., Uejima S., Asai K.** Linear regression analysis with fuzzy model, *IEEE Trans. Systems Man Cybernet*, 1982, vol. 12, no. 6, pp. 903—907.
12. **Tanaka H., Warada J.** Possiblistic linear system and their application tot he linear regression model, *Fuzzy Sets und Systems*, 1988, vol. 27, pp. 275—289.
13. **Celmins A.** Least Squares Model Fitting to Fuzzy Vektor Data, *Fuzzy Sets und Systems*, 1987, vol. 22, pp. 260—269.
14. **Rousseeuw P.** Applyng robust regression to insurance, *Insurance: Mathematics and Economics,* 1984, vol. 3, no. 1, pp. 67—72.
15. **Yang M.-S., Lee H. H.** Fuzzy Least Squares Alqorithmus for interactive Fuzzy Linear Regressions Models, *Fuzzy Sets und Systems*, 2003, vol. 135, no. 2, pp. 305—316.
16. **Aliev R., Fazlollahi B., Vahidov R.** Genetic algorithms-based fuzzy regression analysis, *Soft Computing*, 2002, no. 6, pp. 470—475.
17. **Kuni R. L., Rajfa Ch.** Decision making under many criteria: preferences and substitutions, Moscow, Radio i Svyaz, 1981, 560 p. (in Russian).
18. **Zack Yu. A.** Applied problems of multicriteria optimization, Moscow, Ekonomika, 2014, 455 p. (in Russian).
19. **Zack Yu. A.** Fuzzy-regression models in the presence of non-numerical information in a statistical sample, *Systemni doslidzhennya ta infomatsiyni tekhnolohii*, 2017, no. 1, pp. 88—96 (in Russian).

УДК 51-77

DOI: 10.17587/it.26.144-158

В. И. Левин, д-р. техн. наук, проф.,
Пензенский государственный технологический университет

Математические модели и методы обнаружения коррупции в организационных системах

Сформулирована проблема математического моделирования коррупции. Построена модель коррумпированной системы. Предложены математические методы измерения, обнаружения и локализации коррупции в системе. Приведены реальные примеры решения указанных задач.

Ключевые слова: *коррупция, математическое моделирование, измерение, обнаружение, локализация, организационное управление, экспертиза*

Введение

Одной из наиболее старых и не решенных до сих пор проблем большинства развитых стран является проблема коррупции. Для России она не нова. Достаточно вспомнить многочисленные русские пословицы на указанную тему, хотя бы такую: "Не подмажешь — не поедешь!". Но именно в наше время эта проблема приобрела особенно большой размах и остроту. По мнению многих специалистов, она является одной из главных проблем, которые должны быть решены государством. Однако, на наш взгляд, это не только главенствующая, но и первоочередная проблема современной России, с решения которой нужно начинать. Без этого любые реформы и проекты правительства обречены на неудачу, поскольку требующиеся на них вложения новых сил и средств на деле приводят лишь к дальнейшему расширению "коррупционного поля". Положение очень серьезно, так как нарастающая волна коррупции в стране может привести в конце концов к большой нестабильности, а затем к разрушению российского государства.

Имеется много различных определений коррупции. Согласно работе [1] коррупция — это подкуп взятками, продажность должностных лиц и политических деятелей в буржуазных странах, а согласно работе [2] — это подкуп, продажность общественных и политических деятелей, должностных лиц в капиталистическом обществе. Эти определения близки между собой, они грешат произвольными ограниче-

ниями области явления (на самом деле, коррупция существует в капиталистическом, социалистическом и любом другом обществе), его действующих лиц (взятки берут не только должностные лица, политические и общественные деятели, но и рядовые граждане). Более удовлетворительное определение дано в работе [3]: коррупция — это просто подкуп, продажность, взяточничество. Наиболее емкое и точное из существующих определений приведено, на наш взгляд, в работе [4]. Согласно ему коррупция — это аморальные, развращенные, нечестные действия любых лиц, выражающиеся, в первую очередь, в предложении и получении взяток. Несколько иначе понимают коррупцию в нормативных документах различных стран и международных организаций [5]. Так, в документах ООН по борьбе с коррупцией последняя трактуется как злоупотребление государственной властью для получения личной выгоды, в документах группы по коррупции Совета Европы — как любое поведение лиц (в том числе взяточничество), которым поручено выполнение определенных обязанностей в государственном или частном секторе, ведущее к нарушению данных обязанностей. В России коррупцией считается преступная деятельность в политике или государственном управлении в форме использования должностными лицами властных полномочий для личного обогащения.

Детальные сведения о современной коррупции (ее виды, размах, национальные особенности, связанные с ней опасности, научный подход к ее количественному изучению в рамках специальной науки корруметрии и др.) приведены в работе [6]. Мы же в данной работе изложим простейший коррумметрический, так называемый детерминистский подход, позволяющий обнаруживать и измерять коррупцию в организационных системах из экспертов.

1. Постановка задачи

Теперь приведем формализованную постановку двух основных задач науки корруметрии. *Задача 1*: разработка математической модели и метода, позволяющих по имеющейся информации о работе организационной системы обнаружить факт наличия коррупции в ней, а точнее, установить, имеется ли коррупция в работе указанной системы или нет. *Задача 2*: разработка математической модели и метода, позволяющих по имеющейся информации о работе организационной системы измерить (вычислить) уровень коррупции в ней, точнее, указать точку на некоторой введенной шкале уровней, которая измеряет степень коррупции в работе указанной системы. *Задачу 1* будем называть *задачей обнаружения (идентификации) коррупции*, а *задачу 2 — задачей измерения (анализа) коррупции.*

Далее в статье мы рассматриваем организационные системы, состоящие из экспертов. Каждый участник организационной системы функционирует на основе количественных и/или качественных оценок, которые он дает объектам своей деятельности. Так, врач оценивает состояние здоровья пациента и на этой основе назначает лечение, преподаватель оценивает знания учащегося и на базе этого корректирует программу его подготовки, член конкурсной комиссии оценивает уровень поданного на конкурс проекта и исходя из этого голосует за или против поддержки проекта и т. д. Все эти люди могут быть названы экспертами, поскольку даваемые ими оценки различных объектов являются экспертными, т. е. зависящими от уровня квалификации, честности, добросовестности, независимости служебного поведения и некоторых других качеств конкретного эксперта. Однако должно быть ясно, что разные эксперты, обладающие в высшей степени всеми указанными качествами, будут давать одинаковые оценки одному и тому же объекту (мы здесь не рассматриваем случаи, когда однозначная оценка принципиально невозможна, например, оценка произведений искусства). Эту идеальную ситуацию мы примем за "точку отсчета". В реальности эксперты могут быть малоквалифицированными, недостаточно честными и добросовестными, зависимыми в своем служебном поведении от иных лиц. При этом разные эксперты дают различные оценки одному и тому же объекту, что обусловлено их неквалифицированностью или (гораздо чаще) сугубо личными корыстными интересами, в которых и проявляются их нечестность, недобросовестность и т. д. Последнее и есть проявление коррупции в работе организационной системы. Например, врач сознательно искажает состояние здоровья пациента, побуждая его покупать дорогие лекарства у фирмы, с которой состоит в сговоре; преподаватель сознательно занижает оценку знаний учащегося, заставляя его заключать договор на дополнительные платные образовательные услуги, которые сам и оказывает; член конкурсной комиссии сознательно занижает оценку "чужих" проектов, поданных на конкурс, и завышает оценку "своих" проектов (разумеется, за соответствующую плату) и т. д. Очевидно, что чем в большей степени эксперты обладают указанными отрицательными

качествами, ведущими к коррупции, тем больше расстояние между результатами экспертизы у различных экспертов, а также расстояние между коллективной экспертной оценкой, даваемой одному и тому же объекту реальными, коррумпированными экспертами и идеальными экспертами.

Из сказанного выше вытекает следующая формализованная постановка задач обнаружения и измерения коррупции. Пусть существует некоторая организационная система с конечным числом экспертов. Система считается реальной в том смысле, что по крайней мере часть ее экспертов являются работниками не самого высокого уровня в отношении их квалификации, честности, добросовестности и независимости. Однако считается невозможным сговор всех экспертов в отношении даваемых оценок. Гипотетическую систему, в которую превратилась бы наша реальная организационная система, если бы в один прекрасный день все ее эксперты стали в высшей степени квалифицированными, честными, добросовестными и независимыми, назовем "идеальной". Тогда *задача измерения коррупции* может быть сформулирована следующим образом: 1) найти объективный количественный показатель уровня коррупции в реальной системе в виде подходящего показателя расстояния между результатами экспертизы у разных экспертов реальной системы; 2) построить математическую модель, позволяющую эффективно находить уровень коррупции в реальной системе. Аналогично, *задача обнаружения коррупции* может быть сформулирована так: 1) найти объективный критерий существования коррупции в реальной системе в виде подходящего критического значения показателя уровня коррупции, превышение которого сигнализирует о существовании коррупции в системе; 2) построить математическую модель, позволяющую вести вычисления, нужные для обнаружения коррупции в системе.

2. Решение задачи измерения коррупции

Пусть m экспертов, образующих организационную систему, проводят совместную экспертизу одного и того же объекта, оценивая n его показателей. Любой j-й показатель может принимать r_j возможных значений, вместе составляющих множество

$$A_j = \{a_{j1}, a_{j2}, \ldots, a_{jr_j}\},\ j = \overline{1,n}. \qquad (1)$$

Каждый i-й эксперт, $i = \overline{1,m}$, оценивает каждый j-й показатель объекта, $j = \overline{1,n}$, выбирая при этом одно из r_j возможных значений этого показателя $a_{j1}, \ldots, a_{jr_j}$, указанных в (1). В результате проведения экспертизы имеем матрицу экспертных оценок

$$B = \begin{Vmatrix} b_{11} & \ldots & b_{1n} \\ \ldots & \ldots & \ldots \\ b_{m1} & \ldots & b_{mn} \end{Vmatrix}, \qquad (2)$$

в которой b_{ij}, $i = \overline{1,m}$, $j = \overline{1,n}$, — экспертная оценка, данная i-м экспертом j-му показателю объекта. В матрице B, согласно сказанному, элементы j-го столбца выбираются экспертами из множества A_j, определяемого выражением (1), $j = \overline{1,n}$. Предположим, что все эксперты являются в наивысшей степени квалифицированными, честными, добросовестными и независимыми. В этом идеальном случае, как уже говорилось, экспертные оценки, даваемые различными экспертами одному и тому же j-му показателю объекта, равны. Поэтому и наборы оценок показателей объекта, принадлежащие различным экспертам, совпадают. В терминах матрицы экспертных оценок (2) это означает, что в идеальной системе каждый столбец данной матрицы состоит из равных элементов, а все строки совпадают. Реальная система в силу реальных свойств ее экспертов (см. выше) имеет матрицу экспертных оценок B с существенно другими отношениями элементов b_{ij}, чем идеальная система, а именно с различными значениями элементов в столбцах и с несовпадающими строками. Это подсказывает путь нахождения объективного показателя уровня коррупции в реальной системе в виде подходящего показателя расстояния между результатами экспертизы у различных экспертов реальной системы. Во-первых, расстояние между полными результатами экспертизы у различных экспертов складывается из расстояний между частными результатами их экспертизы в отношении каждого из n показателей оцениваемого объекта. Во-вторых, расстояние между частными результатами экспертизы в отношении любого j-го показателя объекта складывается из расстояний между этими частными результатами для каждой пары различных экспертов. В-третьих, расстояние между частными результатами оценки определенного j-го параметра двумя различными экспертами можно оценивать абсолютной величиной разности соответствующих двух оценок. Таким образом, получаем выражение показателя абсолютного уровня коррупции:

$$K = \sum_{j=1}^{n} \sum_{i<q} \left| b_{ij} - b_{qj} \right|. \qquad (3)$$

Здесь $|M|$ — абсолютная величина числа M. Как следует из формулы (3), для идеальных систем в силу равенства всех элементов любого столбца матрицы B таких систем $b_{ij} = b_{qj}$ для всех i, q, откуда $K = 0$, т. е. абсолютный уровень коррупции таких систем равен нулю. Однако для реальных систем указанное условие для матрицы не выполняется, поэтому найдется такая тройка чисел i, q, j, для которой $b_{ij} \neq b_{qj}$, откуда следует $K > 0$, т. е. абсолютный уровень коррупции таких систем положительный.

Для практических целей обычно удобнее пользоваться показателем относительного уровня коррупции, определяемым в виде

$$k = K/K_{\max}, \tag{4}$$

где $K_{\max}$ — максимальное значение абсолютного показателя K. С учетом значений K у идеальных и реальных систем, из соотношения (4) получаем

$$0 < k \leqslant 1, \tag{5}$$

причем нижняя граница интервала соответствует полному отсутствию коррупции (идеальная система), а верхняя — присутствию коррупции в максимально возможном объеме (неадекватная система).

Найдем выражение для $K_{\max}$. Из выражения (3) для K очевидно, что K достигает своего максимального значения $K_{\max}$ тогда, когда наибольшее возможное число слагаемых в выражении (3) принимает максимально возможное значение благодаря тому, что элемент b_{ij} выбран равным верхней, а b_{qj} — нижней границе (или наоборот) диапазона допустимых значений элементов j-го столбца матрицы B, задаваемого множеством (1). Эти границы, таким образом, равны

$$\begin{aligned} a_{j\max} &= \max_j \{a_{j1}, \ldots, a_{jr_j}\}, \\ a_{j\min} &= \min_j \{a_{j1}, \ldots, a_{jr_j}\}. \end{aligned} \tag{6}$$

Комбинация элементов b_{ij} и b_{qj} в формуле (3), нужная нам, достигается в том случае, когда элементы любого j-го столбца матрицы $B = \|b_{ij}\|$ принимают только два возможных значения: $a_{j\max}$ и $a_{j\min}$, причем число элементов с этими значениями одинаково. При этом наиболее обозримое выражение для $K_{\max}$ можно отыскать, когда указанные значения следуют друг за другом в одном из двух возможных порядков:

$$\begin{aligned} &b_{1j} = a_{j\max},\ b_{2j} = a_{j\min},\ b_{3j} = a_{j\max},\ \ldots \\ \text{или }\ &b_{1j} = a_{j\min},\ b_{2j} = a_{j\max},\ b_{3j} = a_{j\min},\ \ldots\ . \end{aligned} \tag{7}$$

Иными словами, при движении вдоль любого j-го столбца матрицы B ее элементы b_{ij} должны чередовать свои значения, принимая попеременно максимальное и минимальное значения для этого столбца. При этом условии из (3) получим следующее развернутое выражение для $K_{\max}$:

$$\begin{aligned} K_{\max} = &\underbrace{|b_{11} - b_{21}| + |b_{11} - b_{41}| + |b_{11} - b_{61}| + \ldots + |b_{21} - b_{31}| + |b_{21} - b_{51}| + |b_{21} - b_{71}| + \ldots + |b_{m-1,1} - b_{m1}|}_{\text{1-й столбец матрицы } B} + \\ &+ \underbrace{|b_{12} - b_{22}| + |b_{21} - b_{42}| + |b_{12} - b_{62}| + \ldots + |b_{22} - b_{32}| + |b_{22} - b_{52}| + |b_{22} - b_{72}| + \ldots + |b_{m-1,2} - b_{m2}|}_{\text{2-й столбец матрицы } B} + \\ &- \\ &+ \underbrace{|b_{1n} - b_{2n}| + |b_{1n} - b_{4n}| + |b_{1n} - b_{6n}| + \ldots + |b_{2n} - b_{3n}| + |b_{2n} - b_{5n}| + |b_{2n} - b_{7n}| + \ldots + |b_{m-1,n} - b_{mn}|}_{n\text{-й столбец матрицы } B}. \end{aligned} \tag{8}$$

В выражении (8) оставлены лишь те слагаемые общего выражения (3), которые в рассматриваемом случае чередования значений элементов столбцов матрицы B согласно формуле (7) не равны нулю. Легко видеть, что оставленные в выражении (8) слагаемые для одного j-го столбца равны одной и той же величине $a_{j\max} - a_{j\min}$. Заметим, что число слагаемых в выражении (8) для любого j-го столбца одно и то же: оно зависит лишь от числа строк m матрицы B и является некоторой функцией $N(m)$. Учитывая сказанное, формулу (8) можно записать в следующем окончательном виде:

$$K_{\max} = N(m) \sum_{j=1}^{n} (a_{j\max} - a_{j\min}). \tag{9}$$

Функция $N(m)$ в соотношении (9) для конкретных m рассчитывается достаточно легко, а ее значения для $m = 2, \ldots, 7$ приведены в табл. 1.

Таблица 1

m	2	3	4	5	6	7
$N(m)$	1	2	4	6	9	12

$N(m)$ можно также выразить аналитически. Действительно, обозначив через $]x[$ целую часть x, получаем из соотношения (8):

$$N(m) =]m/2[+](m-1)/2[+](m-2)/2[+\ldots+0 =$$
$$= \begin{cases} (m/2)+((m/2)-1)+((m/2)-1)+ \\ +((m/2)-2)+((m/2)-2)+\ldots+0, \\ m-\text{четное}; \\ (m-1)/2+(m-1)/2+(m-3)/2+ \\ +(m-3)/2+\ldots+0, \\ m-\text{нечетное}, \end{cases} \quad (10)$$

или, после суммирования,

$$N(m) = \begin{cases} m^2/4 \text{ при } m-\text{четном}; \\ (m^2-1)/4 \text{ при } m-\text{нечетном}. \end{cases} \quad (11)$$

Двойную формулу (11) сведем в одинарную менее явную формулу

$$N(m) = \{m(m-1)/2+]m/2[\}/2. \quad (12)$$

Подставляя значения K из соотношения (3) и $K_{\max}$ из выражения (9) в соотношение (4), получим явное выражение показателя относительного уровня коррупции k:

$$k = \sum_{j=1}^{n} \sum_{i<q} \left|b_{ij} - b_{qj}\right| \Big/ N(m) \sum_{j=1}^{n} (a_{j\max} - a_{j\min}). \quad (13)$$

На практике, в основном, встречаются организационные системы, состоящие из ограниченного (до 5...7) числа экспертов. Явные выражения показателя k относительного уровня коррупции для нескольких таких систем, вытекающие из общего выражения (12), приведены ниже:

$$k = \sum_{j=1}^{n} \left|b_{1j} - b_{2j}\right| \Big/ \sum_{j=1}^{n} (a_{j\max} - a_{j\min}),\ m = 2;$$
$$k = \sum_{j=1}^{n} \left(\left|b_{1j} - b_{2j}\right| + \left|b_{1j} - b_{3j}\right| + \left|b_{2j} - b_{3j}\right|\right) \Big/ 2\sum_{j=1}^{n} (a_{j\max} - a_{j\min}),\ m = 3;$$
$$k = \sum_{j=1}^{n} \left(\left|b_{1j} - b_{2j}\right| + \left|b_{1j} - b_{3j}\right| + \left|b_{1j} - b_{4j}\right| + \left|b_{2j} - b_{3j}\right| + \right.$$
$$\left. + \left|b_{2j} - b_{4j}\right| + \left|b_{3j} - b_{4j}\right|\right) \Big/ 4\sum_{j=1}^{n} (a_{j\max} - a_{j\min}),\ m = 4; \quad (14)$$
$$k = \sum_{j=1}^{n} \left(\left|b_{1j} - b_{2j}\right| + \left|b_{1j} - b_{3j}\right| + \left|b_{1j} - b_{4j}\right| + \left|b_{1j} - b_{5j}\right| + \left|b_{2j} - b_{3j}\right| + \left|b_{2j} - b_{4j}\right| + \right.$$
$$\left. + \left|b_{2j} - b_{5j}\right| + \left|b_{3j} - b_{4j}\right| + \left|b_{3j} - b_{5j}\right| + \left|b_{4j} - b_{5j}\right|\right) \Big/ 6\sum_{j=1}^{n} (a_{j\max} - a_{j\min}),\ m = 5.$$

Изложенный подход к измерению коррупции организационной системы пригоден только для систем с $m \geqslant 2$ экспертами.

3. Решение задачи обнаружения коррупции

Снова рассмотрим организационную систему с m экспертами, изученную выше в п. 1. Как было показано, относительный уровень k коррупции в деятельности указанной системы можно достаточно объективно измерить (оценить) с помощью формулы (13) (для конкретных систем с конкретным числом m — с помощью производных от (13) формул типа (14)). При этом показателю относительного уровня коррупции $k = 0$ соответствует полностью бескоррупционная (идеальная) система, а $k = 1$ — полностью коррумпированная система. Все возможные значения показателя k находятся в интервале от 0 до 1 (формула (5)), причем возрастание k в этом интервале означает монотонное увеличение уровня коррупции в системе от минимально возможного до максимально возможного, убывание k — монотонное уменьшение от максимально возможного до минимально возможного. Такое взаимно однозначное соответствие между предполагаемым уровнем коррупции в системе и математически сконструированным показателем этого уровня k позволяет решить задачу обнаружения коррупции в системе полностью формализованно. Для решения нужно:

1. Выбрать некоторое достаточно малое значение относительного уровня коррупции k, превышение которого можно обоснованно трактовать как объективное свидетельство наличия в системе коррупции. Это значение (обозначим его k_0) естественно называть порогом коррумпированности системы. Необходимость введения порога коррумпированности системы k_0 связана с тем, что слишком малые значения показателя $k (k < k_0)$ могут быть вызваны не свойствами, связанными с коррумпированностью экспертов (нечестность, недобросовестность, зависимость и т. д.), а совсем иными свойствами (в первую очередь, недостаточной квалификацией), играющими при обнаружении коррупции роль "шума", подмешанного к "полезному сигналу". Значение порога коррумпирован-

ности k_0, таким образом, есть возможная погрешность вычисления по формуле (13) показателя коррупции k из-за влияния на построенную математическую модель указанных иных (некоррупционных) свойств экспертов. Поэтому говорить уверенно о наличии в системе коррупции при $k < k_0$ нельзя — это возможно лишь при $k > k_0$.

2. Вычислить значение показателя относительного уровня коррупции в системе k, опираясь на информацию о работе данной системы, содержащуюся в матрице экспертных оценок B вида (2) и множествах возможных значений показателей подвергаемого экспертизе объекта, задаваемых в форме (1). Для вычисления используем общую формулу (13) или ее конкретизированные варианты (14), относящиеся к системам с конкретизированными числами экспертов m.

3. Сравнить вычисленное значение показателя относительного уровня коррупции в системе k с выбранным значением порога коррумпированности системы k_0. Здесь возможно три случая: а) $k > k_0$, при этом делается заключение о наличии в системе коррупции (коррумпированность системы); б) $k = 0$ (k практически равен 0, при этом делается заключение о полном (практически полном) отсутствии в системе коррупции (полная или практически полная бескоррумпированность системы); в) $0 < k \leqslant k_0$, при этом делается заключение о недостаточности имеющейся информации для заключений о наличии либо об отсутствии коррупции в системе.

Изложенный метод позволяет обнаружить коррупцию в работе организационной системы в целом, но не в работе отдельных частей этой системы и тем более не в работе отдельных элементов этой системы — экспертов. Последнее представляет собой особую задачу коррумметрии — задачу локализации коррупции. Необходимость рассмотрения и решения наряду с задачей обнаружения также задачи локализации коррупции связана с тем, что после обнаружения коррупции в системе возникает вопрос ответственности за коррупционные действия, а ответственность за любые действия по закону является не коллективной, а индивидуальной.

Рассмотренный подход к обнаружению коррупции в организационной системе пригоден только для систем с $m \geqslant 2$ экспертами.

4. Решение задачи локализации коррупции

Наряду с задачами 1 (обнаружение коррупции) и 2 (измерение коррупции), введенными выше в п. 1, рассмотрим теперь *задачу 3*: разработка математической модели и метода, позволяющих по имеющейся информации о работе системы с m экспертами обнаружить факт наличия коррупции в любой подсистеме с произвольным числом экспертов s, где $s \leqslant m$. Эту задачу назовем *задачей локализации коррупции*. Формализованная постановка задачи локализации коррупции в системе выглядит так. Имеется некоторая организационная система с конечным числом экспертов m, которая полагается реальной (в отличие от гипотетической системы, которая является идеализацией заданной — см. п. 1). Далее задается некоторая произвольная подсистема имеющейся системы с $s (s \leqslant m)$ экспертами. Тогда задача локализации коррупции может быть сформулирована таким образом: 1) найти объективный критерий существования коррупции в заданной подсистеме имеющейся реальной системы в виде подходящего критического значения показателя уровня коррупции, превышение которого свидетельствует о существовании коррупции в этой подсистеме; 2) построить математическую модель, позволяющую вести эффективные вычисления, необходимые для обнаружения коррупции в подсистеме.

Как следует из приведенной постановки, задача локализации коррупции принципиально не отличается от задачи ее обнаружения. Разница состоит только в размерности решаемой задачи: во втором случае эта размерность равна $m \times n$ (m — число экспертов в рассматриваемой организационной системе, n — число показателей объекта, которые оценивают эксперты), в первом случае размерность составляет $s \times n$, $s \leqslant m$ (s — число экспертов в рассматриваемой подсистеме заданной организационной системы с m экспертами, n — то же, что и во втором случае). Содержание же решаемой задачи в обоих случаях одно и то же: обнаружение коррупции в рассматриваемой системе. Таким образом, можно сказать, что локализация коррупции — это обнаружение коррупции в некоторой заданной подсистеме исходной системы, имеющей, вообще говоря, меньшее число экспертов, но то же число показателей объекта, которые оценивают эксперты. Отсюда следует, что для решения задачи локализации коррупции могут быть использованы те же методы, что и для решения задачи обнаружения коррупции (см. п. 3), при условии, что подсистема исходной системы, для которой решается задача локализации, уже задана. Таким образом, вопрос сводится к тому, как задавать подсистемы исходной системы, для которых следует решать задачу локализации коррупции. Другими словами, как разбивать исходную систему на подсистемы, чтобы в результате решения

задач локализации для каждой из подсистем 1) коррупция оказалась локализованной на множестве с заданным достаточно малым числом экспертов, 2) потребное для этого число решаемых задач локализации было минимальным.

Итак, для разбиения организационной системы на подсистемы, удовлетворяющего двум поставленным требованиям, необходимо, чтобы на каждом шаге разбиения получалось наибольшее количество информации (снималась наибольшая неопределенность) относительно распределения коррупции в системе. При этом потребное число шагов минимизируется, обеспечивая выполнение требования 2. Выполнение требования 1 обеспечивается тем, что на каждом шаге разбиения в результате уменьшения неопределенности сужается множество экспертов, на котором локализована имеющаяся в системе коррупция, так что при нужном числе шагов объем этого множества можно довести до нужного малого числа экспертов. Выбор нужного разбиения на каждом шаге проводится с учетом имеющейся начальной и получаемой далее информации о распределении коррупции в системе.

Алгоритм решения задачи локализации коррупции в системе состоит в следующем (предполагается, что предварительно была решена задача обнаружения коррупции в системе, которая подтвердила существование коррупции в этой системе).

1. С учетом имеющейся начальной информации о распределении коррупции в системе проводится разбиение имеющейся системы с m экспертами на несколько подсистем так, чтобы в каждой подсистеме оказалось не менее 2 и не более $m - 2$ экспертов.

2. Для каждой образовавшейся подсистемы с помощью алгоритма, представленного в п. 3 данной статьи, в свою очередь решается задача обнаружения коррупции. В результате множество M подсистем распадается в общем случае на три непересекающихся подмножества M_1, M_2, M_3, где M_1 включает все коррумпированные подсистемы, M_2 — все некоррумпированные (или практически некоррумпированные) подсистемы, M_3 — все подсистемы, в отношении которых при имеющейся информации нельзя сделать окончательное заключение о наличии или отсутствии коррупции.

3. Исключается из рассмотрения множество подсистем M_2 и M_3, остается для рассмотрения только множество M_1. Дальше работа проводится по отдельности с подсистемами A_1, A_2, ..., входящими в множество M_1.

4. Возврат к шагу 1, но выполняемому теперь отдельно для каждой подсистемы A_1, A_2, ... множества M_1.

Работа алгоритма заканчивается, когда очередное множество M_1 будет включать подсистемы A_1, A_2, ... с достаточно малым числом экспертов, отвечающим условиям задачи, так что останется лишь решить задачу обнаружения коррупции для каждой из указанных подсистем.

Трудоемкость приведенного алгоритма в наибольшей степени зависит от удачного разбиения организационной системы на подсистемы в процессе выполнения последовательных шагов этого алгоритма. Приведем правила разбиения для возможных типичных случаев.

Случай 1. Существует предварительная информация о том, что в изучаемой системе в точности один эксперт (неизвестно, кто) является коррупционером. В этом случае на 1-м шаге разобьем систему на две подсистемы, по возможности с равным числом экспертов. На 2-м шаге (если предварительная информация о системе верна) выделяем некоторое множество экспертов (подсистему) M_1, содержащее искомого эксперта-коррупционера, и множества экспертов (подсистемы) M_2, M_3, в которых коррупционеров нет. На 3-м шаге исключаем из дальнейшего рассмотрения подсистемы M_2, M_3, оставляя только подсистему M_1. Дальше — возврат к шагу 1, который теперь выполняется уже не со всей системой, а с ее "половиной" — подсистемой M_1. И так далее. На каждом из таких трехшаговых циклов неопределенность (число экспертов в подсистеме, заведомо содержащей коррупционера) уменьшается вдвое, что обеспечивает локализацию эксперта-коррупционера в пределах подсистемы из 2 экспертов за $\log_2 m - 1$ таких циклов, т. е. за $3(\log_2 m - 1)$ шагов алгоритма, где m — число экспертов в системе. Это — самая экономная реализация алгоритма локализации коррупции в рассматриваемом случае, достигнутая благодаря оптимизации разбиения организационной системы на соответствующих шагах алгоритма. (Если предварительная информация о системе была неверна, то сокращение неопределенности вдвое за один цикл не происходит, и требуемое число шагов алгоритма увеличивается.)

Случай 2. Есть предварительная информация о том, что в рассматриваемой системе все m экспертов — коррупционеры. Тогда на первом шаге алгоритма мы разобьем систему на $m/2$ подсистем с (по возможности) только 2 экспертами в каждой. На 2-м шаге (если предварительная информация верна) получаем множество M_1 подсистем с 2 экспертами, содержащих коррупционеров, и пустые множества M_2 и M_3 подсистем, не содержащих коррупционеров. Потребности в выполнении 3-го шага нет, ввиду отсутствия множеств M_2 и M_3.

Локализация m коррупционеров в пределах $m/2$ подсистем из 2 экспертов выполнена. Число потребных для этого шагов оказалось равным 2, но на 2 шаге потребовалось $m/2$ операций обнаружения коррупции в $m/2$ подсистемах, таким образом, общее необходимое число операционных шагов: $1 + m/2$. Это число — минимальное, полученное благодаря оптимальному разбиению системы на 1-м шаге алгоритма. (Если же предварительная информация была неверна, т. е. реально только часть экспертов коррумпированы, то в этом случае можно было предложить лучшее разбиение системы, ведущее к уменьшению общего необходимого числа операционных шагов алгоритма.)

Случай 3. Имеется предварительная информация о том, что в рассматриваемой системе ровно два эксперта (неизвестно кто) коррумпированы. В этом случае на 1-м шаге алгоритма разбиваем систему на две подсистемы с возможно более равным числом экспертов — как в случае 1. На втором шаге в худшем случае с точки зрения получающейся неопределенности (если предварительная информация о системе истинна) будем иметь множество M_1 из двух указанных подсистем, каждая из которых коррумпирована (в нашем случае — содержит по 1 коррумпированному эксперту) и два пустых множества M_2, M_3 подсистем, не содержащих коррупционеров. 3-й шаг алгоритма отсутствует ввиду отсутствия множеств M_2, M_3. Дальше — возврат к шагу 1, который теперь выполняется уже не со всей системой, а с каждой из двух полученных на 2-м шаге подсистем. Причем, так как обе подсистемы содержат ровно по одному коррумпированному эксперту, работаем в соответствии с процедурой, описанной в случае 1. Трудоемкость локализации коррупции в каждой подсистеме будет $3\left(\log_2 \frac{m}{2} - 1\right) = 3(\log_2 m - 2)$ шагов алгоритма, так что общая трудоемкость, с учетом трудозатрат на 2-м шаге, равна $2 + 2 \cdot 3(\log_2 m - 2) = 6\log_2 m - 10$.

В общем случае правила разбиения системы конструируются аналогично правилам, представленным выше для трех типичных случаев. При этом каждый новый изучаемый случай по возможности сводится к уже рассмотренному, подобно тому, как третий случай был сведен к первому случаю. При этом надо иметь в виду, что выигрыш от минимальной трудоемкости алгоритма локализации коррупции, полученный благодаря оптимальному разбиению системы, является существенным только в экспертных системах с достаточно большим числом членов m ($m \geqslant 5 \div 7$). Если же это число мало, как часто бывает на практике ($m = 2, ..., 4$), то реального выигрыша не получается, и потому целесообразно выбирать самые простые правила разбиения, например, те что описаны в случаях 1, 2.

Изложенный подход к локализации коррупции в организационной системе позволяет локализовать коррупцию лишь с точностью до подсистем, содержащих два эксперта. Другими словами, можно указать коррумпированную пару экспертов, но точно сказать, кто именно из них коррупционер, нельзя. Для того чтобы это стало возможным, дополним вышеизложенный подход приемом "сравнение двух экспертов". Рассмотрим матрицу экспертных оценок организационной системы с $m = 2$ экспертами:

$$B = \begin{Vmatrix} b_{11} & \dots & b_{1n} \\ b_{21} & \dots & b_{2n} \end{Vmatrix}. \qquad (15)$$

Как видно из матрицы (15), средние по всем n показателям оценки объекта, даваемые 1-м и 2-м экспертами (усреднение предполагает соизмеримость форм оценок различных показателей), можно записать в виде

$$b_{1,\text{ср}} = \sum_{j=1}^{n} b_{1j}/n; \quad b_{2,\text{ср}} = \sum_{j=1}^{n} b_{2j}/n. \qquad (16)$$

Если оба эксперта не только высококвалифицированные, но и честные, добросовестные и независимые, то оценки b_1 и b_2 должны совпадать или практически совпадать. Если эксперты честные, добросовестные и независимые, но не в высшей мере квалифицированные, эти оценки будут различаться. Наконец, если эксперты нечестные, недобросовестные и зависимые, т. е. коррумпированные, то при любой их квалификации эти оценки будут различаться существенно. Эти соображения подсказывают следующий простой прием выявления заведомо коррумпированного эксперта из системы двух экспертов, в которой ранее была обнаружена коррупция:

1. По формулам (16) вычисляются средние экспертные оценки объекта b_1 и b_2, даваемые 1-м и 2-м экспертами.

2. Вычисляется относительное расхождение между оценками $b_{1,\text{ ср}}$ и $b_{2,\text{ ср}}$:

$$\delta = |b_{1,\text{ ср}} - b_{2,\text{ ср}}|/\min(b_{1,\text{ ср}}, b_{2,\text{ ср}}). \qquad (17)$$

3. Назначается некоторое пороговое достаточно малое значение δ_0 показателя δ, превышение которого можно обоснованно трактовать как объективное свидетельство коррумпированности одного из двух экспертов. Тогда, если окажется $\delta > \delta_0$, то будем считать, что один из

экспертов коррумпирован. Кого именно считать коррумпированным в случае такого превышения, зависит от смысла показателей b_{ij} и оценок b_1, b_2. Если бóльшим значениям показателей и их оценок соответствует более высокое качество оцениваемого объекта, то коррумпированным следует считать того эксперта, который занижает оценку объекта, т. е. дает меньшую из оценок b_1, b_2. Здесь речь идет об основной ситуации *А*, где эксперт не связан с командой, стоящей за объектом, и потому заинтересован в "провале чужого объекта". В двойственной ситуации *Б*, где эксперт заодно с командой объекта, он заинтересован в "вытягивании своего объекта", поэтому здесь коррумпированным нужно считать эксперта, давшего бóльшую из оценок b_1, b_2. Выделение одного из двух, заведомо коррумпированного, эксперта не означает, что второй эксперт некоррумпирован. Однако вопрос о его возможной коррумпированности должен решаться уже иначе — на основании только информации о работе данного эксперта.

Изложим теперь еще один (упрощенный) вариант рассмотренного выше приема. Пусть оценки 1-го эксперта системы с двумя экспертами доминируют над оценками 2-го эксперта, т. е. строки матрицы экспертных оценок системы (15) находятся в отношении

$$b_{1j} \geqslant b_{2j}, \quad j = \overline{1,n}, \tag{18}$$

где хотя бы одно из n неравенств (18) является строгим (т. е. имеет знак >). Тогда очевидно, что при достаточно большом проценте (например, свыше 5÷10 %) строгих неравенств в системе неравенств (18) коррумпированным следует считать: 1-го эксперта — в ситуации *А* и 2-го — в ситуации *Б*.

Наконец, о возможной локализации коррупции в одном отдельно взятом эксперте на основании исключительно информации о работе данного эксперта. Последнее означает, что нам известна только некоторая i-я строка матрицы экспертных оценок *B*, где i — номер этого эксперта. Иными словами, нам известны только оценки, которые выставляет различным показателям анализируемого объекта подозреваемый эксперт, но неизвестны оценки других экспертов. Таким образом, в этом случае решение задачи локализации коррупции на основе сравнения оценок различных экспертов, как это делалось выше, невозможно. Но поставленную задачу все же можно решить. Для этого нужно только в формуляре, содержащем выставленные экспертом оценки, выделить логические следствия вида

$$\{d_{j_1 k}, d_{j_2 s}, \ldots, d_{j_p l}\} \Rightarrow d_{j_q t}. \tag{19}$$

Следствие (19) означает, что исходя из логики и здравого смысла любой эксперт, оценивший j_1-й показатель объекта оценкой k, j_2-й показатель оценкой s, ..., j_p-й — оценкой l, обязан оценить j_q-й показатель оценкой t. Если, например, эксперт, оценивающий представленный на конкурс проект, поставил ему высшие возможные оценки по показателям "Научный интерес цели исследования", "Разработка новых методов исследования", "Новизна и оригинальность решения", "Важность результата для дальнейшего развития науки", "Наличие научного задела", "Адекватность потенциала коллектива поставленной задаче", то он обязан поставить такую же оценку по итоговому показателю "Достоин ли проект присуждения гранта". Если он этого не делает, значит, он коррумпирован, более того, озабочен своей деятельностью в данном направлении настолько, что даже потерял бдительность. Считать, что подобные действия экспертов происходят из-за их недостаточной квалификации, невозможно, поскольку логически грамотные заключения, подобные приведенному, доступны даже школьникам.

Возможны и другие подходы к локализации одного коррумпированного эксперта. Например, если у нас нет никакой информации об истинном значении оцениваемых параметров, то можно просто вычислить некое "среднее" значение каждого из оцениваемых параметров на основе оценок всех экспертов и определить тех экспертов (например, введя порог отличия), чьи оценки сильно отличаются от данных средних значений — таких экспертов можно подозревать в коррупции. Если же у нас есть собственное представление об оцениваемых параметрах, то мы можем сравнить оценки экспертов с ним и также выделить коррупционеров.

5. Задача измерения и обнаружения коррупции при сложном объекте

Усложним теперь задачи, поставленные в п. 1 и рассмотренные нами ранее в пп. 2—4. А именно, пусть m экспертов, образующих организационную систему, проводят совместную экспертизу не одного (как считалось раньше), а $N (N \geqslant 1)$ объектов. Это новая, более сложная постановка задач измерения и обнаружения коррупции в системе, о которой можно сказать, что она предназначена для экспертизы сложного (многокомпонентного) объекта. Дальше мы рассмотрим раздельно два различных возможных случая.

Случай 1. Каждый из N имеющихся объектов, образующих в совокупности сложный (многокомпонентный) объект, подвергается экспертизе

экспертами системы так же, как в прежних постановках подвергался экспертизе единственный имевшийся объект. Иными словами, каждый i-й эксперт, $i = \overline{1,m}$, оценивает каждый j-й показатель k-го объекта, $j = \overline{1,n_k}$, $k = \overline{1,N}$, при этом выбирая одно из r_{jk} возможных значений этого показателя, задаваемых множеством значений

$$A_{jk} = \{a^k_{j1}, a^k_{j2}, \ldots, a^k_{jr_{jk}}\},\ j = \overline{1,n_k},\ k = \overline{1,N}. \quad (20)$$

В результате проведения этой экспертизы получается набор матриц экспертных оценок

$$B_k = \left\| \begin{matrix} b^k_{11} & \ldots & b^k_{1n_k} \\ \ldots\ldots & & \ldots\ldots \\ b^k_{m1} & \ldots & b^k_{mn_k} \end{matrix} \right\|,\ k = \overline{1,N}, \quad (21)$$

где b^k_{ij}, $i = \overline{1,m}$, $j = \overline{1,n_k}$, — экспертная оценка, данная i-м экспертом j-му показателю k-го объекта. Таким образом, матрица $B_k = \left\| b^k_{ij} \right\|$ есть матрица экспертных оценок всеми m экспертами k-го объекта. Соединив эти матрицы для всех возможных значений k (всех объектов), получим объединенную матрицу экспертных оценок системы в виде

$$B = \| B_1 B_2 \ldots B_N \| = \left\| \begin{matrix} b^1_{11} \ldots b^1_{1n_1} \\ \ldots\ldots\ldots\ldots \\ b^1_{m1} \ldots b^1_{mn_1} \end{matrix} \begin{matrix} b^2_{11} \ldots b^2_{1n_2} \\ \ldots\ldots\ldots\ldots \\ b^2_{m1} \ldots b^2_{mn_2} \end{matrix} \ldots \begin{matrix} b^N_{11} \ldots b^N_{1n_N} \\ \ldots\ldots\ldots\ldots \\ b^N_{m1} \ldots b^N_{mn_N} \end{matrix} \right\|, \quad (22)$$

показывающую оценки всех показателей всех объектов всеми экспертами. В матрице B элементы j-го столбца k-й слева подматрицы B_k выбираются экспертами из множества A_{jk}, определяемого выражением (20). Сравнивая матрицу (2) с матрицей (22), видим, что по смыслу они равноценны, так как в той и другой любая i-я строчка содержит оценки, данные i-м экспертом всем показателям всех N оцениваемых объектов, только в первом случае имеется $N = 1$ объект, а во втором N произвольно. Таким образом, разница между организационными системами, которые рассматривались в пп. 1—4 ($N = 1$), и системами, рассматриваемыми теперь (N произвольно), лишь в их размерности: прежде матрица экспертных оценок системы B имела размерность $m \times n$, а теперь — $m \times M$, где $M = \sum_{j=1}^{N} n_j$.

Таким образом, рассматриваемую организационную систему, работающую с совокупностью N оцениваемых объектов, можно рассматривать как уже изученную выше в пп. 1—4 систему, работающую с одним оцениваемым объектом, если в качестве матрицы экспертных оценок системы брать не матрицу (2), а матрицу (22). Отсюда следует, что для решения задач измерения, обнаружения и локализации коррупции в системах с несколькими оцениваемыми объектами можно использовать методы решения этих задач для систем с одним объектом, изложенные выше в пп. 1—4.

Заметим еще, что в ситуации, когда каждый из N объектов оценивается лишь по одному показателю ($n_1 = n_2 = \ldots = n_N$), матрица экспертных оценок (22) принимает вид

$$B = \left\| \begin{matrix} b^1_{11} & b^2_{11} & \ldots & b^N_{11} \\ \ldots\ldots & \ldots\ldots & & \ldots\ldots \\ b^1_{m1} & b^2_{m1} & \ldots & b^N_{m1} \end{matrix} \right\| = \left\| \begin{matrix} b'_{11} & b'_{12} & \ldots & b'_{1N} \\ \ldots\ldots & \ldots\ldots & & \ldots\ldots \\ b'_{m1} & b'_{m2} & \ldots & b'_{mN} \end{matrix} \right\|, \quad (23)$$

где $b'_{ij} = b_{i1}$,

который, по сути, не отличается от матрицы экспертных оценок для системы с одним объектом (2). Это естественно, поскольку суть выполняемой работы в ситуации оценки нескольких показателей одного объекта и в ситуации оценки нескольких объектов по одному показателю в каждом одна и та же.

Случай 2. Результаты работы экспертов организационной системы в виде матрицы (22) экспертных оценок всех показателей всех объектов всеми экспертами неизвестны. Известна, однако, более ограниченная информация об этих результатах, полученная агрегированием полной информации, содержащейся в матрице экспертных оценок, а именно коллективная оценка каждого из N имеющихся объектов, принадлежащая собранию экспертов данной системы. Вопрос заключается в следующем: можно ли по указанной известной информации решать задачи обнаружения и измерения коррупции в нашей системе, используя соответствующие объективные показатели уровня коррупции и объективные критерии ее существования, и как это делать? (Задача локализации коррупции в системе здесь не упоминается, поскольку очевидно, что в условиях отсутствия индивидуальных оценок различных объектов различными экспертами эта задача не может быть решена.) Поставленный вопрос может быть решен положительно с помощью так называемого метода рассечений, который излагается ниже.

Будем называть *рассечением множества объектов* такое его разбиение на непересекающиеся подмножества, которое проведено по признакам, не связанным с показателями, по которым эксперты оценивают эти объекты. Множество студентов, которые сдают экзамен преподавателю, можно, например, разбить на несколько подмножеств по религиозной принадлежности (атеисты, православные,

мусульмане и т. д.), и этот признак не связан с показателем "уровень знаний", по которому преподаватель оценивает студентов. Тогда проведенное разбиение является рассечением. Аналогично, множество поданных на конкурс проектов можно разбить на подмножества по признаку ведомственной принадлежности их заявителей, отношению заявителей к членам конкурсной комиссии и т. д.

Идея метода рассечений достаточно проста. Предположим, что эксперты нашей организационной системы совместно выставили некоторую индивидуальную оценку каждому из N имеющихся объектов. Полученному набору индивидуальных оценок всех объектов соответствует некоторая интегральная оценка всего множества объектов, например, среднее арифметическое индивидуальных оценок отдельных объектов. Возьмем какое-либо рассечение множества объектов и подсчитаем для каждого его подмножества принятую нами интегральную оценку. Если эксперты некоррумпированы, полученные оценки практически совпадут с аналогичной оценкой всего множества объектов — ведь рассечение множества объектов на подмножества проводилось по признакам, не связанным с оцениваемыми показателями объектов. Так, средние арифметические экзаменационных оценок студентов среди атеистов и православных будут практически совпадать между собой и со средней арифметической оценкой по всем студентам. Если же такого совпадения не наблюдается, и интегральные оценки, найденные для отдельных подмножеств объектов, существенно отличаются от такой оценки для всего множества объектов, это свидетельствует о наличии коррупции среди экспертов.

На основе сказанного ранее можно построить следующий очевидный алгоритм метода рассечений, позволяющий обнаружить коррупцию и измерить ее уровень в организационной системе.

1. Выбор некоторого подходящего рассечения имеющегося множества объектов, оцениваемых экспертами организационной системы. Желательно, чтобы это рассечение было максимально эффективным, т. е. в наибольшей степени позволяло обнаружить и измерить уровень коррупции в системе.

2. Определение некоторой подходящей интегральной оценки показателя произвольного множества объектов, объединяющей (интегрирующей) индивидуальные оценки показателей отдельных объектов, выставленные им экспертами системы.

3. Выбор предельно допустимого отклонения между значениями интегральной оценки показателя двух множеств объектов, превышение которого свидетельствует о большом различии между наборами индивидуальных оценок показателей, которые эксперты проставили в этих двух множествах.

4. Нахождение интегральной оценки показателя нашего множества объектов на основе данных об индивидуальных оценках показателей, поставленных им экспертами.

5. Вычисление интегральной оценки показателя каждого из подмножеств объектов рассечения, выбранного на первом шаге, с использованием данных об индивидуальных оценках показателей всех объектов системы, поставленных им экспертами.

6. Сравнение интегральных оценок показателя для всего имеющегося множества объектов и отдельных его подмножеств, входящих в выбранное на шаге 1 его рассечение. Если все отклонения между оценками не превышают предельно допустимого, установленного на шаге 3, делается вывод об отсутствии коррупции в анализируемой организационной системе. Конец алгоритма. Если хотя бы одно отклонение превышает указанное предельно допустимое, делается вывод о наличии коррупции в системе и делается переход к шагу 7.

7. Используя вычисленные на шаге 6 отклонения между интегральными оценками показателя для всего имеющегося множества объектов и отдельных его подмножеств, входящих в выбранное его рассечение, подсчитываем значение показателя уровня коррупции в системе. В качестве такого показателя рекомендуется брать максимальное относительное отклонение между интегральными оценками показателя следующих множеств: всего множества объектов и отдельных его подмножеств, входящих в выбранное его рассечение. Конец алгоритма.

Наиболее трудным в вышеизложенном алгоритме является выполнение шага 1 — выбор рассечения множества объектов, поскольку для этого не существует никакой формализованной процедуры, а перебор всех вариантов рассечения имеющегося множества объектов из-за большой трудоемкости практически невозможен. К счастью, на практике коррупционеры, работающие в качестве экспертов в организационных системах, почти всегда оставляют выразительные следы, подсказывающие нужное рассечение, так что остается лишь прислушаться к этим подсказкам.

6. Примеры обнаружения, измерения и локализации коррупции

Ниже рассмотрены два примера решения задач обнаружения, измерения и локализации

Таблица 2

№	Название показателя	Возможные оценки и баллы	Поставленные оценки	
			1 эксперт	2 эксперт
1	Ясность формулировки научного содержания проекта	Предельно ясно — 1; Достаточно ясно — 0; Неясно — "—"	0	0
2	Представляет ли научный интерес цель исследования	Безусловно, да — 1; Да, в известной степени — 0; Нет — "—"	1	0
3	Предполагается ли разработка новых методов исследования	Да — 1; Нет — 0	1	0
4	Наличие новизны предлагаемого подхода и оригинальности решения	Да — 1; Нет — 0	1	1
5	Важность результата исследований	Важен для дальнейшего развития науки — 1; Представляет только самостоятельный интерес — 0	1	0
6	Возможно ли применение результатов исследований в учебном процессе	Да — 1; Нет — 0	0	0
7	Возможно ли применение результатов исследований в прикладных областях	Да — 1; Нет — 0	1	1
8	Есть ли научный задел по теме проекта	Имеется, есть публикации — 2; Имеется, публикаций нет — 1; В заявке нет данных — 0	2	2
9	Соответствует ли потенциал коллектива уровню поставленной задачи	Да, безусловно — 2; Да, в значительной мере — 1; Нет, не соответствует — 0	2	1
10	Достоин ли проект присуждения гранта	Да, безусловно — 2; Да, в значительной мере — 1; При возможности — 0; Нет — "—"	1	1

коррупции в организационных системах с помощью предложенных методов. Оба примера реальные, взятые из практики работы организационных систем. Все приведенные в них события и количественные данные подлинные. Изменены только названия учреждений, в которых происходили эти события, и названия объектов, подвергавшихся экспертизе в рассматриваемых организационных системах.

Пример 1. В 2003 г. на конкурс грантов Всероссийского научного фонда "Честная наука" Тьмутараканским государственным техническим университетом был представлен проект "Математические методы анализа процессов в условиях неопределенности". Проект был отвергнут фондом. Но по просьбе руководителя проекта, не согласившегося с таким решением, фонд прислал две эксперт-анкеты, содержавшие результаты экспертизы этого проекта двумя экспертами. Фонд отклонил проект на основе этой экспертизы. Эксперт-анкеты приведены в табл. 2.

Здесь мы имеем организационную систему из $m = 2$ экспертов, оценивающих один объект — представленный на конкурс проект, оценка которого происходит по $n = 10$ показателям. В соответствии с этим мы можем применить общую методику измерения, обнаружения и локализации коррупции в системе (пп. 2—4). Прежде всего, представим результаты работы, заданные табл. 2, в стандартной форме матрицы экспертных оценок (2):

$$B = \begin{Vmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 1 & 2 & 2 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 2 & 1 & 1 \end{Vmatrix}.$$

Теперь по первой формуле (14) мы можем вычислить показатель k относительного уровня коррупции в системе. В нашем случае входящие в эту формулу нижняя $a_{j\min}$ и верхняя $a_{j\max}$ границы диапазона возможных значений j-го столбца B (т. е. показателя в j-й строке табл. 2) равны:

$$a_{1\min} = 0,\ a_{1\max} = 1;\ a_{2\min} = 0,\ a_{2\max} = 1;\ a_{3\min} = 0,\ a_{3\max} = 1;\ a_{4\min} = 0,\ a_{4\max} = 1;\ a_{5\min} = 0,\ a_{5\max} = 1;\ a_{6\min} = 0,\ a_{6\max} = 1;\ a_{7\min} = 0,\ a_{7\max} = 1;\ a_{8\min} = 0,\ a_{8\max} = 2;\ a_{9\min} = 0,\ a_{9\max} = 2;\ a_{10\min} = 0,\ a_{10\max} = 2.$$

В результате вычисления получаем

$$k = \frac{|0-0|+|1-0|+|1-0|+|1-1|+|1-0|+|0-0|+|1-1|+|2-2|+|2-1|+|1-1|}{(1-0)+(1-0)+(1-0)+(1-0)+(1-0)+(1-0)+(2-0)+(2-0)+(2-0)} = \frac{4}{13} \cong 0,308 \cong 31\%.$$

Итак, показатель относительного уровня коррупции в системе — 31 %, что, безусловно, очень много.

Методом п. 4 решим задачу обнаружения коррупции в системе. Выберем в качестве порога коррумпированности системы следующее значение показателя k относительного уровня коррупции в системе: $k_0 = 5$ %. Тогда, видя то, что реальное значение показателя $k > k_0$, делаем заключение о наличии коррупции в системе. Более того, поскольку $k \gg k_0$ ($k/k_0 = 6{,}2$, т. е. имеем более чем шестикратное превышение допустимого уровня коррупции), мы должны признать, что уровень коррупции в системе недопустимо большой.

Теперь методом п. 4 решим задачу локализации коррупции в системе, а именно, определим, кто из двух имеющихся в системе экспертов коррумпирован. Воспользуемся приемом "сравнение двух экспертов":

1) по формулам (16), используя матрицу B, вычислим средние по всем 10 показателям оценки проекта, данные 1-м и 2-м экспертами:

$$b_{1,\text{ср}} = \sum_{j=1}^{10} b_{1j}/10 = (1 \cdot 6 + 2 \cdot 2)/10 = 1{,}0;$$

$$b_{2,\text{ср}} = \sum_{j=1}^{10} b_{2j}/10 = (1 \cdot 4 + 2 \cdot 1)/10 = 0{,}6;$$

2) по формуле (17) вычисляем относительное расхождение между найденными оценками:

$$\delta = |b_{1,\text{ ср}} - b_{2,\text{ ср}}|/\min(b_{1,\text{ ср}}, b_{2,\text{ ср}}) = \\ = (1{,}0 - 0{,}6)/0{,}6 \cong 0{,}666 = 66{,}6\ \%.$$

Это очень большое расхождение, свидетельствующее о том, что эксперты оценивали один и тот же проект по различным стандартам;

3) назначаем пороговое значение δ_0 показателя δ, превышение которого будем трактовать как свидетельство коррумпированности одного эксперта. Возьмем $\delta_0 = 5$ %. Тогда имеем $\delta = 66{,}6\ \% > 5\ \% = \delta_0$, т. е. $\delta > \delta_0$. Поэтому заключаем, что один из двух экспертов коррумпирован — тот, который давал более низкие оценки показателям проекта и, как следствие, более низкую среднюю оценку. Это эксперт 2. Основанием данного заключения служит информация, что эксперт 2 не связан с оцениваемым проектом, поэтому его коррумпированность может проявляться только в снижении даваемой проекту оценки с целью его провала. А как же эксперт 1, быть может, хоть он остался честным? Проверим его работу с помощью некоторого выделенного из его формуляра оценок (табл. 2) логического следствия типа (19), в качестве такого следствия возьмем очевидное утверждение

$$\{d_{2,1}, d_{3,1}, d_{4,1}, d_{5,1}, d_{7,1}, d_{8,2}, d_{9,2}\} \Rightarrow d_{10,2},$$

у которого в левой части стоят поставленные экспертом 1 высшие возможные оценки 2-го, 3-го, 4-го, 5-го, 7-го, 8-го и 9-го показателей проекта, а в правой части — логически вытекающая из них высшая возможная оценка по итоговому 10-му показателю. Однако первый эксперт не выполнил этого элементарного требования логики и вместо положенной заключительной оценки $d_{10,2}$ поставил оценку $d_{10,1}$, занизив оценку итогового 10-го показателя примерно вдвое. Поэтому его, как и второго эксперта, следует считать коррумпированным, хотя, возможно, и не в большой степени, поскольку большинство неитоговых показателей он не занизил.

Итак, наугад выбранный нами проект, поданный на конкурс грантов Всероссийского научного фонда "Честная наука", оказался на проверке у пары экспертов, которые должны быть признаны коррумпированными. Читатель, чувствующий статистические закономерности, вероятно, согласится с тем фактом, что теперь, по крайней мере, первое слово в названии фонда должно быть поставлено под сомнение.

Пример 2. В 2006 г. в одном из российских вузов — Тьмутараканском государственном техническом университете — был проведен конкурс грантов на научные исследования. На этот конкурс было представлено 20 проектов. Конкурсная комиссия из 10 членов провела конкурс в два тура. На первом туре комиссия оценивала представленные проекты на основе полученной проектной документации, давая коллективную оценку каждого из 20 проектов. После ранжирования всех проектов по полученным ими оценкам 12 лучших заявок были пропущены во второй тур. На втором туре эти 12 проектов оценивались комиссией по результатам выступлений перед ней руководителей проектов. По полученным оценкам проекты снова ранжировались, и 8 лучших из них были объявлены победителями конкурса, а выделенный денежный фонд был поделен между победителями в соответствии с полученными ими оценками. Так что схема работы конкурсной комиссии была замечательно проста и очевидна. Тем не менее одна ставшая известной деталь конкурса вызвала серьезное сомнение: среди 20 представленных работ пять принадлежали самим членам комиссии, а среди восьми проектов-победителей присутствовали те же пять проектов. Поэтому, не останавливаясь на

вопросах этики, детально проанализируем результаты описанного конкурса.

Таким образом, имеем систему из $m = 10$ экспертов (членов комиссии), оценивающих $N = 20$ объектов — представленные на данный конкурс проекты. При этом, хотя каждый проект оценивался по некоторому набору показателей, полная матрица экспертных оценок всех показателей всех проектов всеми экспертами вида (22) неизвестна. Но известна более ограниченная информация о результатах экспертизы, полученная путем агрегирования полной информации из матрицы экспертных оценок — коллективная оценка экспертами каждого проекта в целом в форме его пропуска или непропуска в следующий тур либо в число победителей. В соответствии с этим применим для анализа результатов конкурса методику измерения и обнаружения коррупции в системах со сложными объектами, случай 2 (п. 5) и используем описанный там семишаговый алгоритм:

1) в качестве подходящего рассечения имеющегося множества $П$ проектов, поданных на конкурс, берем его разложение на подмножества $П_{сотр}$ и $П_{чл}$ проектов, поданных соответственно простыми сотрудниками и членами комиссии. Это рассечение, очевидно, максимально эффективное для задач обнаружения и измерения коррупции в системе;

2) в качестве подходящей интегральной оценки произвольного множества проектов мы возьмем долю этих проектов, включенную экспертами в число победителей конкурса;

3) выбираем предельно допустимое отклонение δ между значениями интегральной оценки двух множеств, превышение которого означает существенное различие, с точки зрения экспертов, уровня проектов выбранных множеств: $\delta = 5$ %;

4) и 5) вычисляем интегральные оценки $И$ множеств $П$, $П_{сотр}$, $П_{чл}$. По условиям задачи из 20 поданных на конкурс проектов пять принадлежало членам комиссии, а $20 - 5 = 15$ — простым сотрудникам. До статуса "победителя конкурса" дошли соответственно 8 (5 и 3) проекта. Интегральные оценки приняли следующие значения: $И(П) = 8/20 = 0{,}4 = 40$ %, $И(П_{сотр}) = 3/15 = 0{,}2 = 20$ %, $И(П_{чл}) = 5/5 = 1 = 100$ %;

6) сравнение интегральных оценок, которые вычислены на шагах 4 и 5, путем подсчета их относительных отклонений:

$$[И(П) - И(П_{сотр})]/И(П_{сотр}) = (40 - 20)/20 = 1 = 100\ \%;$$

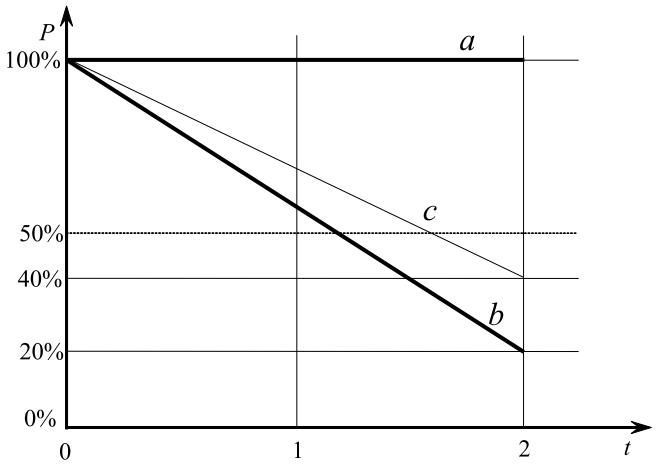
$$[И(П_{чл}) - И(П)]/И(П) = (100 - 40)/40 = 1{,}5 = 150\ \%;$$

$$[И(П_{чл}) - И(П_{сотр})]/И(П_{сотр}) = (100 - 20)/20 = 4 = 400\ \%.$$

Таким образом, все три относительных отклонения интегральных оценок множеств $П$, $П_{сотр}$, $П_{чл}$ во много раз (20, 30 и 80) превышают предельно допустимое отклонение $\delta = 5$ %. А это значит, что в системе проведения конкурса, конкретно — в конкурсной комиссии, наверняка имелась коррупция;

7) в качестве показателя уровня коррупции в системе $К$ принимаем максимальное из относительных отклонений интегральных оценок $И(П)$, $И(П_{сотр})$, $И(П_{чл})$, вычисленных на шаге 6. Имеем $К = 400$ %. Это очень высокий уровень, он в 80 раз превышает максимально допустимый, равный 5 %.

На рисунке показаны оперативные характеристики конкурсного процесса, произошедшего в вузе. Из рисунка хорошо видно, что конкурсная комиссия использовала в работе две принципиально различные стратегии: максимально возможный отсев "чужих" проектов, поданных простыми сотрудниками (кривая *b*), и максимально возможное (здесь — 100 %) сохранение "своих" проектов, поданных самими членами конкурсной комиссии (кривая *a*). Разумеется, ни о какой подлинной экспертизе проектов речь идти не могла. Обратим внимание, что, если судить о конкурсе по усредненной характеристике всего множества проектов, поданных на конкурс (кривая *c*), то он выглядит пристойно: 40 % поданных работ получили грант. Это лишний раз доказывает нам, что полагаться на традиционную статистику типа "средняя температура по палате" нельзя.



Оперативные характеристики конкурсного процесса (0 — старт, 1 — конец 1-го тура, 2 — конец 2-го тура). Кривые показывают динамику отсева проектов (*a* — поданных членами комиссии, *b* — поданных простыми сотрудниками, *c* — в целом). *P* — доля поданных проектов, продолжающих участвовать в конкурсе

К представленному в работе алгоритму обнаружения коррупции в системах необходимо сделать некоторое замечание. А именно, полученные в результате расчета данные о системе целесообразно представить комиссии из специалистов, которая должна определить, насколько выявленный с помощью алгоритма разброс мнений указывает на наличие коррупции в системе, а насколько он характеризует естественный разброс мнений объективных экспертов.

Заключение

Коррупция — большое зло в современной жизни многих стран. В очень большой степени это относится и к России. Однако сложившееся положение не безнадежно. Коррупционеры, как бы они ни старались, всегда оставляют следы своей преступной деятельности. Остается лишь, применяя подходящие методы, включая математические, расшифровать эти следы и использовать результаты в борьбе с угрожающим нам всем злом. Для этого не обязательно дожидаться, когда в борьбу вступит государство — ему это сделать очень трудно, поскольку государевы люди — чиновники — часто сами коррумпированы. Впервые представленные в данной статье простые математические методы, основанные на детерминистском подходе, могут внести свой вклад в эту область. Предложенный подход может быть распространен на организационные системы более общего вида, чем рассмотренные в данной статье, состоящие не только из экспертов.

Список литературы

1. **Ожегов С. И.** Словарь русского языка. М.: Русский язык, 1984.
2. **Словарь** иностранных слов. М.: Русский язык, 1989.
3. **Локшина С. М.** Краткий словарь иностранных слов. М.: Русский язык, 1977.
4. **Hornby A. S.** Oxford Advances Learner's Dictionary of Current English. Oxford, 1988.
5. **Калинин Б. Ю., Калинина С. В., Сумачев Э. В.** Политолого-методологические аспекты проблем коррупции в современной России // Социология социальных трансформаций. Сб. науч. тр. Нижний Новгород: НИСОЦ, 2003.
6. **Левин В. И.** Проблема коррупции в современной России: положение и перспективы решения // Вестник Тамбовского государственного технического университета. 2004. Т. 10, № 3.

V. I. Levin, Dr. of Tech. Sci., Professor, Penza State Technological University, Penza, Russian Federation

Mathematical Models and Methods for Detecting Corruption in Organizational Systems

The problem of mathematical modeling, measurement, detection and localization of corruption is formulated in article. A model of a corrupt system is built. Mathematical methods of measuring, detecting and localizing corruption in the system are proposed. Real examples of solving these problems are given.

Keywords: *corruption, mathematical modeling, detection, localization, measurement, organizational management, examination*

DOI: 10.17587/it.26.144-158

References

1. **Ozhegov S. I.** of Russian Language, Moscow, Russkiy yazyk, 1984 [in Russian].
2. **Dictionary** of Foreign Words, Moscow, Russkiy yazyk, 1989 [in Russian].
3. **Lokshina S. M.** Brief Dictionary of Foreign Words, Moscow, Russkiy yazyk, 1977.
4. **Hornby A. S.** Oxford Advances Learner's Dictionary of Current English, Oxford, 1988.
5. **Kalinin B. Yu., Kalinina S. V., Sumachev E. V.** Political and Methodological Aspects of Corruption Problem in Modern Russia, *Sociologiya social'nyh transformaciy. Sbornik nauchnyh trudov*, Nizhniy Novgorod, NISOC, 2003 [in Russian].
6. **Levin V. I.** The Problem of Corruption in Modern Russia: Position and Perspectives of Solution, *Vestnik Tambovskogo gosudarstvennogo tehnicheskogo universiteta*, 2004, vol. 10, no. 3 [in Russian].

УДК 004.056.53 DOI: 10.17587/it.26.159-168

М. А. Черепнёв, д-р физ. мат. наук, ст. науч. сотр., e-mail: cherepniov@gmail.com,
АО "Концерн "Автоматика",
С. С. Грачева, канд. техн. наук, доц., e-mail: statgracheva@mail.ru,
Национальный исследовательский университет "Высшая школа экономики"

Решение задачи Диффи—Хеллмэна на некоторых эллиптических кривых, удовлетворяющих ГОСТ 34.10—2018

Статья посвящена криптоанализу часто используемой схемы Диффи—Хеллмэна открытого распределения ключа. Начиная со статьи [1], идея которой ранее была изложена в работах И. Семаева, значительный интерес с точки зрения атакующих криптопротоколы на эллиптических кривых стала приобретать степень расширения, или MOV-степень. В англоязычной литературе этот параметр (далее k) принято называть "embedding degree". Имеется в виду расширение поля коэффициентов эллиптической кривой, в котором содержатся все точки исходного простого порядка p. Случайное значение этого параметра приближается к значению p, что приводит к длине записи элемента соответствующего расширения не многим меньше, чем $p \cdot \log p$. В стандарте ГОСТ 34.10—2018 этот параметр предлагается брать больше 31, что позволяет использовать данное расширение, поскольку длина записи его элементов не больше $k \cdot \log p$. В данной статье предложен полиномиальный алгоритм решения распознавательной и обычной задач Диффи—Хеллмэна, эффективный для некоторых таких кривых. Это означает, что схемы открытого распределения ключа, построенные с использованием этих кривых, являются нестойкими. Предлагаемый алгоритм основан на выборе такого спаривания, которое нетривиально определено на всех точках порядка p и может быть представлено в виде рациональной функции относительно небольшой степени. Сведение задачи Диффи—Хеллмэна к такому обращению получено в работе [2]. За основу предлагаемой конструкции взято нередуцированное спаривание Эйта, использованное в работе [19]. Предложены новые механизмы для расширения области определения рассматриваемого спаривания с помощью автоморфизма Фробениуса и сведения обращения по второму аргументу (лежащему в расширении поля коэффициентов кривой) к решению системы линейных уравнений с последующим поиском корней многочленов небольшой степени. Представлены оценки на вероятность разрешимости получаемых уравнений при взятии случайного представителя смежного класса, представляющего значение спаривания.

Ключевые слова: *схема открытого распределения ключа, эллиптические кривые, задача Диффи—Хеллмэна, спаривание Тэйта, спаривание Эйта, автоморфизм Фробениуса*

Введение

Для построения современных схем защиты информации довольно часто используют группу точек на эллиптической кривой. Составной частью многих схем шифрования является протокол открытого распределения ключа Диффи—Хеллмэна. В данной работе получены условия на размер основного поля и порядок используемой группы точек существующей эллиптической кривой, достаточные для решения на ней задачи Диффи—Хеллмэна с полиномиальной сложностью. Отметим, что при псевдослучайном характере построения эллиптических кривых вероятность, с которой выполняются указанные условия, пренебрежимо мала. Вместе с тем некоторые кривые с малой MOV-степенью построены в работе [8], хотя в общем случае задача построения таких кривых пока не решена.

Полиномиальные алгоритмы, решающие ту же задачу для некоторых суперсингулярных кривых со степенью расширения, равной 2 или 3, предложены в работе [3]. Быстрые алгоритмы обращения алгоритма Миллера предложены для некоторых кривых в работах [4—7] в случаях, когда степень расширения не превосходит 12, либо в работе [11] для случая, когда имеется невырожденное спаривание, выражающееся рациональной функцией малой

степени с однозначно определенными значениями в конечном поле (без доказательства его существования). В данной работе предложен полиномиальный алгоритм решения задачи Диффи—Хеллмэна, включающий построение невырожденного спаривания, выражающегося рациональной функцией малой степени (при выполнении некоторых необременительных требований на параметры кривой), со значениями в факторгруппе конечного поля (спаривание Эйта), и обращение алгоритма Миллера для такого спаривания для некоторых кривых с полиномиальной границей на степень расширения.

Рассмотрим эллиптическую кривую над большим простым полем $\mathbb{F}_r$ из r элементов

$$y^2 = x^3 + ax + b,\ a,\ b \in \mathbb{F}_r.$$

Для некоторого простого, отличного от r, делителя p порядка группы $E(\mathbb{F}_r)$, состоящей из $\mathbb{F}_r$ точек этой кривой, рассмотрим все точки порядка p, координаты которых лежат в алгебраическом замыкании $\overline{\mathbb{F}}_r$. Пусть $k = ord_p r > 1$, а ord_p — порядок по модулю p. Хорошо известно [9, 16], что все эти точки образуют группу, являющуюся прямым произведением двух групп порядка p:

$$E[p] = G_1 \times G_2,$$

где $G_1 = E[p] \cap Ker(\pi_r - [1])$, $G_2 = E[p] \cap \cap Ker\{\pi_r - [r]) \in E(\overline{\mathbb{F}}_{r^k})$.

Пусть $e(G_1, G_2)$ — невырожденное билинейное спаривание, т. е. гомоморфизм, отличный от тождественной единицы. Аналогично тому, как это было сделано в работах [10, 11], можно получить следующую оценку сложности решения распознавательной задачи Диффи—Хеллмэна. Пусть (P, aP, bP, P') — элементы G_1, являющиеся входом для распознавательной задачи Диффи—Хеллмэна в группе G_1. Пусть $e(P, Q)$ для некоторого случайного аргумента $Q \in G_2$ отлично от единицы. Вычислим $\tilde{Q}$ такое, что $e(P, Q)^b = e(bP, Q) = e(P, \tilde{Q})$. Такое $\tilde{Q}$ существует, например $\tilde{Q} = bQ$. Проверим $e(aP, \tilde{Q}) = e(P, \tilde{Q})^a = e(P', Q)$. Если да, то $P' = abP$. Таким образом, имеем следующую оценку на сложность распознавательной задачи Диффи—Хеллмэна в группе G_1:

$$DDH(G_1) \leqslant I_2 + 3C,$$

где C — сложность вычисления спаривания, а I_2 — сложность обращения спаривания по второму аргументу. Аналогично

$$DDH(G_2) \leqslant I_1 + 3C,$$

а для сложности решения обычной задачи Диффи—Хеллмэна

$$DH\{G_i) \leqslant I_1 + I_2 + 2C,\ i \in \{1, 2\}.$$

Рассмотрим рациональную функцию $f_{s,Q}(x, y)$ для произвольного целого s как функцию, определенную равенством

$$div(f_{s,Q}) = s(Q) - (sQ) - (s - 1)(\infty). \qquad (1)$$

Такая функция существует согласно работе [9, Следствие III.3.5.]. В ряде случаев значение спаривания [16, 17] задается формулой

$$f_{s,Q}(D_2) = \prod_{P \in Supp D_2} f_{s,Q}(P)^{v_P(D_2)}, \qquad (2)$$

для некоторого дивизора D_2.

Значения вида $f_{s,Q}(D_2)$ и использующие их спаривания могут быть вычислены с помощью алгоритма Давенпорта [12] — Миллера [13] и его обобщений [15]. Этот алгоритм линеен относительно длины входа, поэтому сложность вычисления, например, спаривания Вейля будет $O(k\log r)$ операций в поле $\overline{\mathbb{F}}_{r^k}$, или $O(k^3\log^3 r)$ битовых операций.

Спаривание Эйта [18] выражается формулой (2) с одним множителем. Пусть $P \in G_1$ и $Q \in G_2$. Пусть

$$s = r^i (\mathrm{mod}\ p),\ 1 \leqslant i \leqslant k - 1. \qquad (3)$$

Тогда [19, Theorem 1] обобщенное спаривание Эйта

$$\tilde{e}(P, Q) = f_{s,Q}(P)$$

является невырожденным билинейным отображением $G_1 \times G_2$, если

$$\gamma_p(s^{ord_p s} - 1) \leqslant \gamma_p(r^k - 1), \qquad (4)$$

где $\gamma_p(x)$ — степень вхождения p в x.

1. Представление элементов смежных классов

Для простоты дальнейших рассуждений будем считать, что $p^2 \nmid r^k - 1$. Это, в частности, означает, что в $\mathbb{F}^*_{r^k}$ нет подгруппы порядка p^2, подгруппа порядка p единственна, а также в каждом смежном классе факторгруппы $\mathbb{F}^*_{r^k}/(\mathbb{F}^*_{r^k})^p$ существует единственный элемент порядка p. Рассмотрим этот вопрос несколько подробнее. Представим абелеву группу $\mathbb{F}^*_{r^k}$ в виде разложения на примарные группы, соот-

ветствующие различным простым делителям числа $r^k - 1 = \prod_{i=1}^{i_0} p_i^{\alpha_i}$:

$$\mathbb{F}_{r^k}^* = \bigotimes_{i=1}^{i_0} <g_i>_{p_i^{\alpha_i}},\ p_1 = p,\ \alpha_1 = 1,$$

тогда

$$(\mathbb{F}_{r^k}^*)^p = \bigotimes_{i=1}^{i_0} <g_i^p>_{p_i^{\alpha_i}} = \bigotimes_{i=2}^{i_0} <g_i>_{p_i^{\alpha_i}},$$

так как $g_1^p = 1$, $ord g_i^p = p_i^{\alpha_i}$ при $i > 1$. Таким образом, факторгруппа $\mathbb{F}_{r^k}^* / (\mathbb{F}_{r^k}^*)^p$ изоморфна циклической группе $<g_1>_p$ порядка p. Смежные классы этой факторгруппы состоят из элементов

$$g_1^{\beta_1} \prod_{i=1}^{i_0} g_i^{\beta_i},\ \beta_i = 0, 1, \ldots, p_i^{\alpha_i} - 1,$$

при фиксированном $\beta_1 \in \{0, 1, ..., p - 1\}$.

Определение 1. *Элементы вида*

$$\prod_{i=2}^{i_0} g_i^{\beta_i},\ \beta_i = 0, 1, \ldots, p_i^{\alpha_i} - 1,$$

перечисляющие смежный класс, в соответствующей факторгруппе будем называть сдвигом.

Аналогично, для произвольного натурального n

$$(\mathbb{F}_{r^k}^*)^n = \bigotimes_{i=1}^{i_0} <g_i^{p_i^{\gamma_i}}>_{p_i^{\alpha_i - \gamma_i}},$$

где $p_i^{\gamma_i} = (n, p_i^{\alpha_i})$. Таким образом, факторгруппа $\mathbb{F}_{r^k}^* / (\mathbb{F}_{r^k}^*)^n$ состоит из смежных классов

$$\prod_{i=1}^{i_0} g_i^{\delta_i} g_i^{p_i^{\gamma_i} \beta_i},$$

при фиксированных $\delta_i = 0, 1, ..., p_i^{\gamma_i} - 1$ и произвольных $\beta_i \in \{0, 1, ..., p_i^{\alpha_i - \gamma_i} - 1\}$. При этом она изоморфна циклической группе

$$\bigotimes_{i=1}^{i_0} <g_i^{p_i^{\alpha_i - \gamma_i}}>_{p_i^{\gamma_i}}.$$

Соответственно, сдвиг имеет вид

$$\prod_{i=1}^{i_0} g_i^{p_i^{\gamma_i} \beta_i},\ \beta_i = 0, 1, \ldots, p_i^{\alpha_i - \gamma_i} - 1,$$

а мощность сдвига равна

$$\#(\mathbb{F}_{r^k}^*)^n = \prod_{i=1}^{i_0} p_i^{\alpha_i - \gamma_i} = \frac{r^k - 1}{(n, r^k - 1)}.$$

Все решения уравнения $X^n = 1$ в $\mathbb{F}_{r^k}^*$ будут иметь вид

$$\prod_{i=1}^{i_0} g_i^{p_i^{\alpha_i - \gamma_i} \beta_i},\ \beta_i \in \{0, 1, \ldots, p_i^{\gamma_i} - 1\},$$

а состоящая из них группа изоморфна факторгруппе $\mathbb{F}_{r^k}^* / (\mathbb{F}_{r^k}^*)^n$.

Циклической структурой мультипликативной группы поля $\mathbb{F}_{r^k}^*$ мы здесь (после определения 1) не пользовались, т. е. p_i не обязательно различны. Так что те же выкладки можно провести для аддитивной группы точек на эллиптической кривой и ее факторгруппы по n кратным точкам, которая будет изоморфна подгруппе $Ker[n]$.

2. Модельный пример

Из билинейности следует, что образом для различных видов спариваний, определенных на $G_1 \times G_2$, является некоторая группа порядка p, образованная элементами $\mathbb{F}_{r^k}$. Предположим сначала, что для спаривания Эйта на некоторой эллиптической кривой эта группа является единственной подгруппой $G \subset \mathbb{F}_{r^k}^*$ порядка p, т. е. все элементы образа этого спаривания лежат в $<g_1>_p$. Это модельный пример. Здесь мы не будем обсуждать, бывает такое или нет.

Из формулы (5) [14, с. 239] (см. также алгоритм Миллера [13, 15]) при $s = 2$ получаем

$$f_{2,Q}(P) = \frac{y_1 - \lambda(x_1 - x_2) - y_2}{x_1 - x_3},$$

где $P(x_1, y_1)$, $Q(x_2, y_2)$, $[2]Q(x_3, y_3)$, $\lambda = \dfrac{3x_2^2 + a}{2y_2}$. Поэтому обращение рассматриваемого спаривания по первому аргументу, т. е. по P — это решение системы

$$\begin{cases} \dfrac{y_1 - \lambda(x_1 - x_2) - y_2}{x_1 - x_3} = z; \\ y_1^2 = x_1^3 + a x_1 + b \end{cases} \qquad (5)$$

относительно x_1, $y_1 \in \mathbb{F}_r$ при фиксированных $x_2, y_2, x_3, y_3, \lambda \in \mathbb{F}_{r^k}$; $a, b \in \mathbb{F}_r$, $z \in G$.

Исключая y_1 из первого уравнения, получим кубическое уравнение на x_1. Решая его (это можно сделать по формулам Кардано за $O(1)$ операций в $\mathbb{F}_{r^k}$), получим не более шести точек в $E(\mathbb{F}_r)$, которые можно проверить на принадлежность к G_1, проверяя $[p]P = \infty$, не более чем за $O(\log p)$ операций в $\mathbb{F}_r$. Поскольку

ку исходное уравнение заведомо имело решение в G_1, то мы его получим. Можно поступить и иначе, заменив в системе (5) координаты точек Q, $[2]Q$ соответственно на координаты точек $[t]Q$, $[2t]Q$, а z на z^t для некоторого небольшого t. Тогда получим еще одно кубическое уравнение на x_1. Искомая точка $P \in G_1$ будет, очевидно, решением обоих уравнений. Естественно предположить, что эти уравнения будут разными, поэтому, исключая x_1^3 из двух кубических уравнений, получим квадратное уравнение на x_1. Выбрав еще одно значение для t, получим x_1.

Действуя аналогичным образом в случае небольшого $s > 2$, или в случае обращения спаривания по второму аргументу, сначала подстановкой второго уравнения исключаем из первого уравнения системы вида (5) с переменными соответственно $x' = x_1$, $y' = y_1$ или $x' = x_2$, $y' = y_2$ все степени переменной y', большие единицы. Затем в получившемся уравнении выражаем y' через x' и подставляем во второе уравнение. Получится полином от x' степени $O(s)$ (см. формулу (5) на с. 239 [14]), среди корней которого, как и раньше, найдем координату искомой точки. При этом в случае обращения по второму аргументу при отборе корней нужна дополнительная проверка $\pi_r(Q) = [r]Q$.

Вместо проверки $[p]P = \infty$ можно предложить процедуру побыстрее (схожие идеи описаны в работе [21]). Если $E(\mathbb{F}_r) = G_1 \times \tilde{G}$ для некоторой группы $\tilde{G}$ с небольшим порядком t, что обычно бывает в криптографически значимых случаях, то для получения искомой точки $P \in G_1$ надо для произвольного решения $\tilde{P} \in E(\mathbb{F}_r)$ воспользоваться тем, что $(p, t) = 1$, откуда $P = [1 + kp]\tilde{P}$ для k, удовлетворяющего $1 + kp \equiv 0 \pmod t$.

3. Свойства спаривания Тэйта

Рассмотрим теперь N спаривание Тейта $f_{N,Q}(P)$ [17] для некоторого натурального N. Гомоморфным образом рассматриваемого спаривания, определенного на $(P, Q) \in E(\mathbb{F}_{r^k})[N] \times E(\mathbb{F}_{r^k})/[N]E(\mathbb{F}_{r^k})$ является факторгруппа $G = \mathbb{F}_{r^k}^*/(\mathbb{F}_{r^k}^*)^N$. Следует иметь в виду, что, как было показано ранее, имеется изоморфизм $E(\mathbb{F}_{r^k})[N] \cong E(\mathbb{F}_{r^k})/[N]E(\mathbb{F}_{r^k})$.

Теорема 1 *(Теорема 3 [17]). Спаривание $f_{N,Q}(P)$ является билинейным и невырожденным, если поле $\mathbb{F}_{r^k}$ содержит корень N-й степени из единицы.*

Заметим, что невырожденность в этой теореме понимается в "особом" смысле, т. е. спаривание $t : A \times B \to Z$ на абелевых группах A, B, Z невырождено в "особом" смысле, если соответствующие гомоморфизмы $A \to Hom(B; Z)$ и $B \to Hom(A; Z)$ инъективны.

Свойство гомоморфизма переводить сумму точек в произведение их образов будем в дальнейшем называть линейностью. Поскольку при этом порядок точек образа делит порядок точек прообразов (в данном случае их два), то значение спаривания в указанной факторгруппе зависит только от примарных компонент области определения, для которых соответствующие простые входят в НОД$(\#E(\mathbb{F}_{r^k})[N], r^k - 1)$.

Наличие в $\mathbb{F}_{r^k}$ корня из единицы степени N или циклической группы порядка N означает, что $N = \prod_{i=0}^{i_0} p_i^{\gamma_i}$ делит $r^k - 1 = \prod_{i=0}^{i_1} p_i^{\alpha_i}$, т. е. $\alpha_i \geqslant \gamma_i$, $i = 0, 1, ..., i_0, i_1 \geqslant i_0$ (здесь все p_i различны). Для некоторого $p = p_i$, $i \leqslant i_0$ пусть g и $\bar{g}$ — образующие соответствующих примарных компонент порядка p^δ в разложении изоморфных групп $E(\mathbb{F}_{r^k})[N] \cong E(\mathbb{F}_{r^k})/[N]E(\mathbb{F}_{r^k})$. Пусть $f_{N,\bar{g}}(g) = \tilde{g} \in \mathbb{F}_{r^k}^*/(\mathbb{F}_{r^k}^*)^N$. Тогда из линейности получаем, что $ord\tilde{g} = p^\sigma$, $\delta \geqslant \sigma$ (так как единица переходит в единицу), а $\sigma \leqslant \alpha_i - \gamma_i$. Далее $f_{N,\bar{g}^a}(g^b) = \tilde{g}^{Cab}$, для некоторой константы C, где $ord\tilde{g} = p_i^{\alpha_i - \gamma_i}$. Если группа $E(\mathbb{F}_{r^k})[N]$ содержит несколько примарных компонент, отвечающих одному простому p, то значение спаривания на них будет иметь вид

$$\tilde{g}^{C_1 a_1 b_1 + \ldots + C_j a_j b_j}, \qquad (6)$$

причем из-за линейности это значение не зависит от примарных компонент аргументов рассматриваемого спаривания, отвечающих простым не равным p. Элементы примарных компонент аргументов, для которых соответствующие простые p не содержатся среди p_i могут влиять лишь на сдвиг (предположительно случайно). Таким образом, доказана следующая теорема.

Теорема 2. *Пусть $\mathbb{F}_{r^k}^*/(\mathbb{F}_{r^k}^*)^N = \Pi_i \tilde{G}_i$ — разложение в примарные компоненты $\#\tilde{G}_i = p_i^{\alpha_i}$. Рассмотрим i-ю координатную функцию рассматриваемого спаривания, отвечающую компоненте $\tilde{G}_i$, где соответствующее p_i делит $\#E(\mathbb{F}_{r^k})$. Тогда в условиях теоремы 1 она линейно, невырожденно и корректно отображает произведения элементов примарных компонент области определения, отвечающих одному и тому же простому p_i, на примарную группу $\tilde{G}_i$. Примарные компоненты области определения, соответствующие простым, не входящим в* НОД$(\#E(\mathbb{F}_{r^k})[N], r^k - 1)$ *влияют только на сдвиг*

(т. е. меняют представителя, но не класс смежности, в который попадает результат спаривания).

Корректность означает независимость от примарных компонент, отвечающих другим простым. Из теоремы 2, в частности, следует, что координатные функции, отвечающие простым множителям, не входящим в НОД(#$E(\mathbb{F}_{r^k})$[N], $r^k - 1$) тождественно равны единице. Следует также иметь в виду, что число примарных компонент в группе точек на эллиптической кривой, отвечающих одному простому числу, ограничено двумя (следствие 6.4 с. 89 [9]). Поэтому число слагаемых в показателе равенства (6) не более четырех.

Заметим, что выполнения условия существования соответствующего корня из единицы можно добиться, перейдя к расширению поля $\mathbb{F}_{r^k}$, а именно, добавив все корни многочлена X^{N-1}, т. е. $\mathbb{F}_{r^{kw}}$, где $w = ord_N r^k$. При этом, если $E(\mathbb{F}_{r^k}) \in [N]E(\mathbb{F}_{r^{kw}})$, то на аргументах из $E(\mathbb{F}_{r^k})$ рассматриваемое спаривание принимает только значение в единичном смежном классе. Однако, если в качестве N взять универсальную экспоненту группы $E(\mathbb{F}_{r^k})$, условие $E(\mathbb{F}_{r^k}) \in [N]E(\mathbb{F}_{r^{kw}})$ приводит к необходимости того, чтобы универсальная экспонента группы $E(\mathbb{F}_{r^{kw}})$ делилась на N^2. Из-за аддитивной зависимости между порядками эллиптической кривой над полем и его расширением проследить, при каких r, k это бывает, в общем виде достаточно трудно.

Для построения спаривания Тейта, определенного на всех точках $E(\mathbb{F}_{r^k})$, необходимо выбирать в качестве N кратное универсальной экспоненте группы $E(\mathbb{F}_{r^k})$. Но наличие дополнительных множителей может привести к вырождению, т. е. к увеличению объема сдвига, чего нам бы не хотелось. Проведя численные эксперименты, мы в ряде случаев убедились, что при выборе в качестве N универсальной экспоненты группы $E(\mathbb{F}_{r^k})$ спаривание Тейта не тождественно равно единице на $E(\mathbb{F}_{r^k}) \times E(\mathbb{F}_{r^k})$. Однако оно выражается рациональной функцией слишком большой степени, что не позволяет быстро его обращать.

4. Вариант спаривания с сжимающими отображениями

Пусть x — какой либо корень характеристического многочлена автоморфизма Фробениуса π_r по модулю N:

$$x^2 - tx + r \equiv 0 (\mathrm{mod}\ N),$$
$$t = r + 1 - \#(E(\mathbb{F}_r)),\ x \equiv r\ (\mathrm{mod}\ p). \qquad (7)$$

(В этом месте можно использовать также характеристическое уравнение для π_{r^l}: $x^2 - t_l x + r^l \equiv 0$ (mod N), $t_l = 1 + r^l - \#(E(\mathbb{F}_{r^l}))$). Тогда, очевидно, ($x$, N) = 1, при $r \nmid N$. Поскольку $t_l = \alpha^l + \beta^l$, где $\alpha\beta = r$, $\alpha + \beta = t$ (Теорема 2.4 [9]), то $t_k \equiv t^k$ (mod$r$), так как симметрический многочлен от α, β с целыми коэффициентами является целым числом. Подставляя t^k в выражение для t_k, получим $t^k \equiv 1 - \#(E(\mathbb{F}_{r^k}))(\mathrm{mod}r)$, откуда, ввиду того, что N и $\#(E(\mathbb{F}_{r^k}))$ состоят из одинаковых простых множителей, получим, что условие $r \nmid N$ равносильно тому, что $ord_r t \nmid k$. Это и будем в дальнейшем предполагать.

Поскольку $x - t + rx^{-1} \equiv 0 (\mathrm{mod}\ N)$, то для любой точки $Q \in E(\mathbb{F}_{r^k})$ и преобразования $\tilde{Q}(Q) = (\pi_r - [rx^{-1}])[n^{-1}\ (\mathrm{mod}\ p)][n]Q$ имеем

$$\pi_r \tilde{Q} = (\pi_r^2 - [rx^{-1}]\pi_r)\widehat{Q} =$$
$$= ([t - rx^{-1}]\pi_r - [r])\widehat{Q} = ([x]\pi_r - [r])\widehat{Q} = [x]\tilde{Q}.$$

Пусть теперь $T \equiv x^j \neq 1$ (mod N) мало при некотором j, и $\hat{t} \equiv rx^{-1}$ (mod N). Тогда, в силу (7), $\hat{t} \equiv 1$ (mod p). Следуя плану, изложенному в работе [18], построим некоторое новое спаривание. Рассмотрим отображение на $E(\mathbb{F}_{r^k})^2$, заданное равенством

$$\tilde{e}_T(Q, P) = \frac{f'_{T,Q}(trP)}{f'_{T,Q}(\infty)}, \qquad (8)$$

$$\text{где } trP = \sum_{i=0}^{k-1} \pi_r^i P \in E(\mathbb{F}_r),\ \tilde{Q} = \tilde{Q}(Q + R),$$

где $f'_{T,Q} = \dfrac{f_{T,\tilde{Q}}}{f_{T,\tilde{R}}}$ для некоторой случайной фиксированной точки $\tilde{R} \in E(\mathbb{F}_{r^k}) \cap Ker(\pi_r - [x])$, например $\tilde{R} = \tilde{Q}(R)$, $R \in E(\mathbb{F}_{r^k})$.

Для редуцированного N спаривания Тэйта имеем (см. [17] определение 2, где сделаны следующие переобозначения: $D = \tilde{Q} - \tilde{R}$, $E = trP - (\infty)$, $m = N$, $k = \mathbb{F}_{r^{kw}}$, $w = ord_N r^k$):

$$\varepsilon(\tilde{Q}, trP) = \left(\frac{f'_{N,\tilde{Q}}(trP)}{f'_{N,\tilde{Q}}(\infty)}\right)^{\frac{r^{wk}-1}{N}}$$

для любых P, $Q \in E(\mathbb{F}_{r^k})$. Пусть $v = ord_N T$, а $L \in \mathbb{N}$ определено равенством $LN = T^v - 1$. Докажем, что формула (8) задает билинейное отображение $(E(\mathbb{F}_{r^k}))^2 \to \mathbb{F}^*_{r^{kw}}/(\mathbb{F}^*_{r^{kw}})^N$, нетривиальное на $G_2 \times G_1$. Доказательство проводится по плану работы [18]. Имеем

$$\varepsilon(\tilde{Q}, trP)^L = \left(\frac{f'_{LN,\tilde{Q}}(trP)}{f'_{LN,\tilde{Q}}(\infty)}\right)^{\frac{r^{wk}-1}{N}} = \left(\frac{f'_{T^v-1,\tilde{Q}}(trP)}{f'_{T^v-1,\tilde{Q}}(\infty)}\right)^{\frac{r^{wk}-1}{N}} = \left(\frac{f'_{T^v,\tilde{Q}}(trP)}{f'_{T^v,\tilde{Q}}(\infty)}\right)^{\frac{r^{wk}-1}{N}}. \quad (9)$$

Последнее равенство верно, так как по построению $T^v \equiv 1 (\mathrm{mod}\ N)$, и

$$(T^v\tilde{Q}) = \tilde{Q},\ ((T^v - 1)\tilde{Q}) = (\infty).$$

Согласно Лемме 2 [22] имеем

$$f_{T^v,\tilde{Q}} = f_{T,\tilde{Q}}^{T^{v-1}} f_{T,T\tilde{Q}}^{T^{v-2}} \cdots f_{T,T^{v-1}\tilde{Q}}.$$

Аналогично это же будет верно с заменой $\tilde{Q}$ на $\tilde{R}$. Применяя свойства отображения $\tilde{Q}(Q)$, получим $[T]\tilde{Q} = \pi_r^j\tilde{Q}$, $[T]\tilde{R} = \pi_r^j\tilde{R}$. Заметим, что при $P \in E(\mathbb{F}_{r^k})$

$$\frac{f_{T,T^l\tilde{Q}}(trP)}{f_{T,T^l\tilde{R}}(trP)} = \frac{f_{T,\pi_r^{jl}\tilde{Q}}(trP)}{f_{T,\pi_r^{jl}\tilde{R}}(trP)} = \left(\frac{f_{T,\tilde{Q}}(trP)}{f_{T,\tilde{R}}(trP)}\right)^{r^{jl}},$$

поэтому

$$f'_{T^v,\tilde{Q}}(trP) = (f'_{T,\tilde{Q}}(trP))\sum_{l=0}^{v-1} T^{v-1-lr^{jl}} = (f'_{T,\tilde{Q}}(trP))^M,$$

$$M = \frac{r^{jv} - T^v}{r^j - T}.$$

Аналогично это же будет верно с заменой trP на (∞), так как коэффициенты кривой лежат в $\mathbb{F}_r$. Таким образом, из (9) имеем

$$(\varepsilon(\tilde{Q}, trP))^L = (\tilde{e}_T(Q,P))^{M\frac{r^{wk}-1}{N}}. \quad (10)$$

Справедливость этого равенства была проверена численно, однако дальнейшее использование рассматриваемого спаривания в нашем алгоритме при

$$(L, p) = 1$$

стало невозможным из-за слишком малой вероятности обращения уравнения

$$\tilde{e}_T(Q,P) = z\tilde{z}^N, \quad (11)$$

при случайном выборе $\tilde{z}^N \in \mathbb{F}_{r^k}$.

5. Основной вариант спаривания

В этом разделе сконструировано новое спаривание, с предположительно более высокой вероятностью обращения равенства вида (11) при случайном выборе $\tilde{z}^N \in \mathbb{F}_{r^k}$. В подтверждение этого проведено сравнение мощностей сдвига и свободных компонент аргументов.

При подстановке $N = \#E(\mathbb{F}_{r^k})$ спаривание ε для некоторых тестовых эллиптических кривых оказалось тождественно равно единице на $E(\mathbb{F}_{r^k})$. В этом примере универсальная экспонента $\Sigma = \Sigma(E(\mathbb{F}_{r^k}))$ группы точек $E(\mathbb{F}_{r^k})$ является делителем числа N. Оказалось, что если вместо N подставить универсальную экспоненту, то рассматриваемое спаривание при подстановке точек из $E(\mathbb{F}_{r^k})$ уже не является тождественной единицей. Поскольку основным объектом наших исследований далее является нередуцированное Σ спаривание Тейта, то результаты дальнейших исследований мы будем формулировать для него. Будем обозначать его ε'.

Путем применения к спариванию Тейта с универсальной экспонентой процедуры понижения степени было получено новое спаривание $\tilde{e}_T$, выражающееся рациональной функцией малой степени. Оно определено на всех точках из $E(\mathbb{F}_{r^k})$. При условии $p \nmid L$ это спаривание, как и ожидалось, оказалось невырожденным на $G_2 \times G_1$, однако вероятность разрешимости уравнений, эквивалентных его обращению, оказалась слишком низкой. Причина этого в присутствии сжимающих отображений tr, $\tilde{Q}$ в результате действия которых мощность образа рассматриваемого нередуцированного спаривания существенно меньше, чем порядок мультипликативной группы $\mathbb{F}_{r^k}^*$, в которой для решения задачи обращения предполагалось выбирать элемент с помощью случайного выбора представителя смежного класса. Таким образом, для повышения этой вероятности возникла необходимость корректировки конструкции используемого спаривания, чтобы рассматриваемая процедура понижения степени работала при условии произвольности (или почти произвольности) его аргументов в $E(\mathbb{F}_{r^k})$ и приводила к невырожденному на $G_2 \times G_1$ спариванию.

Наше желание использовать нередуцированное спаривание Тейта обосновано тем, что согласно полученной нами выше Теореме 2 оно отличается от редуцированного только наличием сдвига и, по существу ничего не меняющим, возведением в степень других компонент

результата. При этом отсутствие редуцирующей степени делает задачу обращения этого спаривания намного более простой. Мы будем пользоваться Теоремой 2 вместо Теоремы 3 [17].

Если при некоторой фиксированной компоненте из G_1 (см. Теорему 2), некотором сдвиге и фиксированном одном (первом или втором) аргументе спаривание ε' обратимо по другому аргументу, то для получения существенной компоненты этого аргумента, принадлежащей G_2 или G_1, применяется сжимающее отображение

$$[(1-r)^{-1} \pmod p](\pi - [r])\left[\left(\frac{\Sigma}{p}\right)^{-1} \pmod p\right]\left[\frac{\Sigma}{p}\right]$$

для получения компоненты из G_1 или

$$[(r-1)^{-1} \pmod p](\pi - [1])\left[\left(\frac{\Sigma}{p}\right)^{-1} \pmod p\right]\left[\frac{\Sigma}{p}\right]$$

для получения компоненты из G_2, соответственно.

Таким образом, осталось решить задачу обращения в элементы, лежащие в $E(\mathbb{F}_{r^k})$. Для этого используемое спаривание должно быть представлено рациональной функцией малой степени и принимать "почти" все возможные значения в $\mathbb{F}_{r^k}$, чтобы случайный выбор сдвига с хорошей вероятностью приводил к разрешимому уравнению.

Для сохранения свойств линейности, невырожденности и широкой области определения Σ спаривания Тейта ε' при процедуре понижения степени предлагается следующая его модификация. Пусть

$$T \equiv r^j \pmod \Sigma, \tag{12}$$

$(j, k) = k_0$. Будем считать, что величина T полиномиально зависит от $\log p$. Кривые с одновременно малыми значениями T и k получатся, если взять, например, простые p и r так, что $p|T^k - 1$, $r = p + T$ (можно перебирать T и k, проверяя на простоту получающееся r). Тогда вероятность выполнения равенства (12) представляется достаточно высокой. Кривая с такими параметрами существует по известной теореме Ваттерхауза [23], хотя алгоритма, который бы мог ее построить, пока нет.

Рассмотрим спаривание, заданное формулой

$$\prod_{l=0}^{\frac{k}{k_0}} f_{\Sigma,Q}(P^{\pi^{lk_0}}).$$

Здесь мы учли то, что в результате численных экспериментов выяснилось, что алгоритм Миллера выдает функцию $f_{N,Q}(P)$ в нормированном виде (т. е. она равна единице при подстановке бесконечной точки вместо любого из аргументов). Поэтому спаривание Тейта, например, можно записать просто $\varepsilon(Q, P) = f_{N,Q}(P)$.

Аналогично уже рассмотренной ранее процедуре понижения степени при $T^v - 1 = L\Sigma$ получим

$$\left(\prod_{l=0}^{\frac{k}{k_0}} f_{\Sigma,Q}(P^{\pi^{lk_0}})\right)^L = \prod_{l=0}^{\frac{k}{k_0}} f_{T^v,Q}(P^{\pi^{lk_0}}) =$$

$$= \prod_{l=0}^{\frac{k}{k_0}} f_{T,Q}(P^{\pi^{lk_0}}) \prod_{l=0}^{\frac{k}{k_0}} f_{T,[T]Q}(P^{\pi^{lk_0}}) \cdots$$

$$\cdots \prod_{l=0}^{\frac{k}{k_0}} f_{T,[T^{v-1}]Q}(P^{\pi^{lk_0}}) = \left(\prod_{l=0}^{\frac{k}{k_0}} f_{T,Q}(P^{\pi^{lk_0}})\right)^M$$

с тем же самым значением $M = \dfrac{r^{jv} - T^v}{r^j - T}$, но уже при всех $P \in E(\mathbb{F}^*_{r^k})$ и $Q \in E(\mathbb{F}^*_{r^k}) \cap \cap Ker(\pi - [r])$.

Обозначая новое спаривание

$$\varepsilon''(Q, P) = \prod_{l=0}^{\frac{k}{k_0}} f_{T,Q}(P^{\pi^{lk_0}}),$$

при $p \nmid L$ получим для него те же свойства, что и для ε', а именно выполнение утверждения Теоремы 2.

Пусть теперь требуется обратить рассматриваемое спаривание по первому аргументу. В этом случае исключение переменной y (второй координаты точки Q) из системы, состоящей из уравнения кривой и уравнения

$$\varepsilon''(Q, P) = z\tilde{z},\ z \in \tilde{G}_1, \tag{13}$$

ввиду малости параметров k, T приводит, как и выше, к решению многочлена от одной переменной относительно малой степени.

Пусть теперь требуется обратить рассматриваемое спаривание по второму аргументу. В этом случае в рассматриваемые уравнения входят не только неизвестные координаты точки P, лежащие в $\mathbb{F}_{r^k}$, но и величины, полученные из них применением автоморфизма Фробениуса. Поэтому предлагается некоторая новая процедура, сводящая данную задачу к решению системы линейных уравнений.

Исключая степени, старше первой, переменной y с помощью уравнения кривой, получим

$$f_{T,Q}(P(x,y)) = \frac{yf_{11} + f_{12}}{yf_{21} + f_{22}}.$$

Пусть

$$yf_{11} + f_{12} = \sum_{i=0,1,\ldots,S-1;\ j=0,1} a_{ij}x^i y^j;$$
$$yf_{21} + f_{22} = \sum_{i=0,1,\ldots,S-1;\ j=0,1} b_{ij}x^i y^j.$$

Тогда уравнение (13) можно записать в виде

$$\prod_{l=0}^{\frac{k}{k_0}} \sum_{i=0,1,\ldots,S-1;\ j=0,1} a_{ij}x_l^i y_l^j = z\prod_{l=0}^{\frac{k}{k_0}} \sum_{i=0,1,\ldots,S-1;\ j=0,1} b_{ij}x_l^i y_l^j, \quad (14)$$

где x_l, y_l — координаты точки $P^{\pi^{lk_0}}$. Раскрывая скобки, получим

$$\prod_{l=0}^{\frac{k}{k_0}} \sum_{i=0,1,\ldots,S-1;\ j=0,1} a_{ij}x_l^i y_l^j = \sum_{\psi} c_\psi \psi(x_1, y_1, \ldots, x_{\frac{k}{k_0}}, y_{\frac{k}{k_0}}), \quad (15)$$

где c_ψ — все различные произведения вида $\prod_{i,j} a_{ij}^{\alpha_{ij}}$, $\alpha_{ij} \in \mathbb{N} \cup \{0\}$, $\sum_{i,j}\alpha_{ij} = \frac{k}{k_0}$, а ψ — некоторые многочлены от $2\frac{k}{k_0}$ переменных. Число таких произведений (а значит, и многочленов ψ) не больше чем $C^{2S-1}_{\frac{k}{k_0}+2S-1} \leqslant \left(\frac{k}{k_0} + 2S\right)^{2S-1}$. Как уже неоднократно отмечалось (см., например, [10]), равенство $T \equiv r^j \pmod p$ приводит к тому, что $T^k \approx p$. Однако, если выбрать S, T небольшими константами, а $k \approx \log p$, то число многочленов ψ оценится небольшой величиной k^{2S}. Подставляя выражение (15) и аналогичное ему выражение для правой части в уравнение (14), получим линейное однородное уравнение от переменных ψ. Заменяя в равенстве (13) P на $[m]P$, а z на z^m для различных значений m, аналогично получим другие линейные однородные уравнения для тех же переменных ψ. Решая полученную систему за $O((\log p)^{6S})$ арифметических операций, получим все переменные ψ с точностью до мультипликативной константы C. Поскольку $f(\lambda) = \Pi_l(\lambda - x_l)$ при фиксированном λ также имеет вид (15), мы получим его с точностью до умножения на C. Выбирая $\frac{k}{k_0}$ различных значений для λ, по интерполяционной формуле Лагранжа получим многочлен $Cf(x) \in \mathbb{F}_{r^k}[x]$, среди корней которого найдем искомое нами решение.

Осталось оценить вероятность разрешимости уравнения (13) при случайном выборе $\tilde{z}$. В случае обращения по второму аргументу (т. е. при фиксированном Q) искомое значение $P \in G_1$ может быть "сдвинуто" с помощью любых значений остальных компонент разложения группы $E(\mathbb{F}_{r^k})$. Мощность такого "сдвига" оценивается величиной $\#E(\mathbb{F}_{r^k})/\#G_1 \approx \frac{r^k}{r}$. В то же время мощность возможных $\tilde{z}$ оценивается такой же величиной. При этом, если Q лежит в G_2 надо компоненты $\tilde{z}$, лежащие в примарных компонентах $\tilde{G}_i$, $i \geqslant 2$, соответствующих простым p_i из разложения НОД($\#E(\mathbb{F}_{r^k})[N]$, $r^k - 1$) (см. Теорему 2) положить равными единице. В случае, если мы прибавим к Q произвольную точку из $ker(\pi - [r])$ с нулевой компонентой, соответствующей G_2, этого обнуления можно не делать.

В случае обращения уравнения (13) по первому аргументу (т. е. при фиксированном P) искомое значение $Q \in G_2$ может быть "сдвинуто" с помощью любых значений остальных компонент разложения группы $E(\mathbb{F}_{r^k}) \cap Ker(\pi - [r])$. Мощность такого "сдвига" оценивается величиной $\#E(\mathbb{F}_{r^k})/(\#Ker(\pi - [1])\#G_2) \approx \frac{r^k}{rp}$. В то же время мощность возможных $\tilde{z}$ оценивается величиной $\frac{r^k}{p\Pi p_i}$, где произведение берется по $i \geqslant 2$, для которых p_i делит НОД($\#E(\mathbb{F}_{r^k})[N]$, $r^k - 1$), поскольку при $P \in G_1$ компоненты $\tilde{z}$, лежащие в примарных компонентах $\tilde{G}_i$, $i \geqslant 2$, соответствующих простым p_i из разложения НОД($\#E(\mathbb{F}_{r^k})[N]$, $r^k - 1$) (см. Теорему 2), равны единице и число таких различных $\tilde{z}$ оценивается величиной $\frac{r^k}{p\Pi p_i}$, где произведение берется по $i \geqslant 2$, для которых p_i делит НОД($\#E(\mathbb{F}_{r^k})[N]$, $r^k - 1$). Такие i для некоторых кривых существуют.

Таким образом, существуют кривые, для которых случайный выбор $\tilde{z}$ при обращении равенства (13) может оказаться эффективным, если значительная доля сдвигов образа может

быть получена с помощью сдвигов прообразов, относительно соответственно G_2 и G_1. Поскольку мощности этих сдвигов близки этому, может помешать только "слипание" сдвигов прообразов рассматриваемого спаривания в один сдвиг образа. Однако поскольку наше спаривание задается рациональной функцией малой степени, это не может привести к существенному снижению рассматриваемой вероятности разрешимости уравнения (13).

Список литературы

1. **Menezes A., Okamoto T., Vanstone S.** Redusing elliptic curve logarithms in a finite field // IEEE Trans. Inf. Theory. 1993. Vol. IT-39, N. 5. P. 1639—1646.

2. **Verheul E. R.** Evidence that XTR is more secure than supersingular elliptic curve cryptosystems // J. Cryptology. 2004. Vol. 17. P. 277—296. doi: 10.1007/s00145-004-0313-x

3. **Takakazu Satoh.** Miller is Easy for the Redused Tate Pairing on Supersingular Curves of Embedding Degree Two and Three. URL: https://eprint.iacr.org/2019/385.

4. **Akagi S., Nogami Y.** Exponentiation inversion problem redused from fixed argument pairing inversion on twistable Ate pairing and its difficulty // Advances in Information and ComputerSecurity. IWSEC 2014, Lect. Notes in Comput. Sci. Vol. 8639. P. 240—249.

5. **Barreto P. S. L. M., Naehrig M.** Pairing-friendly elliptic curves of prime order // SAC 2005. Lect. Notes in Comput. Sci. Vol. 3897. P. 319—331.

6. **Bresing F., Weng A.** Ellcurves suitable for pairing based cryptography // Des. Codes Crypt. 2005. Vol. 37. P. 133—141.

7. **Freeman D.** Constructing pairing-friendly ellcurves with embedding degree 10 // Algorithmic Number Theory, Proc. 7th Internat. Sympo, ANTS-VII 2006. Lect. Notes in Comput. Sci. Vol. 4076. P. 452—465.

8. **Dupont R., Enge A., Morain F.** Building curves with arbitrary small MOV-degree over finite prime fields // J. Cryptol., 2005. Vol. 18. P. 79—89. URL: http://eprint.iacr.org/2002/094.pdf

9. **Silverman J.** The Arithmetic of Elliptic Curves. Springer, 1986. 513 + xviiip.

10. **Черепнев М. А.** Обращение спариваний для решения задачи дискретного логарифмирования // Фундаментальная и прикладная математика. 2013. Т. 18, Вып. 4. С. 185—195 (Journal of Mathematical Science. 2015. 206, Iss. 6. P. 734—741).

11. **Galbraith S., Hess F., Vercauteren F.** Aspects of pairing invertion // IEEE Trans. Dec. 2008. Vol. 54, Iss. 12. P. 5719—5728.

12. **Davenport J. H.** On the integration of Algebraic Functions // LNCS. 1979. Vol. 102. Springer-Verlag, Berlin.

13. **Miller V. S.** Short Programs for functions on Curves. Unpublished manuscript. 1986. URL: http://crypto.stanford.edu/miller/

14. **Miller V. S.** The Weil Pairing, and Its Efficient Calculation // J. Cryptology. 2004. Vol. 17. P. 235—261.

15. **Cohen H., Frey G.** ets. Handbook of Elliptic and Hyperelliptic curve Cryptography. Chapman and Hall, London, New York, Singapore, 2006.

16. **Hess F.** Pairing Lattices // LNCS. 2008. Vol. 5209, p. 18—38, Springer-Verlag, Berlin-Heidelberg-New York, 2008.

17. **Hess F.** A Note on the Tate Pairing of Curves over Finite Fields // Arch. Math. 2004. Vol. 82. P. 28—32.

18. **Hess F., Smart N. P., Vercauteren F.** The Eta-pairing revisited // IEEE Transactions on Information Theory. Oct. 2006. Vol. 52. P. 4595—4602.

19. **Chang-An Zhao, Fangguo Zhang, Jiwu Huang.** A Note on the Ate Pairing Cryptology // ePrint Archive: Report 2007/247. URL: http://eprint.iacr.org/2007/247.pdf

20. **Василенко О. Н.** Теоретико-числовые алгоритмы в криптографии. М.: МЦНМО, 2006. 325 с.

21. **Barreto P. S. L. M., Castello C., Misoczki R., Naehrig M., Pereira G. C. C. F., Zanon G.** Subgroup security in pairing-based cryptography // Progress in Cryptology — ATINCRYPT 2015. LNCS. Vol. 9230, Springer-Verlag, 2015. P. 245—265. Cryptology ePrint Archive, Report 2015/247.

22. **Barreto P. S. L. M., Galbraith S., Oh'Eigeartaigh C., Scott M.** Efficient Pairing Computation on Supersingular Abelian Varieties // Designs, Codes and Cryptography. 2007. Vol. 42, N. 3. P. 239—271.

M. A. Cherepniov, Ph. D., Professor, e-mail: cherepniov@gmail.com,
"Kontzern "Avtomatika",
S. S. Gracheva, Cand. Sci. (Tech.), Associate Professor,
Department of Statistics and Data Analysis, e-mail: statgracheva@mail.ru,
National Research University Higher School of Economics

Solution of the Diffie-Hellman Problem on Some Elliptic Curves Satisfying GOST 34.10—2018

This article is devoted to cryptanalysis of the often used Diffie-Hellman scheme of public key distribution. Starting with the article Menezes-Okamoto-Vanstone, the idea of which was previously stated in the works of I. Semaev, considerable interest from the point of view of attacking crypto-protokols on elliptic curves, began to acquire the degree of expansion or MOV-degree. In English literature, this parameter (hereinafter k) is called "embedding degree". We mean the extension of the coefficient field of the elliptic curve, which contains all the points of the original Prime order p. The random value of this parameter approaches the value of p, which results in the length of the entry of the element of the corresponding extension not much less than plogp. In the standard GOST 34.10—2018, this parameter is proposed to take more than 31, which allows you to use this extension, since the length of the record of its elements is not more than κ log p. In this paper, we propose a polynomial algorithm for solving the recognition and ordinary Diffie-Hellman problem, effective for some such curves. This means that public key distribution schemes constructed using these curves are insecure. The proposed algorithm is based on the

choice of such a pairing, which is nontrivially defined at all points of order p and can be represented as a rational function of relatively small degree. A reduction of the Diffie-Hellman problem to such an address is obtained by Verheul. The proposed construction is based on the non-reduced Ate pairing. Proposed new mechanisms to extend the scope of the considered pairing with the Frobenius automorphism and the reduction of inversion for the second argument (lying in the extension field of coefficients of the curve) to the solution of a system of linear equations with the subsequent search of the roots of polynomials of small degree. Estimates for the probability of solvability of the equations obtained by taking a random representative of an adjacent class representing the pairing value are presented.

Keywords: *public key distribution scheme, elliptic curves, Diffie-Hellman problem, Tate pairing, Ate pairing, Frobenius automorphism.*

DOI: 10.17587/it.26.159-168

References

1. **Menezes A., Okamoto T., Vanstone S.** Redusing elliptic curve logarithms in a finite field, *IEEE Trans. Inf. Theory*, 1993, vol. IT-39, no. 5, pp. 1639—1646.

2. **Verheul E. R.** Evidence that XTR is more secure than supersingular elliptic curve cryptosystems, *J. Cryptology*, 2004, vol. 17, pp. 277—296, doi: 10.1007/s00145-004-0313-x.

3. **Takakazu Satoh.** Miller is Easy for the Redused Tate Pairing on Supersingular Curves of Embedding Degree Two and Three, available at: https://eprint.iacr.org/2019/385.

4. **Akagi S., Nogami Y.** Exponentiation inversion problem redused from fixed argument pairing inversion on twistable Ate pairing and its difficulty, *Advances in Information and ComputerSecurity. IWSEC 2014, Lect. Notes in Comput. Sci.,* vol. 8639, pp. 240—249.

5. **Barreto P. S. L. M., Naehrig M.** Pairing-friendly elliptic curves of prime order, *SAC 2005. Lect. Notes in Comput. Sci.,* vol. 3897, pp. 319—331

6. **Bresing F., Weng A.** Ellcurves suitable for pairing based cryptography, *Des. Codes Crypt.,* 2005, vol. 37, pp. 133—141.

7. **Freeman D.** Constructing pairing-friendly ellcurves with embedding degree 10 // Algorithmic Number Theory, *Proc. 7th Internat. Sympo, ANTS-VII 2006. Lect. Notes in Comput. Sci.*, vol. 4076, pp. 452—465.

8. **Dupont R., Enge A., Morain F.** Building curves with arbitrary small MOV-degree over finite prime fields, *J. Cryptol.*, 2005, vol. 18, pp. 79—89, available at: http://eprint.iacr.org/2002/094.pdf

9. **Silverman J.** The Arithmetic of Elliptic Curves. Springer, 1986. 513 + xviiip.

10. **Cherepniov M. A.** *Fundamentalnayi prikladnaya matematika,* 2013, vol. 18, iss. 4, pp. 185—195 (*Journal of Mathematical Science*, 2015, vol. 206, iss. 6, pp. 734—741).

11. **Galbraith S., Hess F., Vercauteren F.** Aspects of pairing invertion, *IEEE Trans.* Dec. 2008, vol. 54, iss. 12, pp. 5719—5728.

12. **Davenport J. H.** On the integration of Algebraic Functions, *LNCS,* 1979, vol. 102, Springer-Verlag, Berlin.

13. **Miller V. S.** Short Programs for functions on Curves. Unpublished manuscript, 1986, available at: http://crypto.stanford.edu/miller/

14. **Miller V. S.** The Weil Pairing, and Its Efficient Calculation, *J. Cryptology*, 2004, vol. 17, pp. 235—261.

15. **Cohen H., Frey G.** ets. Handbook of Elliptic and Hyperelliptic curve Cryptography, Chapman and Hall, London, New York, Singapore, 2006.

16. **Hess F.** Pairing Lattices, *LNCS,* 2008, vol. 5209, p. 18—38, Springer-Verlag, Berlin-Heidelberg-New York.

17. **Hess F.** A Note on the Tate Pairing of Curves over Finite Fields, *Arch. Math.*, 2004, vol. 82, pp. 28—32.

18. **Hess F., Smart N. P., Vercauteren F.** The Eta-pairing revisited, *IEEE Transactions on Information Theory*, Oct. 2006, vol. 52, pp. 4595—4602.

19. **Chang-An Zhao, Fangguo Zhang, Jiwu Huang.** A Note on the Ate Pairing Cryptology, *ePrint Archive: Report 2007/247*, available at: http://eprint.iacr.org/2007/247.pdf

20. **Vasilenko O. N.** Numerical Algorithms in Cryptography, Moscow, MCPME, 2006, 325 p. (in Russian).

21. **Barreto P. S. L. M., Castello C., Misoczki R., Naehrig M., Pereira G. C. C. F., Zanon G.** Subgroup security in pairing-based cryptography, *Progress in Cryptology — ATINCRYPT 2015, LNCS,* vol. 9230, Springer-Verlag, 2015, pp. 245—265, Cryptology ePrint Archive, Report 2015/247.

22. **Barreto P. S. L. M., Galbraith S., Oh'Eigeartaigh C., Scott M.** Efficient Pairing Computation on Supersingular Abelian Varieties, *Designs, Codes and Cryptography*, 2007, vol. 42, no. 3, pp. 239—271.

УДК 004.4’422 DOI: 10.17587/it.26.169-176

С. М. Салибекян, канд. техн. наук, e-mail: ssalibekyan@hse.ru,
Национальный исследовательский университет "Высшая школа экономики", Москва

Трансляция арифметико-логического выражения с использованием формата внутреннего представления на базе парадигмы dataflow

Статья посвящена разработке методики разбора, внутреннего представления и трансляции в машинный код инфиксных арифметико-логических выражений. Отличительной чертой разработки является применение нового формата внутреннего представления выражения, основанного на парадигме dataflow (вычисления с управлением потоком данных). Методика может найти применение в компиляторах и интерпретаторах языков программирования высокого уровня.

Ключевые слова: *разбор арифметико-логического выражения, инфиксная форма записи арифметического выражения, компиляция, внутреннее представление арифметического выражения, языки программирования высокого уровня, вычисления с управлением потоком данных*

Введение

Арифметико-логические выражения (АЛВ) являются частью синтаксиса всех языков высокого уровня (ЯВУ). Например, целью создания первого в истории ЯВУ Fortran как раз и была автоматизация разбора арифметических выражений: Fortran — это сокращение от англ. выражения "FORmula TRANslator" ("транслятор формул"). Наиболее удобной для человека является инфиксная форма записи АЛВ. И для перевода формул из инфиксной формы в машинный код довольно часто применяется промежуточное представление АЛВ [1], называемое внутренним представлением, из которого впоследствии генерируется машинный код.

Темой настоящей статьи является разработка нового формата промежуточного представления АЛВ, способов трансляции инфиксной записи АЛВ в это представление и генерация машинного кода. Необходимость разработки методики возникла в ходе создания компилятора объектно-атрибутного (ОА) языка (ОА язык) [2]. Язык создавался для управления параллельными вычислениями в ОА вычислительной системе (ВС), относящейся к классу dataflow (вычисления с управлением потоком данных) [3, 4], а также для описания сложно структурированных динамических информационных конструкций, обрабатываемых ею. Компилятор для ОА языка разрабатывался методом "раскрутки", когда сначала создается простейший компилятор, затем с его использованием пишется программа новой версии компилятора, расширяющего функциональность языка, и т. д. В настоящий момент уже реализована вторая версия ОА языка, где пока нет автоматического распознания АЛВ. И поэтому самой важной задачей при создании третьей версии языка стала ее реализация.

Таким образом, целью исследования является создание и формализация методики анализа АЛВ, представленного в инфиксной форме. Задачами настоящего исследования является разработка:

- формата внутреннего представления арифметических выражений, приспособленного для реализации на базе ОА графа (динамическая информационная конструкция для представления сложно структурированных данных в ОА ВС);
- методики синтеза внутреннего представления инфиксного АЛВ, содержащего бинарные арифметико-логические операции ("+", "–" и т. д.), а также функции с фиксированным и произвольным числом аргументов во внутреннее представление;
- методики выполнения АЛВ во внутреннем представлении или перевода внутреннего представления в машинный код.

Автором был проведен анализ существующих решений в данной области, который пред-

ставлен в следующей главе, однако ни одно из них не оказалось приемлемым решением поставленной задачи. Дело в том, что, во-первых, эти методики ориентированы на разбор только АЛВ с бинарными операторами и не приспособлены, например, для разбора функций с переменным числом операндов. Во-вторых, в ОА ВС применяется универсальный формат представления данных и программы в виде семантической сети специального формата (ОА граф), потому-то и понадобилась разработка новой методики анализа и внутреннего представления АЛВ на его основе. Однако разработанные в ходе исследования формат внутреннего представления АЛВ и метод распознания АЛВ универсальны и могут быть использованы в компиляторе любого другого ЯВУ.

1. Обзор существующих решений

Для выбора наиболее подходящего метода внутреннего представления был выполнен обзор существующих решений. Разбор АЛВ можно разделить на три составные части: метод распознания выражений (перевод из инфиксной формы во внутреннее представление), внутреннее представление и перевод из внутреннего представления в машинный язык. Начнем с обзора форматов внутреннего представления арифметических выражений.

Наиболее известными форматами внутреннего представления АЛВ являются: тетрады, обратная польская запись и арифметический граф (ациклический ориентированный граф) [1].

При представлении в виде четверок (тетрад) каждая АЛ операция представляется в виде общности, состоящей из четырех полей: код арифметической операции, первый и второй операнды (константа или ссылка на переменную), ссылка на переменную для результата. Поле "результат" может хранить либо указатель на переменную, либо идентификатор поля операндов той тетрады операции, куда этот результат передается в качестве операнда. С помощью такого метода можно описывать только унарные или бинарные операции. Также применяется и представление в виде триад: каждая триада имеет поля: операция, операнд 1, операнд 2. В качестве операнда может выступать константа, переменная или ссылка на другую триаду, результат выполнения операции которой является операндом.

Формат "обратная польская запись" получил свое название в честь польского ученого Лукасевича, предложившего его. В нем описание АЛВ представляет собой последовательность констант, переменных и операций над ними. Обозначение операции записывается после обозначения операндов. Например, инфиксное выражение $(3 + x)\cdot(y + 5)$ в обратной польской записи имеет вид: $3\ x + y\ 5 + *$. Недостатком такого способа является то, что возможно только представление операций с фиксированным числом операндов. Например, функцию минимума, где может быть произвольное число операндов, таким способом описать не представляется возможным.

Арифметический граф (или синтаксическое дерево) — еще один способ промежуточного представления, где АЛВ записывается в виде ориентированного графа-дерева: узлы графа ассоциируются с арифметическими операциями, а дуги идентифицируют данные (начало дуги обозначает результат операции, а конец направлен на узел, обозначающий операцию, для которой данные являются операндом). Пример программной реализации синтеза арифметического графа приведен в работе [5]. Этот формат описания имеет преимущество перед тетрадами и польской записью, так как может описывать операции с нефиксированным числом операндов. Арифметический граф можно реализовать, например, с помощью динамических структур: каждый узел представляет собой структуру (в нашем случае имеется в виду конструкция с ключевым словом struct в языке программирования Си), которая включает в себя поле кода АЛ операции и несколько ссылок на другие структуры, описывающие операцию, функцию или операнд. В качестве недостатка такого представления можно отметить сложность перевода из инфиксной записи АЛВ во внутреннее представление.

Одним из наиболее удачных алгоритмов анализа АЛВ является алгоритм Бауэра—Замельзона — его можно использовать как для перевода инфиксной формы АЛВ в обратную польскую форму, так и для непосредственного выполнения вычислений по АЛВ во время интерпретации программы. По этому алгоритму разбор АЛВ ведется с использованием двух стеков: в один из них помещаются операнды, в другой — операции. Приоритет и порядок обработки арифметических операций задается с помощью таблицы функций перехода, описывающей действие компилятора при переходе от одного символа операции к другому. Данный алгоритм был упрощен Э. Дейкстрой (метод стека с приоритетами) [6]. В нем каждой операции присваивается число, отражающее ее приоритет, и поэтому отпадает необходимость применения громоздкой таблицы обработки операций. Вышеприведенные алгоритмы при-

меняются и для перевода инфиксной записи АЛВ в обратную польскую, впрочем, их легко модифицировать для синтеза тетрад или арифметического дерева.

Еще одна интересная методика разбора АЛВ предложена в работе [7]. В ней используется вектор типов операции. Каждый тип подразумевает обозначение операции, ее арность (сколько аргументов имеется у операции) и приоритет. Синтаксис АЛВ задается с помощью контекстно-свободной грамматики, а синтаксический анализ АЛВ осуществляется с помощью LR(1) разбора [8]. После разбора АЛВ представляется в виде арифметического графа, а граф может быть использован, например, для вычисления значения, записанного с помощью АЛВ. Данный способ подходит для разбора математических операций с фиксированным числом аргументов.

Однако все алгоритмы, попавшие в обзор данной статьи, в состоянии анализировать только унарные (с одним аргументом) и бинарные (с числом аргументов, равным двум) операции, а операции с фиксированным числом операндов в состоянии анализировать только алгоритм, предложенный в работе [7]. Таким образом, все вышеперечисленные методики внутреннего представления, синтеза и перевода на машинный язык АЛВ оказались неудобными при реализации на dataflow вычислительной системе. Поэтому и потребовалась разработка собственного формата внутреннего представления арифметических выражений, а также методов синтеза АЛВ и их исполнения или трансляции на машинный язык.

2. Объектно-атрибутный граф как способ представления арифметических выражений

В настоящей работе внутреннее представление арифметических выражений строится на базе ОА подхода к организации вычислений и структур данных. ОА ВС относится к классу ВС, управляемых потоком данных, где акцент делается на описание обмена данными [9], а не последовательности операций, как в классической парадигме с управлением вычислениями посредством потока команд. Еще одной особенностью ОА подхода является работа с динамическими данными, упакованными в информационную конструкцию под названием ОА граф (или ОА сеть, ОА дерево), включающего в себя как данные, так и программный код [3]. Для описания предложенного подхода к обработке АЛВ введем два понятия: АЛ ОА граф, который будет использоваться как реализация арифметического графа, и арифметико-логическое устройство (АЛУ). АЛУ вычисляет значение АЛВ, представленного в виде ОА графа (ОА граф также может найти применение и для перевода АЛВ в машинный код).

"Кирпичиком", из которых строится ОА граф, является милликоманда (МК). Она состоит из двух полей: нагрузка (данные или указатель) и атрибут, идентифицирующий нагрузку. Формально МК — это кортеж $c = \langle a, l \rangle$, где l — множество нагрузок (константы или указатель на ячейку памяти), $a \in A$ — множество атрибутов (A — счетное множество). В частности, атрибут может хранить код операции, которую необходимо проводить над операндом, хранящимся в нагрузке МК.

В состав АЛУ входит регистр под названием "аккумулятор". Один из операндов АЛ операции обязательно должен находиться в аккумуляторе, и результат выполнения операции помещается в аккумулятор — такое решение аналогично аккумуляторной архитектуре процессора [10]. Например, операция сложения описывается с помощью трех МК: первая — запись значения из нагрузки МК в аккумулятор, вторая — сложение нагрузки со значением в аккумуляторе (результат помещается в аккумулятор) и третья — запись результата из аккумулятора в ячейку памяти. Любое АЛВ можно представить как линейную последовательность МК, исполняемых АЛУ.

Для описания последовательности МК будем применять следующую нотацию: нагрузка отделяется от атрибута с помощью знака "#", а несколько последовательно выполняемых МК разделяются знаком пробела. Атрибут МК обозначается с помощью мнемоники, например, Set — установить значение в аккумулятор АЛУ, Add — сложить, Sub — вычесть, Out — выдать результат из аккумулятора. Так, операция сложения в предложенной нотации выглядит следующим образом: Set#2 Add#2 (по первой МК АЛУ запишет 2 в аккумулятор, по второй — прибавит 2 к значению из аккумулятора и полученный результат поместит в аккумулятор). Для выдачи результата сложения можно использовать МК выдачи с мнемоникой "Out", в нагрузке которой находится указатель на ячейку памяти (переменную), куда необходимо поместить результат следующей операции:

$$\text{Set\#x Add\#y Out\#Var}, \qquad (1)$$

где x, y — имена переменных-операндов; Var — имя переменной для записи результата.

Последовательность МК может быть сколь угодно длинной, если приоритет операций будет одинаков. Последовательность МК будем называть информационной капсулой (ИК).

Однако в инфиксной форме АЛВ первым идет указание переменной для записи результата ("y = ..."). И тот факт, что МК записи результата (Out) находится в конце ИК, вызывает небольшие сложности при разборе выражения: необходимо запоминать адрес, а затем записывать его в конец ИК. Поэтому для удобства разбора АЛВ введем в состав АЛУ регистр, хранящий адрес для записи результата вычислений, и расширим функциональность АЛУ атрибутом МК "OutSet", по которой АЛУ выполнит запись адреса из нагрузки МК в этот регистр. В этом случае запись результата АЛУ будет проводиться после того, как оно выполнит все МК из ИК с описанием АЛВ. После того как АЛУ выполнит все МК из ИК, значение из аккумулятора (т. е. результат вычислений) будет записан по адресу, установленному в регистре адреса результата Var:

OutSet=Var Set#x Add#y. (2)

Для реализации записи результата сразу в несколько ячеек памяти, например, как в выражении Var1 = Var2 = x + y, необходимо заменить регистр записи адреса на буфер (вектор) адресов. Тогда ИК с АЛВ примет вид (3), в которой встречаются две МК OutSet. После выполнения последней МК из ИК результат вычислений будет записан по адресам Var1 и Var2:

OutSet#Var1 OutSet#Var2 Set#x Add#y. (3)

Следующей задачей, требующей решения, является представление арифметических выражений с различным приоритетом операций (приоритет задается как приоритетом самой АЛ операции, так и с помощью скобок при записи АЛВ). Наиболее удачным оказалось следующее решение: расположить описание операций более высокого приоритета в отдельной ИК, а ссылку на нее расположить в нагрузке МК, которая использует результат вычисления этого выражения в качестве операнда. Таким образом, описание АЛВ представляет собой ОА граф топологии "дерево". На рис. 1 представлен ОА граф, описывающий выражение: Y = D − − 10 + max(A + B, C − 5, 0). На рис. 1 прямоугольником показан атрибут МК, а овалом — его нагрузка. Номер уровня, где расположена ИК, соответствует уровню приоритета операций. Например, на рис. 1 на первом уровне находятся операции OutSet, Set, "−" и "+". У всех операций, кроме последней, операндами являются константы или переменные, а нагрузка последней МК хранит ссылку на описание АЛВ более высокого приоритета, а именно функцию Max. В нагрузке МК с атрибутом "Max", в свою очередь, находится указатель на ИК с описанием трех операндов для этой функции: первые два операнда — выражения, третий — константа. Первая МК устанавливает в аккумулятор значение, а последующие (с атрибутом "MaxCalc") записывают в аккумулятор максимальное значение аккумулятора и нагрузки.

Для каждого уровня АЛВ ОА графа ИК используется свой отдельный аккумулятор (т. е. применяется стек аккумуляторов): при переходе на более высокий уровень иерархии задействуется следующий аккумулятор из стека; при возврате на предыдущий уровень активным становится предыдущий аккумулятор из стека, а значение из аккумулятора предыдущего уровня используется в качестве операнда для операции на нижележащем уровне. Таким образом, АЛУ может вычислить значение АЛВ путем последовательного выполнения МК из ИК, входящей в состав ОА графа (наподобие левостороннего обхода графа в глубину [11]).

Для формализации выполнения АЛВ необходимо ввести индексацию МК ОА графа (каждая МК имеет свой уникальный индекс). МК дополняется полем *inext*, в которой хранится индекс следующей МК из ИК. Специальный индекс с обозначением *nil* используется в качестве индекса в поле *inext* в конце ИК. С помощью поля *inext* МК можно выстроить в цепочку, заканчивающуюся индексом *nil*, которая формирует ИК. В состав АЛУ также вводится регистр индекса текущей (выполняемой) МК. АЛУ обрабатывает МК, индекс которой находится в этом регистре, далее же в регистр записывается индекс из поля *inext*, и цикл выполнения МК повторяется.

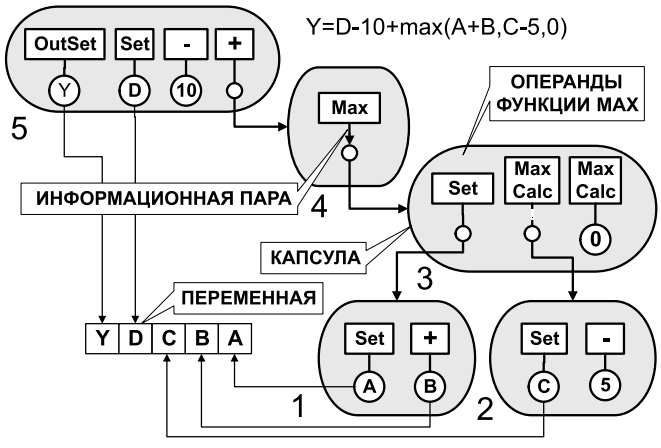


Рис. 1. Внутреннее представление арифметического выражения с многоаргументной функцией

В том случае, когда в регистр записывается *nil*, АЛУ может осуществить две реакции. Если ИК имеет наинизший уровень, то АЛУ записывает результат по адресам, хранящимся в буфере адресов для записи результата, иначе передает значение из аккумулятора текущего уровня в качестве операнда на уровень ниже.

Осуществим формализацию предложенной модели описания и выполнения АЛВ посредством математического аппарата, предложенного в работе [12]. Итак, АЛ ОА граф можно представить в виде тройки:

$$OAG = \{MKA,\ L,\ \bar{C}\},$$

где *MKA* — множество индексов операций, выполняемых АЛУ;

$$L = \{Const \cup I \cup \Omega\},$$

где *Const* — это множество любых констант (например, $Const = \{\mathbb{R} \cup \Sigma\}$, где $\mathbb{R}$ — множество вещественных чисел, Σ — множество цепочек символов ($\Sigma = A^*$, где A — алфавит символов); Ω — множество индексов (адресов) ячеек памяти (*RAM*), где хранятся данные; I — множество адресов МК. *RAM* (память переменных) формализуется как вектор, элементами которого являются константы, т. е. $RAM_k \in Const$ ($|RAM| = |\Omega|$, $\Omega = 1, 2, \ldots |\overline{RAM}|$). Пусть $\bar{C}$ — память МК:

$\bar{C} = <c>^k$ — это вектор кортежей, где каждое *c* представляет собой тройку

$$c = <mka,\ l,\ inext>, \qquad (3)$$

где $mka \in MKA$ — атрибут МК (атрибут идентифицирует операцию, которую необходимо применить к операнду, например, Set, OutSet, Add, Sub и т. д.); *inext* — индекс следующей МК из ИК ($inext \in I$, если МК последняя в ИК, то для нее $inext = nil$); l — нагрузка МК ($l \in L$).

Если нагрузкой МК является индекс МК (т. е. $Load(mk) \in I$, где функция *Load* выделяет из МК нагрузку), то это значит, что операнд для текущей операции получается путем вычисления АЛВ более высокого уровня иерархии (в нагрузке МК хранится индекс первой МК из ИК более высокого приоритета).

Начальное состояние АЛУ можно формализовать в виде четверки:

$$ALU = \{a_0,\ OAG,\ istart,\ \bar{F}\}, \qquad (4)$$

где a_0 — начальное значение аккумулятора ($a \in Const$); $istart \in I$ — индекс МК, с которого начинается обход (выполнение) АЛ ОА графа; $\bar{F}$ — вектор функций, описывающих вычислительные операции; операции F_i: $Const \boxtimes Const \to Const$. Множество индексов элементов вектора $\bar{F}$ совпадает с множеством элементов *MKA*, т. е. $|\bar{F}| = |MKA|$. Иными словами, каждому атрибуту МК соответствует своя функция вычисления результата АЛ операции.

Состояние АЛУ во время вычислительного процесса можно описать с помощью пятерки элементов:

$$S = \{\bar{A},\ Out,\ \overline{ind},\ Level,\ ALU\}, \qquad (5)$$

где *Level* — текущий уровень обрабатываемой ИК из АЛ ОА графа (в самом начале работы АЛУ $Level := 0$);

$\bar{A} = <Const>^{Level}$ — вектор значений аккумулятора (для каждого уровня приоритета АЛВ имеется свой аккумулятор);

$Out = \{\Omega\}^{Level}$ — множество адресов для записи результата вычисления;

$\overline{ind}$, ($ind_i \in I$) — вектор индексов текущей МК (МК, которая распознается); число элементов в векторе эквивалентно приоритету (уровню) разбираемой ИК АЛ ОА графа.

Тогда изменение контекста АЛУ за один такт можно формализовать как:

$$A^{k+1}_{Level} := F_{mka}(A^k_{Level},\ mkl),$$

где A^k_{Level}, A^{k+1}_{Level} — состояние аккумулятора на уровне *Level* на предыдущем и последующем тактах работы АЛУ, где $mka = atr(ind_{Level})$, $mkl = load(ind_{Level})$ — атрибут и нагрузка текущей МК (адрес текущей МК находится в верхушке стека $\overline{ind}$). По окончании выполнения ИК, если $Label \neq 0$ (т. е. распознавание проводится не на первом уровне АЛВ), выполняется запись результата в аккумулятор более низкого уровня $A^{k+1}_{Level-1} := F_{mka}(A^k_{Level},\ mkl)$, $Level := Level - 1$, иначе происходит запись результата вычислений в RAM по индексам из множества *Out* ($RAM_i = A_1$, где $i \in Out$, A_1 — аккумулятор первого уровня), и АЛУ останавливает свою работу.

Следующая МК для выполнения выбирается таким образом. Если $next(mk) \neq nil$, то $ind_{Level} := next(mk)$, где функция *next* выдает поле *inext* из МК. В том же случае, когда $Load(ind_{Level}) \in I$, $Level := Level + 1$. Когда $next(mk) = nil$, то в том случае, если $Level = 1$ (наинизший приоритет ИК), происходит запись значения аккумулятора в ячейки RAM, адреса которых входят в множество *Out*, и сброс множества *Out* (т.е. $Out = \varnothing$); если же $Level \neq 0$, то $A_{Level-1} := F_{mka}(A_{Level-1},\ A_{Level})$, т. е. полученный результат передается на уровень ниже в качестве операнда.

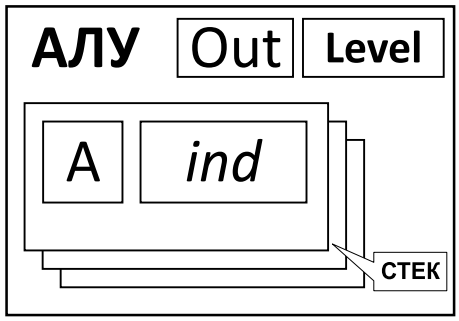


Рис. 2 Схема виртуального устройства АЛУ

Модель функционирования АЛУ можно схематично представить следующим образом. Имеются буфер адресов для записи результата *Out*, ячейка памяти для хранения текущего уровня приоритета разбираемой ИК *Level*, а также стек, который на каждом уровне хранит два элемента: аккумулятор (*A*) и индекс (адрес) анализируемой в настоящее время МК (*ind*). При переходе к анализу ИК более высокого уровня (например, уровень *i*) в стек добавляется новый элемент (т. е. A_i и ind_i), а при окончании разбора ИК элемент из верхушки стека удаляется. Схема АЛУ приведена на рис. 2.

3. Алгоритм преобразования АЛВ, представленного в инфиксной форме, во внутреннее представление и перевод внутреннего представления в машинный код

В ходе исследования выяснилось, что все приведенные в обзоре алгоритмы распознания АЛВ и синтеза внутреннего представления неудобны для реализации на базе dataflow вычислительной системы. Поэтому потребовалась разработка собственного алгоритма, удовлетворяющего заявленным требованиям. На вход алгоритма с лексического анализатора поступает поток описаний лексем, например, представленных в виде токенов [13]. На выходе — АЛ ОА граф, с описанием АЛВ.

Разработанный алгоритм распознания АЛВ достаточно прост, так как в нем применяется только один стек, который используется для хранения адресов ИК, входящих в состав синтезируемого АЛ ОА графа, и приоритета операции каждого уровня графа. Разбор арифметического выражения выполняется слева направо. Пусть S — упорядоченное множество адресов памяти МК ($s \in S$, $s \in I$), а упорядоченное множество *Priority* хранит приоритет последней анализируемой операции АЛВ на данном уровне приоритета АЛВ. Первоначально в S присутствует только один элемент — адрес пустой ИК (в нее будут помещаться МК с описанием операций первого (низшего) уровня приоритета), а уровень *Level* будет равен 1 (индексация элементов S и *Priority* начинается с 1). Пусть функция p возвращает приоритет операции, заданный атрибутом МК. Пусть t — текущая распознаваемая лексема из АЛВ $t \in \{C \cup V \cup O \cup F \cup "(" \cup ")" \cup ","\}$, где C — константы, V — переменные, O — АЛ операции, F — упорядоченное множество применяемых в анализируемой программе функций со знаком "(" в конце, например, "sin(". Пусть *mk* — это текущая синтезируемая МК из ИК, адрес которой хранится в S_{Level}. Пусть первоначально в S хранится один элемент — ссылка на *mk*, причем *atr*(*mk*) = *OutSet*, а в нагрузке находится указатель на переменную, в которую будет записываться результат вычислений (т. е. уже проведен разбор операции присвоения результата: например, "x = "), *Level* = 1, $Priority_1 = 0$ (операция присвоения имеет наинизший приоритет). Все функции из множества F проиндексированы. Далее приведем упрощенный алгоритм синтеза АЛ ОА графа, который работает при условии, что АЛВ написано синтаксически правильно, а в качестве аргументов функции выступают только переменные или константы (но не выражения) и функции, имеющие не менее двух аргументов.

1. Если $t \in \{C \cup V\}$ и $S_{Level} \neq nil$, то *load*(*mk*): = *t*.
2. Если $t \in \{C \cup V\}$ и $S_{Level} = nil$, создать новую МК *mk* = <Set, *t*> и добавить ее в ИК, по адресу из S_{Level}.
3. Если *t* = "(", то создать ИК *IC* (*IC* = *nil*), *load*(*mk*): = *adr*(*IC*), *Level*: = *Level* + 1, S_{Level}: = *adr*(*IC*).
4. Если *t* = ")", то *Level*: = *Level* − 1.
5. Если $t \in O$ и ($Priority_{Level} = p(t)$ или *atr*(*mk*) = = *Set*), то создать новую МК *mk* = <*t*, *nil*>, $Priority_{Level} = p(t)$.
6. Если $t \in O$ и $Priority_{Level} > p(t)$, то *Level*: = = *Level*-1, перейти к началу алгоритма (т. е. заново проанализировать лексему *t*).
7. Если $t \in O$ и $Priority_{Level} < p(t)$, то сформировать новую ИК *IC* = *nil*, сформировать МК *mk* = <*t*, *adr*(*IC*)>, *Level*: = *Level* + 1, S_{Level}: = *adr*(*IC*).
8. Если $t \in F$, то создать новую ИК *IC*, создать новую МК *mk* = < *t*, *adr*(*IC*)>, *Level*: = *Level* + 1, S_{Level}: = *adr*(*IC*), $Priority_{Level}$: = −*t*, добавить в *IC* МК <*Set*, *nil*>.
9. Если *t* = ", ", то считать следующую лексему *t*; создать МК *mk* = < −$Priority_{Level}$, *t*>

где *adr* — функция, возвращающая адрес ИК.

В пункте 8 алгоритма выражение $Priority_{Level}$:= := −*t* происходит сохранение индекса функции (знак минус необходимо для того, чтобы отличать приоритет операции от индекса функции). Данный прием необходим для упрощения алгоритма, чтобы не вводить новый элемент. В пунк-

те 9 $mk = < -Priority_{Level}, t>$ выполняется присвоение индекса функции атрибуту МК. Процесс синтеза АЛ ОА графа проиллюстрирован на рис. 3.

АЛ ОА граф можно использовать и для промежуточного представления АЛВ с последующим преобразованием в машинный код. Для осуществления генерации в машинный код можно использовать АЛУ-транслятор: его функцией является не вычисление АЛВ, а генерация машинного кода по нему. Реализация АЛУ-транслятора входит в планы будущих исследований. Поскольку в машинном коде большинство команд бинарные, то основной задачей АЛУ-транслятора является формирование из ОА графа последовательности бинарных (содержащих два операнда) машинных команд. Процедура трансляции представляет собой рекурсивный алгоритм обхода АЛ ОА графа в глубину. Например, МК Set при трансляции заменяется установкой первого операнда машинной инструкции (рис. 4). Следующая за ней МК с АЛ операцией транслируется в код операции машинной инструкции, а нагрузка транслируется во второй операнд. Если в ИК далее следует МК с операцией, то формируется следующая машинная команда, у которой первый операнд является результатом выполнения предыдущей команды. Если при обходе ОА графа встречается МК со ссылкой на ИК с описанием выражения более высокой иерархии, тогда промежуточный результат запоминается в ячейке памяти, а затем, после того, как алгоритм закончит обход ИК, в качестве операнда машинной инструкции добавляется ссылка на данную переменную. И в качестве адреса для записи результата в последней машинной инструкции указывается адрес ячейки для результата вычисления выражения (на рис. 4 это — переменная *Y*).

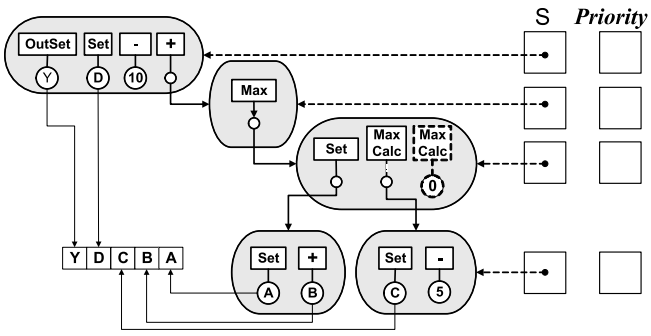


Рис. 3. Разбор АВЛ

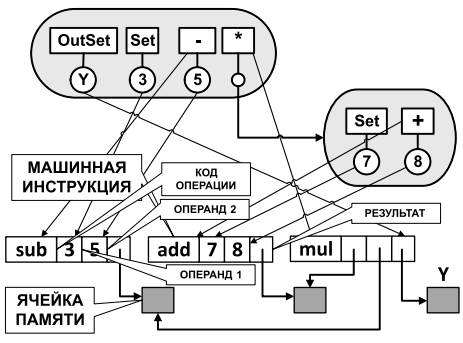


Рис. 4. Формирование машинного кода из АЛ ОА графа

Заключение

АЛУ может быть реализовано в виде виртуального устройства, входящего в состав выполняемой программы. Например, АЛУ было программно реализовано автором и в настоящий момент входит в состав среды имитационного моделирования вычислительного процесса в ОА ВС. Также функциональность компилятора ОА языка была дополнена распознанием АЛВ, представленных в инфиксной форме, и синтезом АЛ ОА графа. На рис. 5 представлена визуализация синтезированного в среде ОА имитационного моделирования АЛ ОА графа (пометка "Var" обозначает, что значение хранится в переменной). В среде программирования и моделирования также было интегрировано разработанное виртуальное устройство АЛУ, функциональность которого заключается в обработке АЛ ОА графа. Таким образом, ОА граф в ОА системе является не промежуточным, а конечным представлением АЛВ.

Разработанный метод представления АЛВ в виде арифметического ОА графа обладает по сравнению со способами, приведенными в обзоре, некоторыми преимуществами. Во-первых, у операндов АЛВ может быть сколь угодно большое (в том числе и переменное) число операндов. Во-вторых, можно задать несколько

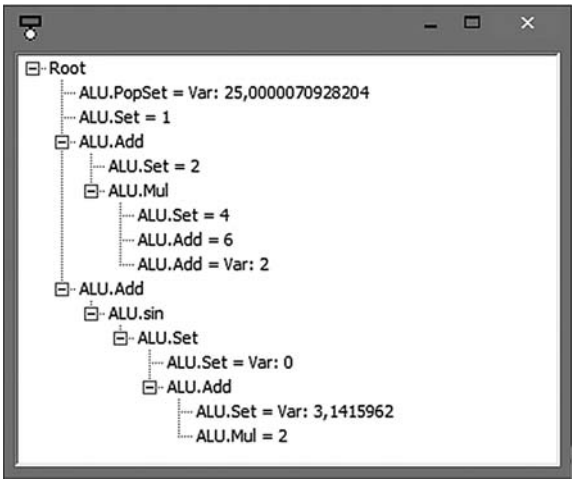


Рис. 5. Визуализация ОА графа арифметического выражения x = 1 + 2*(4 + 6 + x1) + sin(x2 + x3*2)

адресов ячеек памяти, куда будет записываться результат вычисления АЛВ. В-третьих, ОА граф удобен для трансляции в него инфиксной формы АЛВ: трансляция проводится непосредственно в промежуточную форму без использования дополнительных информационных конструкций (применяются только стек ссылок на адреса ИК для каждого яруса ОА графа и приоритет последней АЛ операции).

Следует отметить, что данный способ, в первую очередь, разрабатывался для применения в составе ОА вычислительной системы. В ней АЛ ОА граф используется не в качестве промежуточного, а в качестве конечного представления АЛВ. Однако методика может успешно применяться и для классической архитектуры ВС. Поэтому разработанная методика анализа АЛВ может найти применение в компиляторах ЯВУ на фазе семантического анализа АЛВ.

Список литературы

1. **Ахо А. В., Лам М. С., Сети Р., Ульман Д. Д.** Компиляторы: принципы, технологии и инструментарий. 2-е изд.: Пер. с англ. М.: ООО "И. Д. Вильямс", 2019. 1184 с.

2. **Салибекян С. М., Панфилов П. Б.** Объектно-атрибутная архитектура — новый подход к созданию объектных систем // Информационные технологии. 2012. № 2. С. 8—14.

3. **Data** flow computing: theory and practice / edited by John A. Sharp. Ablex Publishing Corp. Norwood, NJ, USA,1992. 569 с.

4. **Milutinovic V., Trifunovic N., Salom J., Giorgi R.** The guide to dataflow supercomputing. USA: Springer, 2015.

5. **Губаев Т. О., Петрова Н. К.** Разработка синтаксического анализатора арифметических выражений на языке С++ // Вестник Казанского государственного энергетического университета. 2018. Т. 10. № 2 (38). С. 32—40.

6. **Вторников А.** Арифметические выражения: анатомия, разбор, программирование // Системный администратор. 2013. № 10 (131). С. 68—73.

7. **Charles N. Fischer.** On Parsing and Compiling Arithmetic Expressions on Vector Computers ACM // Transactions on Programming Languages and Systems (TOPLAS). 1980. Vol. 2, N. 2. P. 203—224.

8. **Grune D. et al.** Modern Compiler Design. Second Edition. Springer, 2012. 822 p.

9. **Milutinovic V. et al.** The guide to dataflow supercomputing. USA: Springer, 2015.

10. **Корнеев В. В., Киселев А. В.** Современные микропроцессоры. СПб.: БХВ-Петербург, 2003. 448 с.

11. **Скиена С.** Алгоритмы. Руководство по разработке. 2-е изд.: Пер. с англ. СПб.: БХВ-Петербург. 2017. 720 с.

12. **Салибекян С. М., Панфилов П. Б.** Вопросы автоматно-сетевого моделирования вычислительных систем с управлением потоком данных // Информационные технологии и вычислительные системы. 2015. № 1. С. 3—9.

13. **Dick Grune et al.** Modern Compiler Design, Second Edition. Springer, 2012. 822 p.

S. M. Salibekyan, Assistant Professor, e-mail: salibek@yandex.ru,
National Research University Higher School of Economics, Moscow, Russion Federation

Translation of Arithmetic-Logical Expression Using Internal Representation Format Based on Dataflow Paradigm

The paper is devoted to the development of methods of parsing, internal representation and translation of infix arithmetic-logical expressions into machine code. A distinctive feature of the development is the use of a new format for the internal representation of the expression, based on the paradigm dataflow (calculations with data flow control). The technique can be used in high-level language compilers.

Keywords: *analysis of arithmetic-logical expression, infix form of arithmetic expression recording, compilation, internal representation of arithmetic expression, high-level languages, dataflow*

DOI: 10.17587/it.26.169-176

References

1. **Aho Al'fred V., Lam Monika S., Seti Ravi, Ul'man Dzhefri D.** Compilers: principles, technologies and tools, Moscow, I. D. Vil'yams, 2019, 1184 p.

2. **Salibekyan S. M., Panfilov P. B.** *Informacionnye Tekhnologii,* 2012, no. 2, pp. 8—14.

3. **John A. Sharp** ed. Data flow computing: theory and practice, Ablex Publishing Corp. Norwood, NJ, USA, 1992, 569 c.

4. **Milutinovic V., Trifunovic N., Salom J., Giorgi R.** The guide to dataflow supercomputing, USA, Springer; 2015.

5. **Gubaev T. O., Petrova N. K.** *Vestnik Kazanskogo Gosudarstvennogo Energeticheskogo Universiteta,* 2018, vol. 10, no. 2 (38), pp. 32—40.

6. **Vtornikov A.** *Sistemnyj Administrator*, 2013, no. 10 (131), pp. 68—73.

7. **Charles N. Fischer.** On Parsing and Compiling Arithmetic Expressions on Vector Computers, *ACM Transactions on Programming Languages and Systems (TOPLAS),* 2 (2), 1980, p. 203—224.

8. **Grune D. et al.** Modern Compiler Design. Second Edition. Springer, 2012, 822 p.

9. **Milutinovic V. et al.** The guide to dataflow supercomputing, USA, Springer, 2015.

10. **Korneev V. V., Kiselev A. V**. Modern microprocessors, SPb., BHV-Peterburg, 2003, 448 p.

11. **Skiena S.** Algorithms. Development guide, SPb., BHV-Peterburg, 2017, 720 p.

12. **Salibekyan S. M., Panfilov P. B.** *Informacionnye Tekhnologii i Vychislitel'nye Sistemy*, 2015, no. 1, pp. 3—9.

13. **Dick Grune et al.** Modern Compiler Design, Springer, 2012, 822 p.

УДК 004.773.5 DOI: 10.17587/it.26.177-184

И. В. Лобов, канд. физ.-мат. наук, ст. науч. сотр., e-mail: lobov@ihep.ru,
В. Г. Готман, мл. науч. сотр., e-mail: vladislav.gotman@ihep.ru,
НИЦ "Курчатовский институт" ФГБУ ГНЦ РФ — Институт физики высоких энергий

Адаптивная бесшовная потоковая трансляция в реальном времени над протоколом HTTP методом опережающей загрузки

Предлагается технология организации адаптивной бесшовной подстройки качества изображения при изменении эффективной пропускной способности канала клиента для потоковой трансляции в реальном времени методом опережающей загрузки. Предложен способ измерения эффективной пропускной способности канала клиента не на самом клиенте, а на сервере в процессе проведения трансляции. Обсуждаются различные технические вопросы, связанные с реализацией адаптивности, и определены условия применимости адаптивной подстройки качества изображения для рассматриваемого подхода. Описана работающая диспетчерская система, реализующая технологию адаптивной бесшовной трансляции в реальном времени методом опережающей загрузки.

Ключевые слова: *адаптивная бесшовная потоковая трансляция, опережающая загрузка, Ogg, Apple HLS, Adobe HDS, Microsoft Smooth Streaming, MPEG DASH*

Введение

В настоящее время организация живой потоковой трансляции медиаинформации через всемирную сеть осуществляется с помощью различных технологий. Наиболее известные и используемые из них — Apple HLS [1], Adobe HDS [2], Microsoft Smooth Streaming [3] и MPEG DASH [4], которые в дальнейшем для краткости будем называть "манифестными" (manifest-технологиями). В этих технологиях клиент является активным участником процесса, управляющим процессом передачи фрагментов от сервера. Клиент время от времени получает от сервера "свежий" файл-манифест, который постоянно обновляется и отражает последнюю информацию о наличии на сервере доступных фрагментов потока для загрузки. Основываясь на метаинформации в файле-манифесте, клиент выполняет загрузку подходящих фрагментов потока для обеспечения непрерывного воспроизведения.

В работах [5, 6] задача организации живой потоковой трансляции решается с помощью технологии, базирующейся на методе опережающей загрузки (progressive download) потока. На первый взгляд эта технология существенно отличается от "манифестных", поскольку клиент является пассивным участником процесса. Клиент по своей инициативе делает только один единственный запрос на старт загрузки потока при подключении к серверу. После этого процессом передачи фрагментов от сервера управляет сервер.

Несмотря на указанное отличие, в обоих случаях проблемы передачи потока через реальную физическую среду у этих двух подходов одни и те же. Независимо от того, кто именно (клиент или сервер) управляет потоком, потоковую трансляцию реального времени можно рассматривать как процесс передачи от сервера к клиенту последовательности небольших фрагментов, снабженных меткой времени. Этот процесс зависит от текущего качества канала, которое может меняться со временем весьма непредсказуемо. При ухудшении качества канала скорость передачи будет падать, фрагменты начнут накапливаться в канале передачи данных и уже не смогут успевать поступать клиенту вовремя. Если не предпринять никаких действий по управлению потоком, то в итоге приемный буфер клиента может опорожниться, и воспроизведение живой презентации прервется.

Рассмотрим описанную выше ситуацию немного подробнее. С точки зрения процесса передачи потока от сервера к клиенту важнейшим параметром потока является его битрейт,

который далее для удобства будет измеряться в байтах потока, передаваемых в единицу времени. Битрейт потока является свойством потока и потому не зависит от качества канала. Битрейт потока меняется со временем, так как зависит от параметров кодирования потока — степени сжатия изображения, размера кадра, скорости изменения изображения, числа кадров в секунду, соотношения между ключевыми и разностными кадрами. Но как бы ни менялся со временем битрейт передаваемого через канал потока, он всегда должен быть меньше пропускной способности этого канала, т. е. меньше битрейта гипотетического потока, который бы заполнил собой весь канал передачи данных. В противном случае (что может случиться при ухудшении качества сети) канал не будет успевать пропускать через себя поток: непереданные вовремя данные будут накапливаться в буфере сервера, а в приемном буфере клиента объем данных начнет уменьшаться. Воспроизведение потока клиентом начнет опережать поступление потока и рано или поздно остановится.

Решение этой проблемы состоит в своевременном адаптивном изменении битрейта потока таким образом, чтобы фрагменты потока поступали клиенту "без опозданий" и, тем самым, воспроизведение презентации не останавливалось. Если качество канала упало, то надо уменьшить и битрейт передаваемого клиенту потока. И наоборот, если качество канала восстановилось, то можно будет вернуть потоку его прежний битрейт.

Технологии Apple HLS, Adobe HDS, Microsoft Smooth Streaming и MPEG DASH являются адаптивными. Сервер предоставляет клиенту на выбор несколько вариантов одного и того же фрагмента, но с разными битрейтами (т. е. качествами). Клиент, ориентируясь по ходу получения потока, загружает фрагменты того или иного битрейта, причем переключение между фрагментами делается по границам кадров (бесшовно — seamlessly), что обеспечивает плавность воспроизведения. Более детальное описание адаптивности для этих технологий приведено в п. 1.

Как было отмечено выше, в технологии опережающей загрузки потока [5, 6] клиент пассивно принимает передаваемый ему поток презентации и не может менять битрейт приходящего потока. Поэтому при падении пропускной способности канала клиента воспроизведение презентации на клиенте попросту остановится. Все, что клиент может сделать по своей инициативе — это прекратить воспроизведение текущей трансляции и начать новую, подключившись к потоку другого битрейта. Такую подстройку трудно назвать "адаптивной", и тем более она не будет бесшовной, поскольку произойдет неизбежная остановка воспроизведения презентации клиентом на время загрузки фрагментов, необходимых для начала воспроизведения трансляции другого качества. Каким же образом можно решить задачу организации адаптивной бесшовной трансляции для технологии опережающей загрузки потока, в которой клиент является пассивным участником?

В настоящей работе предлагается решение этой задачи. Адаптивность и бесшовность достигается тем, что процессом изменения качества видеоизображения управляет не клиент, а сервер. Сервер непрерывно вычисляет текущую ("эффективную") пропускную способность канала клиента и автоматически меняет качество передаваемого клиенту потока. Клиент при этом просто воспроизводит тот поток, который поступает ему от сервера. Такой подход диктует следующие условия на метод кодирования потока:

- метаданные (заголовок потока) должны передаваться перед медиаданными;
- изменение битрейта потока путем изменения качества изображения должно осуществляться *без изменения метаданных* (т. е. заголовка потока). Метаданные потока отправляются клиенту только один раз при его подключении к серверу, после чего качество создаваемых кадров меняется по ходу проведения трансляции;
- контейнер потока должен быть потоковым (однопроходным), т. е. декодирование медиаданных должно основываться на информации, содержащейся в самих текущих медиаданных, и не требовать предыдущих медиаданных.

Примером такого контейнера является контейнер Ogg [7] с видеокодеком Theora [8] и аудиокодеком Vorbis [9].

1. Адаптивная трансляция в manifest-технологиях

Схема работы механизма адаптивности для manifest-технологий следующая. Имея несколько *взаимозаменяемых* потоков разных битрейтов для одной и той же презентации, можно по ходу проведения трансляции подменять фрагменты презентации фрагментами с той же самой видео- и аудиоинформацией, но с другим битрейтом. Сервер создает несколько потоков разного качества (битрейта) для одной

и той же презентации. Клиент загружает поток презентации маленькими порциями — фрагментами длительностью воспроизведения в несколько секунд. Каждый фрагмент имеет уникальный URL. Клиент через периодически скачиваемый с сервера файл-манифест знает список всех имеющихся на сервере потоков, их битрейты, а также список доступных для скачивания фрагментов (т. е. их URL) каждого из этих потоков. Тем самым, клиенту всегда известны ссылки на свежие фрагменты одной и той же живой презентации, но с разным битрейтом. Основываясь на скорости загрузки фрагментов, клиент делает вывод о пропускной способности канала и может запросить следующий фрагмент с тем битрейтом (качеством), которое наиболее соответствует текущему состоянию канала передачи данных. Для того чтобы переключение потоков было бесшовным, начало каждого фрагмента, на котором проводится переключение потока, обязательно должно совпадать с началом кадра.

2. Адаптивная трансляция в технологии опережающей загрузки. Постановка задачи и общая схема ее решения

В случае технологии живой трансляции методом опережающей загрузки передача потока презентации от сервера к клиенту проводится также фрагментами, однако реализация этой передачи иная, чем описанная в предыдущем параграфе. Клиент не получает от сервера манифест и, соответственно, ничего не знает о существующих вариантах качеств потоков. Более того, он не запрашивает каждый раз фрагменты потока, а пассивно принимает их от сервера. При таком подходе организация бесшовной адаптации возможна, если решение о переключении отсылаемых клиенту фрагментов будет принимать не клиент, а сервер.

В настоящей работе предлагается следующая схема организации адаптивности. Источник презентации посылает серверу несколько потоков разного битрейта (рассмотрим случай двух). Сервер должен постоянно определять пропускную способность канала и посылать клиенту фрагменты того потока, битрейт которого более всего соответствует текущей пропускной способности канала. В случае необходимости смены битрейта сервер должен подменять фрагменты так, чтобы переключение делалось именно на границе кадра, а не на его середине, чем будет обеспечена бесшовность переключения.

На рис. 1 (см. вторую сторону обложки) проиллюстрирован случай, когда сервер вначале посылал клиенту поток высокого качества, затем некоторое время посылал поток низкого качества, после чего снова вернул потоку высокое качество. При этом последовательность кадров не изменилась: $n-1$, n, $n+1$, ..., $m-1$, m.

В предложенной выше схеме остался нераскрытым следующий вопрос — каким образом сервер сможет узнать текущую пропускную способность канала, чтобы осуществить переключение качества в нужную сторону?

Эффективная пропускная способность канала

Пропускной способностью канала называется наибольшая возможная скорость передачи данных. Таким образом, пропускная способность — это характеристика канала, имеющая отношение к его конструктивным свойствам и не имеющая отношение к загрузке канала. Пропускную способность канала можно измерить, проведя достаточно большое число измерений скорости передачи пакетов определенной длины (скажем, 512 байт) и найдя наибольшее значение из них. Если канал сильно загружен, то скорость доставки фрагмента, равная пропускной способности канала, достигается лишь для очень малого числа пакетов, тогда как для остальных пакетов скорость доставки окажется значительно ниже. В этом случае для пользователя гораздо большее значение имеет *эффективная пропускная способность* (ЭПС) канала — фактическая пропускная способность, которая может значительно меняться от степени загрузки канала в каждый момент времени. Эта величина представляет собой не характеристику канала, а переменную во времени характеристику загрузки канала. Измерить эффективную пропускную способность можно путем измерения скорости передачи данных при посылке максимально возможного объема данных за заданный интервал времени $\Delta t_{\text{ЭПС}}$. Вследствие неравномерности загрузки канала при разных измерениях мы будем получать разные значения ЭПС.

На рис. 2 (см. вторую сторону обложки) приведен пример типичной картины изменения ЭПС во времени. Мы видим, что временной тренд ЭПС в целом "поджат" к верхнему порогу $9 \cdot 10^6$ байт/с, что соответствует теоретической пропускной способности использовавшегося канала в 100 Мбит/с. Тренд несимметричен по оси ординат и содержит нерегулярные провалы в направлении меньших значений пропускной способности. Их значение и частота должны возрастать с ухудшением работы канала связи, например, при увеличении общей нагрузки на канал от разных пользователей.

Адаптивность и ЭПС

Суть использования адаптивности заключается в том, чтобы битрейт передаваемого клиенту потока был всегда ниже ЭПС. В этом случае скорость воспроизведения потока на клиенте всегда будет оставаться меньше, чем скорость передачи фрагментов, вследствие чего приемный буфер клиента будет всегда достаточно заполнен. В изображенном на рис. 2 случае в течение первых 20 мин трансляции оптимальный битрейт передаваемого клиенту потока не должен был превышать $6 \cdot 10^6$ байт/с (зеленая линия). В момент времени "B" (1260 с) ЭПС вдруг "просела" и далее в течение более чем трех минут регулярно оказывалась ниже $6 \cdot 10^6$ байт/с. Если не менять битрейт потока, то приемный буфер клиента рано или поздно опустошится, вследствие чего возникнет задержка воспроизведения видеоинформации. Поэтому в момент времени "B" сервер должен был переключиться на поток с меньшим битрейтом $4 \cdot 10^6$ байт/с (красная линия).

В момент времени 1400 с восстановилась прежняя ЭПС. Сервер, конечно, может и не предпринимать никаких действий и продолжать по-прежнему передавать поток с низким битрейтом. Но все же неплохо было бы воспользоваться улучшением ЭПС канала и переключить поток обратно на высокое качество $6 \cdot 10^6$ байт/с (зеленая линия).

Изображенная на рис. 2 ситуация длительного (более секунды) изменения ЭПС возникает не всегда. Весьма частыми событиями являются однократные понижения ЭПС — "просадки". Если длительность однократной "просадки" такова, что это не приведет к опустошению видеобуфера клиента, то такое событие не вызовет задержки отображения видеоинформации. Однако каждая такая "просадка" будет приводить к уменьшению накопленных данных в приемном буфере клиента. Например, в течение времени от "A" до "B" сервер мог бы передавать клиенту поток с битрейтом $7 \cdot 10^6$ байт/с. При этом мы наблюдаем четыре однократные "просадки" ЭПС ниже уровня $7 \cdot 10^6$ байт/с. При каждой такой "просадке" приемный видеобуфер клиента уменьшается на длительность воспроизведения:

$$\Delta t_{\text{прием.буфер}} = \Delta t_{\text{просадки}}(1 - Bwe/Br), \qquad (1)$$

где Bwe — ЭПС в момент "просадки"; Br — битрейт передаваемого клиенту потока; $\Delta t_{\text{просадки}}$ — общая длительность "просадки".

Если "просадка" недолгая, то приемный видеобуфер клиента не успевает опустошиться, и через некоторое время он снова будет заполнен, так как в итоге все задержанные фрагменты потока будут переданы клиенту "в срок". Определить, является очередное уменьшение ЭПС кратковременным или нет, сервер конечно не может и *обязан* переключиться на менее качественный поток. Если "просадка" ЭПС недолгая, то после нее сервер тут же вернется к передаче клиенту предыдущего потока с высоким битрейтом. Произойдет однократное кратковременное адаптивное переключение качества потока "высокий битрейт → низкий битрейт → высокий битрейт", которое, по сути, будет паразитным. Частота таких однократных паразитных переключений качества увеличивается с возрастанием частоты "просадок" и уменьшается с возрастанием длительности времени измерения ЭПС $\Delta t_{\text{ЭПС}}$. Все, что мы можем сделать для уменьшения числа таких однократных переключений качества — это выбрать некий *оптимальный* интервал $\Delta t_{\text{ЭПС}}$ измерения ЭПС.

Совмещение измерения ЭПС с процессом трансляции потока

В процессе трансляции потока от сервера к клиенту проводить какие-то специальные измерения ЭПС канала "сервер—клиент" избыточно, поскольку это можно делать с помощью самого же отправляемого потока. Это позволяет совместить передачу потока с постоянным измерением ЭПС этого же потока "в режиме online". Для этого сервер должен посылать данные клиенту не непрерывно по мере их поступления от источника презентации, а накапливать данные в буфере FIFO и отсылать блоками так, чтобы *длительность отсылки блока* клиенту равнялась заданному значению $\Delta t_{\text{ЭПС}}$. В настоящей работе в качестве оптимального было принято значение $\Delta t_{\text{ЭПС}} = 1000$ мс.

Как было отмечено в предыдущем параграфе, величина $\Delta t_{\text{ЭПС}}$ не должна быть чересчур малой (менее секунды), так как это может привести к частым паразитным переключениям битрейта потока. Вместе с тем она и не должна быть слишком большой (более десяти секунд), так как тогда измеренная ЭПС будет представлять усредненное значение ЭПС канала связи за эти десять секунд и не будет отражать его текущего состояния. Кроме того, значение задержки воспроизведения презентации на клиенте прямо связано со временем накопления блока. Длительность накопления $\Delta t_{\text{накопл}}$ блока, который потом будет отправлен клиенту за время $\Delta t_{\text{ЭПС}}$, равна

$$\Delta t_{\text{накопл}} = \Delta t_{\text{ЭПС}} \cdot Bwe/Br. \qquad (2)$$

Если битрейт потока (*Br*) в несколько десятков раз меньше ЭПС канала клиента (*Bwe*), то выбрав $\Delta t_{ЭПС}$ = 1000 мс, получим накопления блока в несколько десятков секунд и соответственно такую же задержку воспроизведения живой презентации на клиенте.

Строго говоря, необходимость введения блочной передачи потока требуется только для адаптивного переключения на повышенный битрейт. Если же ЭПС канала упала, то это можно понять уже по нескольким первым фрагментам отправляемого блока.

Бесшовное переключение качества

Требование бесшовности накладывает условие: переключение на поток другого качества (битрейта) не должно выполняться в произвольный момент времени, а только в момент начала нового кадра. Таким образом, блоки в FIFO должны содержать целое число кадров и быть взаимозаменяемыми соответствующими блоками FIFO потока другого битрейта. На рис. 3 (см. вторую сторону обложки) представлен пример переключения отправляемому клиенту потока, иллюстрирующий ситуацию, показанную на рис. 2. Отдельные прямоугольники изображают кадры, которые двумя потоками с разным качеством поступают от источника презентации на сервер (ретранслятор), размеры прямоугольников отражают размеры кадров: кадры большего качества имеют большие размеры. Кадры поступают в два буфера FIFO соответственно и накапливаются там блоками. Каждый блок изображен горизонтальной полоской, состоящей из нескольких кадров. Блок потока высокого качества изображен состоящим из трех кадров, а блок потока низкого качества состоит из 10 кадров. Поскольку время отсылки каждого блока, независимо от его качества, одинаковое (1000 мс), то размеры блоков разного качества изображены также одинаковыми.

В процессе отсылки блока сервер может обнаружить падение ЭПС, при этом он должен переключиться на поток с меньшим качеством. Он делает это не сразу, а сначала дожидается отсылки текущего кадра клиенту. Это проиллюстрировано укороченными блоками, но все равно состоящими из целого числа кадров.

От начала трансляции "A" до момента времени "B" сервер отправлял клиенту поток высокого качества. Он успел передать несколько блоков, после чего в момент времени "B" произошла "просадка" ЭПС. В процессе отправки очередного блока сервер обнаружил "просадку", дождался завершения отправки текущего кадра и переключил отправку на поток низкого качества. В момент времени "C" ЭПС канала восстановилась до прежнего значения, сервер дождался завершения отправки текущего кадра и переключил отправку на поток высокого качества.

Сервер должен структурировать все поступающие ему медиапотоки на отдельные кадры и отслеживать последовательность номеров кадров в каждом потоке. Если сервер выявил необходимость переключения на поток другого битрейта, то прежде всего он должен дождаться завершения отправки клиенту данных текущего кадра *N*. Затем найти кадр $N + 1$ в буфере FIFO потока необходимого битрейта, дождаться накопления в нем блока (начиная с кадра $N + 1$) такого размера, чтобы его пересылка клиенту составляла величину 1000 мс и только потом отправить этот блок.

Область применимости и эффективность предлагаемого подхода

Рассмотрим случай, когда ЭПС клиентского канала превышает битрейт потока наибольшего качества. По сути, в этом случае адаптивность уже не требуется, и необходимости в блочной передаче нет. Тем не менее сервер все равно должен постоянно контролировать ЭПС, чтобы не пропустить ситуацию с ее ухудшением. Для этого сервер должен накапливать блок в течение времени, которое может оказаться весьма значительным (равным десяткам секунд). Например, в соответствии с формулой (2) для типичной ЭПС = 10^7 байт/с и потока с битрейтом $Br = 2 \cdot 10^5$ байт/с (копия экрана 1280 × 1024, 30 кадров/с, кодек Theora) серверу придется каждый раз ожидать 50 с, чтобы накопить очередной блок, который он будет потом отправлять клиенту в течение 1 с. Такого же порядка (50 с) будет и задержка воспроизведения презентации на клиенте. И чем больше ЭПС будет превышать битрейт потока, тем дольше сервер будет накапливать блоки. В принципе, если трансляция не налагает требований реального времени, такая задержка не будет представлять серьезную проблему для клиента. Тем не менее, чтобы не снижать эффективность использования блоков, требуется ограничить сверху размер блока. В настоящей работе максимальный размер блока был ограничен значением 1 Мбайт.

3. Практическая реализация изложенной схемы бесшовной адаптивной подстройки качества видеоизображения

Идеи, изложенные выше, были реализованы практически путем расширения разработан-

ной ранее системы трансляций презентаций в реальном времени (диспетчерской системы) [10], структурная схема которой изображена на рис. 4 (см. третью сторону обложки).

Диспетчерская система состоит из трех базовых частей:

1. Кодировщик медиапотока, который принимает медиаданные с источника презентации (персональный компьютер, ip/web камера, микрофон), кодирует их в поток Ogg с кодеками Theora / Vorbis и отправляет кодированные медиаданные на Ретранслятор.

2. Ретранслятор, который принимает медиаданные от нескольких Кодировщиков и ретранслирует эти потоки клиентам, а статусную информацию обо всех медиапотоках помещает в базу данных. Один и тот же медиапоток может отправляться нескольким клиентам одновременно.

3. Набор HTML-страниц на web-сервере для выбора и просмотра клиентом презентаций. Страницы формируются автоматически, основываясь на информации из базы данных.

Расширение диспетчерской системы на адаптивность потребовало ряда изменений в структуре диспетчерской системы, рассмотренной в работе [10], которые базировались на уже функционирующей системе промежуточных FIFO-буферов. Первое изменение состояло во введении структурирования потоков от Кодировщика к Ретранслятору, которые в системе [10] просто передавались Ретранслятору "как есть" по мере кодирования презентации. В адаптивной версии диспетчерской системы к потоку был добавлен специальный пакет — маркер начала кадра, который передается Ретранслятору Кодировщиком перед каждым кадром. Ретранслятор должен сам надежно выявлять наличие маркера начала кадра в потоке данных.

Второе изменение состояло в том, что Кодировщик медиапотока стал кодировать презентацию не в один, а в три параллельных потока — высокого, среднего и низкого качества. Эти три потока отправляются на Ретранслятор по трем различным TCP-портам. Ретранслятор хранит медиаданные каждого потока в отдельном кольцевом буфере FIFO. Каждый буфер FIFO теперь разбивается на блоки, в каждом из которых содержится по нескольку целых кадров, их число определяется изложенным выше алгоритмом. Ретранслятор пересылает клиенту медиаданные блоками по мере их появления в буфере FIFO.

Третье изменение касалось алгоритма работы Ретранслятора. Теперь Ретранслятор постоянно следит за двумя величинами:

- скоростью поступления потока в FIFO с Кодировщика (*Bwe*);
- скоростью отправки блока данных из FIFO Клиенту (*Br*).

Ретранслятор переключает качество отправляемого Клиенту потока по условию повышения и понижения *Br* относительно *Bwe*.

4. Проверка и анализ реализации бесшовной адаптивной подстройки качества

Тестирование адаптивной версии системы трансляций проводили путем искусственного изменения пропускной способности канала клиента с помощью коммутатора второго уровня с фиксированными значениями пропускной способности (512, 1024, 2048, ..., Unlimit kbit/sec). Использовали три градации качества видеопотока, получаемых разной степенью сжатия при Theora-кодировании, причем разрешение экрана для всех качеств было одинаковым 640 × 480@20 fps. На рис. 5 (см. третью сторону обложки) представлен типичный результат изменения ЭПС канала клиента (тренд изображен синим цветом) относительно битрейтов трех принимаемых потоков от Кодировщика. Высокий, средний и низкий битрейты изображены фиолетовым, зеленым и красным цветами соответственно. При этом в процессе изменения ЭПС канала клиента Ретранслятор совершил три переключения качества — от высокого к среднему, от среднего к низкому и от низкого к среднему:

- в момент времени 180 с ЭПС упала до ~350 000 байт/с, что уже сопоставимо с битрейтом H-потока, поэтому сервер переключил поток клиента на среднее качество (H → M);
- в момент времени 210 с ЭПС упала до ~60 000 байт/с, и сервер переключил поток клиента на плохое качество (M → L);
- в момент времени 810 с ЭПС возросла до ~350 000 байт/с, и для клиента появилась возможность воспроизводить поток среднего качества. Сервер переключил качество на среднее (L → M).

В моменты переключения качества передаваемого клиенту потока воспроизведение видео- и аудиоинформации на клиенте продолжает идти плавно, без потерь видеокадров и аудиосэмплов. При резком падении ЭПС канала клиента остановок воспроизведения не наблюдалось, так как при падении приемный буфер клиента не успевал опустошаться из-за своевременного изменения качества потока.

Дополнительно проводилось сравнение процесса адаптивного переключения качества системы трансляций [10] с работой сервиса YouTube. Сравнение велось по двум критериям — битрейту установившегося потока и задержке между реальным событием и его воспроизведением в браузере клиента. В системе трансляций использовались три уровня качества потока одинакового разрешения 1280 × 720@30 fps и с разной степенью сжатия Theora-кодирования. Сервис YouTube предоставляет пять градаций качества, отличающихся разрешениями экрана 144p (254 × 144@30 fps), 240p (426 × 240@30 fps), 360p (640 × 360@30 fps), 480p (854 × 480@30 fps), 720p (1280 × 720@30 fps).

При искусственном установлении пропускной способности канала на фиксированные уровни 512, 1024 и 2048 Кбит/с в обоих случаях происходило адаптивное переключение трансляции на поток подходящего качества со сравнимыми битрейтами (см. таблицу).

Адаптивные битрейты установившихся потоков сравниваемых технологий для заданных значений пропускной способности

Система конференций, битрейт потока	YouTube, битрейт потока	Пропускная способность канала
2000 Кбит/с (высокое качество)	2000 Кбит/с (720p)	2048 Кбит/с
1000 Кбит/с (среднее качество)	640 Кбит/с (480p)	1024 Кбит/с
370 Кбит/с (низкое качество)	240 Кбит/с (240p)	512 Кбит/с

Минимальная задержка между реальным событием и его воспроизведением для системы трансляций [10] составила 10...13 с. Соответствующий параметр плеера YouTube "Live Latency" давал значение в районе 30 с, что более чем вдвое превысило полученный выше результат. Впрочем, задержку "Live Latency" можно искусственно уменьшить. Идея состоит в том, что частью этой задержки является буферизация (около 20 с), соответственно время декодирования составляет 10 с. Можно ускорить воспроизведение до такой степени, когда буферизация составит 5 с, после чего вернуть нормальную скорость воспроизведения. Тогда общая задержка YouTube будет равной 15 с, однако воспроизведение при этом становится неустойчивым.

Заключение

В настоящее время разработан и активно используется ряд технологий адаптивной трансляции в реальном времени [1—4]. Эти технологии объединяет тот факт, что клиент сам определяет эффективную пропускную способность сети и организует адаптивность путем загрузки фрагментов презентации того качества, которое является оптимальным в данный момент.

Для рассмотренной в работе технологии трансляции методом опережающей загрузки такой способ организации адаптивности невозможен, так как клиент является пассивным участником процесса трансляции. В настоящей работе предложено решение этой задачи. В основе решения лежит идея о том, что инициатором адаптивной подстройки является не клиент, а сервер. В работе обсуждено поведение ЭПС канала клиента и предложена идея совмещения измерения ЭПС с отправкой клиенту потока. Для этого сервер посылает клиенту медиаданные не равномерно, а группирует их в блоки. Во время отсылки байтов блока сервер определяет ЭПС сети клиента. Сервер переключает передаваемый клиенту поток на битрейт, который наиболее подходит для ЭПС канала клиента. Для обеспечения бесшовности переключения потока на другое качество сервер всегда заканчивает отсылку кадра текущего потока и начинает отсылку блоков потока другого качества строго на границе кадра. Область применимости предложенного решения адаптивной трансляции определяется условием на соотношение между ЭПС канала клиента и битрейтом потока: ЭПС канала клиента не должна превышать битрейт потока самого высокого качества.

Предложенный в работе способ организации адаптивности был реализован на практике путем расширения разработанной ранее системы трансляций [10]. Ретранслятор реагирует на все изменения эффективной пропускной способности канала клиента и своевременно переключает передаваемый ему поток на другое качество. При уменьшении ЭПС канала происходит переключение потока на меньший битрейт, буфер клиента не опустошается, а последовательность кадров не прерывается. В итоге клиент воспроизводит трансляцию без перерывов, аудио-, видеоискажений или задержек.

Сравнение работы системы трансляций для 1280 × 720@30 fps с работой видеосервиса YouTube по критерию "битрейт потока" дало сравнимые результаты, а для критерия "задержка между реальным событием и его воспроизведением" система трансляций дала более хороший результат.

В сравнении с обычными методами переключения качества потока, когда решение

о переключении принимает клиент, в настоящей работе предложена схема адаптивности, в которой решение о переключении принимает сервер. Это является главной особенностью используемого в работе подхода к организации адаптивности, и она может представлять собой определенное преимущество по сравнению с другими технологиями для тех случаев, когда клиент не может использовать специальные плагины или расширения для воспроизведения потока в условиях изменяющейся пропускной способности канала. От клиента в этом случае лишь требуется поддержка опережающей загрузки потока в рамках спецификации HTML5.

Список литературы

1. **Pantos R., May W.** HTTP Live Streaming. draft-pantos-http-live-streaming-18. Apple Inc. 2015. 49 p.

2. **HTTP Dynamic Streaming Specification Version 3.0 FINAL.** Adobe Systems Incorporated. 2013. 31 p.

3. **Smooth Streaming Protocol.** [MS-SSTR] — v20160714. Microsoft Corporation. 2016. 64 p.

4. **ISO/IEC 23009-1.** Information technology — Dynamic adaptive streaming over HTTP (DASH) — Part 1: Media presentation description and segment formats. Second edition. 2014. 144 p.

5. **Лобов. И. В., Готман В. Г.** Трансляция мультимедиа в реальном времени над протоколом HTTP методом опережающей загрузки // Технологии и средства связи. 2016. № 5. С. 36—40.

6. **Лобов И. В., Готман В. Г.** Использование контейнера Ogg для организации потоковой трансляции в реальном времени над протоколом HTTP методом опережающей загрузки // Информационные технологии. 2018. № 2. С. 87—96.

7. **Pfeiffer S.** The Ogg Encapsulation Format Version 0. Request for Comments: 3533, 2003, 15 p.

8. **Theora Specification.** Xiph.Org Foundation, 2011, 196 p.

9. **Vorbis I specification.** Xiph.Org Foundation, 2015, 74 p.

10. **Лобов И. В., Готман В. Г.** Система трансляций презентаций в реальном времени над протоколом HTTP // Известия Института инженерной физики (ИИФ). 2018. № 3. С. 60—66.

I. V. Lobov, Senior Researcher, e-mail: lobov@ihep.ru,
V. G. Gotman, Junior Researcher, e-mail: vladislav.gotman@ihep.ru,
Institute for High Energy Physics, National Research Center "Kurchatov Institute",
Moscow, Russian Federation

Adaptive Bitrate Seamless Live Streaming over HTTP by Progressive Download Method

The paper proposes a technology for organizing the adaptive bitrate seamless live adjustment of image quality while changing the client channel effective in live streaming system based on the proggresive download. A server hosted method of measuring client channel bandwidth while live streaming has been proposed. Diverse technical issues related to the implementation of adaptive bitrate seamless streaming were discussed. Applicability conditions for the adaptive image quality adjustment for the technology under consideration were determined. A fair-functioning dispatch system that implements the adaptive bitrate seamless live streaming by proggresive download method has been described.

Keywords: *adaptive bitrate streaming, live streaming, progressive download, Ogg format, Apple HLS, Adobe HDS, Microsoft Smooth Streaming, MPEG DASH*

DOI: 10.17587/it.26.177-184

References

1. **Pantos R., May W.** HTTP Live Streaming. draft-pantos-http-live-streaming-18, Apple Inc, 2015, 49 p.

2. **HTTP Dynamic Streaming Specification Version 3.0 FINAL.** Adobe Systems Incorporated, 2013, 31 p.

3. **Smooth Streaming Protocol.** [MS-SSTR] — v20160714. Microsoft Corporation, 2016, 64 p.

4. **ISO/IEC 23009-1.** Information technology — Dynamic adaptive streaming over HTTP (DASH) — Part 1: Media presentation description and segment formats. Second edition, 2014, 144 p.

5. **Lobov I., Gotman V.** Real time media streaming over HTTP using progressive download method, *Tehnologii i Sredstva Svyazi*, 2016, no. 5, pp. 36—40 (in Russian).

6. **Lobov I., Gotman V.** The Utilization of Ogg Multimedia Container Format for Live Streaming over HTTP Using Progressive Download Method, *Informatsionnye Tekhnologii*, 2018, no. 2, pp. 87—96 (in Russian).

7. **Pfeiffer S.** The Ogg Encapsulation Format Version 0. Request for Comments: 3533, 2003, 15 p.

8. **Theora Specification.** Xiph.Org Foundation, 2011, 196 p.

9. **Vorbis I specification.** Xiph.Org Foundation, 2015, 74 p.

10. **Lobov I., Gotman V.** Live streaming system over http, Izvestiya Instituta Inzhenernoj Fiziki (IIF), 2018, no. 3, pp. 60—66 (in Russian).

УДК 004.89 DOI: 10.17587/it.26.185-191

Е. А. Басыня, канд. техн. наук, доц.[1], директор[2], e-mail: director@nii-ikt.ru,
[1]Новосибирский государственный технический университет,
[2]Научно-исследовательский институт информационно-коммуникационных технологий, г. Новосибирск

Метод интеллектуально-адаптивного управления информационной инфраструктурой предприятия

Предлагается новый метод интеллектуально-адаптивного управления информационной инфраструктурой предприятия, позволяющий обеспечить исправное и отказоустойчивое функционирование технических систем и объектов со снижением загрузки канала связи. Выбор рациональной стратегии реагирования на различные типы воздействий осуществляется их интеллектуальной обработкой с прогнозированием реакции сервисов на изолированных модельных объектах.

Ключевые слова: *системный анализ, интеллектуально-адаптивное управление, обработка, сетевой трафик, локальные информационные процессы, нештатные воздействия, информационная безопасность, TCP/IP, угрозы, атаки*

Введение

Развитие информационных технологий является глобальным трансграничным процессом, затрагивающим все сферы деятельности общества. Повышение эффективности, надежности, отказоустойчивости и качества различных технических систем являются приоритетными задачами научного сообщества и бизнеса. Их успешное решение позволяет оптимизировать информационные и рабочие процессы хозяйствующих субъектов, повысить рентабельность их функционирования. На сегодняшний день практически любой вид экономической деятельности использует сетевое взаимодействие на основе стека протоколов TCP/IP (англ. Transmission Control Protocol/ Internet Protocol). Технические отрасли не составляют исключение, невозможно представить современную автоматизацию любого производственного процесса без использования информационных технологий.

Важно отметить, что задачи системного анализа, управления и обработки информационных потоков и процессов являются приоритетными, смещается вектор их развития в сторону программных и аппаратно-программных решений. Уровень информационной безопасности предприятия становится следствием эффективности решения данных задач.

Разработкой методов управления информационными потоками и процессами занимаются российские и зарубежные ученые: О. Б. Калугина, С. М. Трошина, Н. В. Штуллер, L. Sung-Ho, P. Jun-Sang, J. Woo-Suk, P. Jun-Sang и др. [1—4]. Предлагаемые подходы обеспечивают надежную и отказоустойчивую работу информационных систем посредством использования сигнатурной обработки трафика. Другой интересный подход излагается в работах Р. Р. Файзулина, А. Я. Инсарова, L. He, Y. Cuibo, G. Xuerong, H. Jieying, J. Z. Zhang, L. Lishi и заключается в идентификации автомодельности сетевых потоков [5—8].

К сожалению, данные методы не выполняют поиск рациональной стратегии реагирования на несанкционированные внутренние и внешние воздействия различного уровня риска. Осуществляется лишь сигнатурная обработка инцидентов или идентификация автомодельности трафика, которые в случае отсутствия корректных начальных условий не могут быть выполнены. В качестве простого примера стоит упомянуть шифрование информацион-

ных потоков и использование виртуальных защищенных каналов связи.

Под рациональной стратегией реагирования подразумевается комплекс мер, позволяющий максимально снизить загрузку канала связи в сравнении с альтернативными решениями, но без негативных воздействий на штатные информационные потоки и процессы, а также на уровень безопасности информационно-коммуникационного сектора предприятия. При этом важно отметить, что производители программных и аппаратно-программных сетевых решений могут иметь различные правила обработки идентичных инцидентов.

Соответственно, возрастает актуальность разработки проблемно-ориентированных методов управления сетевым трафиком и локальными информационными процессами, позволяющих в автоматическом режиме проводить поиск и применение рациональной стратегии реагирования.

1. Цель работы

Целью данной работы являлась разработка нового метода интеллектуально-адаптивного управления информационной инфраструктурой предприятия. Необходимо было обеспечить исправное и отказоустойчивое функционирование технических систем и объектов со снижением загрузки каналов связи при разнообразных внутренних и внешних воздействиях различного уровня риска.

Следовало минимизировать риск перехода информационных систем в режим недоступности. Требовалось обеспечить высокий уровень безопасности информационной инфраструктуры предприятия без ущерба штатным информационным потокам, которые могли бы проводиться и с применением технологий анонимизации.

Ставилась задача разработки концепции автоматического поиска и применения рациональной стратегии реагирования на различные типы возмущений с возможностью самообучения и самоорганизации правил и модулей управления.

2. Предлагаемое решение

Целевое и практическое назначение предлагаемого метода — обеспечение функционирования авторской программной системы интеллектуально-адаптивного управления информационной инфраструктурой предприятия (далее именуемой "Система" или "СИАУ ИИП"). Ее разработка, проектирование, программная реализация и исследование будут представлены в следующей статье.

Рассмотрим метод интеллектуально-адаптивного управления информационной инфраструктурой предприятия на примере работы СИАУ ИИП (рис. 1, см. четвертую сторону обложки).

Система осуществляет управление всеми объектами информационной инфраструктуры предприятия: пользовательскими электронно-вычислительными машинами, выделенными и виртуальными серверами, управляемым сетевым оборудованием (коммутаторами, маршрутизаторами и другими объектами), авторскими наукоемкими системами и сервисами, техническими объектами и системами промышленности (в том числе компонентами автоматизированных систем управления технологическим процессом), функционирующими на основе стека протоколов TCP/IP.

Повышение надежности, отказоустойчивости и качества технических систем достигается комплексным управлением трафиком и информационными процессами на всех уровнях взаимодействия.

2.1. Задача управления информационными потоками и процессами

Задача управления трафиком вычислительной сети может быть решена различными способами. Оценка эффективности вариантов решения сводится к сравнению загрузки канала связи при условии сохранения штатного режима обработки легитимных информационных потоков и процессов (рис. 2, см. четвертую сторону обложки). На программном уровне проводится ряд наблюдений за изменением загрузки канала связи в определенный интервал времени. Осуществляется оценка быстродействия принятия решений с мониторингом уровня безопасности информационно-коммуникационного сектора предприятия.

Сетевое оборудование (маршрутизаторы, межсетевые экраны) имеют несколько сетевых интерфейсов, организующих разные каналы связи в рамках локальных или глобальных сетевых взаимодействий. Соответственно, для каждого из них в параллельном режиме может осуществляться поиск рациональной стратегии обработки различных типов сетевых воздействий.

В качестве математической абстракции описания данного подхода может быть введен безразмерный критерий рациональности J, представляющий суммарный критерий снижения загрузки линии связи J_1 и быстродействия J_2:

$$J = J_1 + J_2 = F(u, h, t) + \frac{kr}{d}T, \qquad (1)$$

где $F(u, h, t)$ — функция загрузки канала, в связи с отсутствием возможности представления в математическом виде рассматриваемая как нелинейная, псевдослучайная, описывающая изменение пропускной способности вычислительной сети во времени t; h — нежелательные возмущения; $u \in \Omega_u$ — управляющее воздействие (набор команд модулям системы согласно стратегии реагирования), Ω_u — рабочая область управляющих воздействий; T — время принятия решений по управлению; k — стоимостной коэффициент, определяемый новизной управляющего воздействия (чем "новее" решение, тем меньше коэффициент; тем самым предоставляется небольшое окно по времени поиска новых решений, а не использованию старых; базовый интервал начальных значений: [0,5; 1], автономно корректируются системой для различных типов активности); d — коэффициент достоверности определения типа возмущения, $d \in [0; 1]$; r — коэффициент риска, который назначен системой определенному виду нештатных воздействий, $r \in [0, \infty)$. Бесконечность подразумевает, что решение не может быть принято ни при каких условиях, так как несет риски информационной безопасности, сопровождающие его принятие.

Под внешними воздействиями понимаются информационные потоки трафика вычислительной сети, предназначенные непосредственно Системе или объектам информационной инфраструктуры предприятия, которыми она управляет. Это могут быть штатные информационные взаимодействия, несанкционированные внешние или внутренние возмущения, способные нанести существенный вред нормальному функционированию технических объектов. Природа последних запросов может быть обусловлена как аппаратно-программными неполадками, так и злоумышленными намерениями сторонних лиц или сотрудников из числа доверенных пользователей.

Существующие системы управления не являются комплексными продуктами, включают атомарный функционал и используют "жесткую" логику поведения, не проводящую сравнительного анализа и поиска рационального решения (снижающего загрузку канала связи и обеспечивающего исправное, надежное, отказоустойчивое и безопасное функционирование технических систем и объектов). Это позволяет хакерам однозначно идентифицировать продукт защиты атакуемого объекта посредством инструментов активного и пассивного анализа трафика и информационных ресурсов (сканеров/зондеров, инструментов пентеста и др.) с последующей эксплуатацией уязвимости для нарушения работоспособности узлов/сети.

2.2. Блок генетической алгоритмизации и нечеткой логики

В целях нивелирования описанных рисков было решено спроектировать и реализовать методы, обладающие интеллектуально-адаптивными свойствами. Важно отметить, что в рассматриваемой задаче поиска рационального решения в управлении информационными потоками существует значительный недостаток априорной информации о структуре всех объектов и систем, а также о характерах возмущений. Для получения устойчивых решений в данной ситуации было сделано заключение о применении генетической алгоритмизации (ГА), обладающей гибкостью функционирования, возможностью выхода из локальных на глобальные экстремумы, возможностью эффективного распараллеливания вычислений, высокой скоростью поиска решений на нелинейных функциях и другими преимуществами.

Необходимость принятия решений в условиях приближенных рассуждений аргументировало применение связки генетической алгоритмизации с нечеткой логикой (англ. fuzzy logic). Технологии нейронных, гибридных и других сетей являются избыточным инструментом в срезе исследуемой задачи.

Поскольку в задачах снижения загрузки канала связи и обеспечения исправного, надежного, отказоустойчивого и безопасного функционирования технических систем и объектов некоторые экземпляры решений недопустимы (могут нарушить штатное функционирование инфраструктуры предприятия), а также в связи с необходимостью реализовать механизмы подмены и прогнозирования реакций на модельных объектах (МО) разрабатываемой Системы была проведена модернизация блока генетической алгоритмизации. Введен контур функционирования нечеткой логики с блоками прогнозирования на модельных объектах СИУА ИИП,

обработки результатов прогнозирования с дополнительной множественной фильтрацией после создания поколений (рис. 3). Под модельными объектами понимается эталонная копия актуального состояния системы и реальных корпоративных серверных решений и сервисов.

Для более качественного распознавания неблагоприятных сетевых воздействий и предотвращения возможных побочных эффектов от принятых решений система формирует аддитивный фоновый тестовый сетевой трафик для модельных объектов. Удостоверившись, что экземпляр решения уменьшает нагрузку на систему в целом и одновременно с этим не препятствует прохождению полезного трафика, фильтр системы может признать решение допустимым.

С использованием модельных объектов СИАУ ИИП осуществляет прогнозирование реакции системы/объектов, по обратной связи корректируется начальная выборка (новые поколения). Идентичная подстройка фильтра с использованием нечеткой логики позволяет выбирать из правильных решений рациональное, что приводит к снижению загрузки канала связи. Осуществляется нивелирование негативных воздействий от потенциально подозрительных возмущений различного уровня риска с обеспечением исправного обслуживания штатных воздействий.

Концепция генетической алгоритмизации заключается в организации эволюционного процесса и наследует идеи природы. Популяцией особей выступает конечное множество альтернативных решений. Хромосомы выражают составляющие действия в решении (стратегии реагирования) и являются упорядоченными последовательностями генов, описывающих параметры задачи. В качестве оценки приспособленности предлагается использовать стоимостную функцию

$$\omega_i(\Delta T) = \sum_{n\in[0,\,N-1]} |traffic(n) - \overline{traffic}| = N \cdot \overline{traffic} + \frac{}{2}, \quad (2)$$

n — отсчеты в рассматриваемом интервале времени ΔT; N — число измерений загрузки канала связи в данном интервале; $\overline{traffic}$ — среднее арифметическое значение загрузки канала за этот интервал времени; $traffic(n)$ — значения загрузки канала для каждого отсчета времени.

Правое слагаемое представляет собой интегральное выражение (площадь) всплесков трафика выше среднего арифметического значения за рассматриваемый период.

Для повышения быстродействия системы проводится распараллеливание анализа генерируемых решений СИАУ ИИП перенаправлением идентичных информационных потоков на модельные объекты, подключенные к альтернативному каналу связи. Для каждого из этих решений на соответствующем МО высчитывается функция приспособленности. Расчет условия остановки алгоритма выполняется в условиях приближенных рассуждений. Соответственно, здесь задействуется блок нечеткой логики. Для оценки эффективности решения используется функция приспособленности:

$$F_i = \frac{\omega(\Delta T_0) - \omega_i(\Delta T)}{\omega(\Delta T_0)} \cdot 100\,\% = \frac{\Delta\omega_i(\Delta T)}{\omega(\Delta T_0)} \cdot 100\,\%. \quad (3)$$

Она отображает процентное соотношение снижения значения стоимостной функции $\Delta\omega_i(\Delta T)$ к значению стоимостной функции в момент до принятия решения $\omega(\Delta T_0)$. Данный показатель рассчитывается для каждой изучаемой особи (хромосомы, экземпляра решения, стратегии реагирования) на модельных объектах.

Система анализирует текущие и статистические значения функций приспособленности (F_i) всей популяции особей. Вычисляя средние значения и среднеквадратичные отклонения (СКО) для их множеств, блок нечеткой логики может принять одно из следующих решений (процентные соотношения могут динамически изменяться):

1) если текущее среднее значение лучше статистического более

Рис. 3. Блок-схема модернизированного генетического алгоритма поиска рационального решения задачи управления трафиком вычислительной сети

чем на 7 %, то выбрать лучшую особь, применить к реальной системе, не останавливать работу генетического алгоритма (ГА);

2) если число итераций < 3, то не останавливать работу ГА;

3) если число итераций ⩾ 3, все решения хуже статистических, то остановить работу ГА;

4) если число итераций ⩾ 3, текущее среднее значение лучше статистического не более чем на 7 %, и СКО ⩽ 7 %, то выбрать лучшую особь, применить к реальной системе, остановить работу ГА;

5) если число итераций ⩾ 3, значение для лучшей особи превосходит статистическое не более чем на 7 %, и СКО ⩽ 40 %, то выбрать лучшую особь, применить к реальной системе, перезапустить работу ГА;

6) если число итераций ⩾ 3, значение для лучшей особи превосходит статистическое не более чем на 7 %, и СКО > 40 %, то выбрать лучшую особь, применить к реальной системе, перезапустить работу ГА с дополнительными параметрами фильтра.

С первого по шестой пункты динамически подстраиваются параметры фильтра. В случае идентификации крайне отрицательного решения, подтвержденного статистикой к различным типам воздействий, особь исключается из допустимой выборки начальной популяции. Временные интервалы исследования решений и условия выбора корректируются блоком нечеткой логики. В случае отключения информационных потоков, направленных на модельные объекты (например, источники прекратили взаимодействие), СИАУ ИИП перенаправляет очередную порцию клиентов на данный объект. Действие выполняется для обеспечения равномерного распределения нагрузки на линии связи, предоставляемые модельным объектам.

Система содержит звено аналитики, которое ведет статистику по объектам, классам и группам объектов. Разграничиваются воздействия, риски и решения. Процентное соотношение генетических рулеток адаптивно изменяется в зависимости от статистики звена аналитики. Однако для повышения эффективности работы генетической алгоритмизации элитарное доминирование пресекается контуром нечеткой логики.

2.3. Обработка воздействий различного уровня риска

Существующие системы управления информационными потоками и процессами (в том числе маршрутизаторы, межсетевые экраны и другие единицы управляемого сетевого оборудования) в общем случае работают по схеме, показанной на рис. 4.

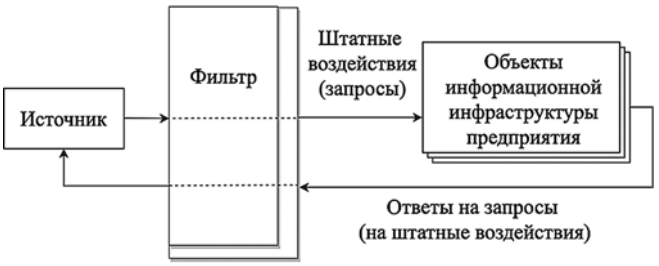


Рис. 4. Общая схема обработки внешних воздействий

Важно отметить, что легитимные запросы с применением технологий анонимизации нередко сразу блокируются существующими решениями. Не предоставляется доступ к объектам сетевой инфраструктуры источникам, обеспечивающим свою конфиденциальность.

Рассматриваемый метод интеллектуально-адаптивного управления информационной инфраструктурой предприятия вводит множественную фильтрацию, изменяет стандартную логику обработки воздействий в зависимости от их уровня риска (рис. 5, см. четвертую сторону обложки). Задействуются новые компоненты: блок прогнозирования с модельными объектами, блок фальсификации с имитированными объектами в изолированной среде, модули множественной фильтрации.

Данным комплексом выполняется идентификация и классификация внешних возмущений в зависимости от рисков негативных воздействий как на саму Систему, так и на объекты информационной инфраструктуры предприятия [9, 10]. Далее, в зависимости от типа воздействия и оценки потенциальных последствий, выявленных в результате анализа, информационные потоки перенаправляются от источников возмущений на блок прогнозирования либо блок фальсификации.

Первый блок задействуется в случае потенциальных подозрительных возмущений низкого риска. На нем проводится поиск рациональной стратегии реагирования через ранее описанный блок генетической алгоритмизации и нечеткой логики.

Второй блок подключается для обработки потенциальных подозрительных возмущений среднего и высокого риска. На нем проводится не только поиск рациональной стратегии реагирования через механизмы "информацион-

ных ловушек", но и автоматическая идентификация недостатков системы с декларированием и последующей нейтрализацией.

Результаты из блоков прогнозирования и фальсификации через обратную связь передаются системе управления с множественной фильтрацией, которая динамически перестраивается и самообучается таким образом. Затем выполняются управляющие воздействия на источники возмущений или их информационые потоки, нивелируются потенциальные риски и снижается загрузка каналов связи.

Описанный метод разрешает анонимным источникам получать доступ к ресурсам информационной инфраструктуры при отсутствии злоумышленных возмущений. Дополнительно реализована интеллектуальная поддержка принятия решений через блок прогнозирования и накопленную статистику. Если технический специалист хочет апробировать определенные действия или осуществить интеграцию новых модулей, то Система может предложить проведение итерационного исследования на модельных объектах с привнесением штатных тестовых сигналов и отслеживанием реакции. Детализированный отчет с оценкой рисков и рекомендациями будет предоставлен системному администратору.

Заключение

Был разработан новый метод интеллектуально-адаптивного управления информационной инфраструктурой предприятия, позволяющий обеспечить исправное и отказоустойчивое функционирование технических систем и объектов со снижением загрузки канала связи при разнообразных воздействиях различного уровня риска.

Автоматический поиск и применение рациональной стратегии реагирования на различные типы воздействий осуществляется путем их интеллектуальной обработки с прогнозированием реакции сервисов на изолированных модельных объектах. Для достижения данной цели применяется блок генетической алгоритмизации и нечеткой логики с самоорганизацией правил и модулей управления.

Минимизирован риск перехода информационных систем в режим недоступности, обеспечен высокий уровень безопасности информационной инфраструктуры предприятия без ущерба штатным информационным потокам, которые могли бы выполняться с применением технологий анонимизации.

Разработка, проектирование, программная реализация и исследование системы интеллектуально-адаптивного управления информационной инфраструктурой предприятия будут представлены в следующей статье.

Список литературы

1. **Кудрявцев М. Е., Калугина О. Б.** Сигнатуры систем обнаружения вторжений: основы IDS сигнатур // Актуальные проблемы современной науки, техники и образования. 2019. Т. 10, № 1. С. 80—83.

2. **Трошина С. М., Штуллер Н. В.** Система обнаружения атак // Вестник Уральского финансово-юридического института. 2016. № 4 (6). С. 109—112.

3. **Sung-Ho L., Jun-Sang P., Sung-Ho Y., Myung-Sup K.** High performance payload signature-based Internet traffic classification system // Proceedings of the 17th Asia-Pacific Network Operations and Management Symposium (APNOMS), Busan, South Korea. 2015. P. 491—494.

4. **Woo-Suk J., Jun-Sang P., Myung-Sup K., Jae-Hyun H.** Efficient payload signature structure for performance improvement of traffic identification // Proceedings of the 17th Asia-Pacific Network Operations and Management Symposium (APNOMS), Busan, South Korea. 2015. P. 180—185.

5. **He L., Cuibo Y., Xuerong G.** Analysis of traffic model and self-similarity for QQ in 3G mobile networks // Proceedings of the International Conference on Advanced Intelligence and Awareness Internet (AIAI), Shenzhen, China. 2011. P. 131—135.

6. **Faizullin R. R., Yaushev S. T., Insarov A. Y.** Modeling and Self-Similarity Analysis of Non-Poissonian Traffic Represented by Multimodal Non-Typical Pascal and Rice Distributions // Proceedings of the International Conference on Systems of Signals Generating and Processing in the Field of on Board Communications, Moscow, Russia. 2019. P. 1—4.

7. **Jieying H., Zhang J. Z.** Network traffic anomaly detection using weighted self-similarity based on EMD // Proceedings of the IEEE Southeastcon, Jacksonville, FL, USA. 2013. P. 1—5.

8. **Ye T., Dongqi H., Lishi L., Yu F.** A self-similar traffic generation model based on time // Proceedings of the 7th IEEE International Symposium on Microwave, Antenna, Propagation, and EMC Technologies (MAPE), Xi'an, China. 2017. P. 160—163.

9. **Французова Г. А., Гунько А. В., Басыня Е. А.** Самоорганизующаяся система управления трафиком вычислительной сети: метод противодействия сетевым угрозам // Программная инженерия. 2014. № 3. С. 16—20.

10. **Басыня Е. А.** Распределенная система сбора, обработки и анализа событий информационной безопасности сетевой инфраструктуры предприятия // Безопасность информационных технологий. 2018. Т. 25, № 4. С. 43—52.

E. A. Basinya, Ph. D., Professor[1], Director[2], e-mail: director@nii-ikt.ru,
[1]Novosibirsk State Technical University,
[2]Research Institute of Information and Communication Technologies, Novosibirsk, Russian Federation

Method of Intellectually-Adaptive Management of the Enterprise Information Infrastructure

The aim of this work was to develop a new method of intellectually adaptive management of the enterprise information infrastructure. It was necessary to ensure the serviceable and fault-tolerant functioning of technical systems and facilities with reducing communication channel load under various internal and external influences of a different risk level. The task was also to develop the concept of automatic search and apply a rational response strategy to various types of disturbances. A rational response strategy implied as a set of measures that would minimize the load on the communication channel in comparison with alternative solutions, without adversely affecting regular information flows and processes, as well as the level of security of the enterprise's information and communication sector. As a result, a new method of intelligent adaptive management of the enterprise information infrastructure is proposed. The choice of a rational response strategy to various types of influences is carried out by their intellectual processing with prediction of the response of services on isolated model objects. To achieve this goal, a block of genetic algorithmization and fuzzy logic is applied with self-organization of rules and control modules. The risk of information systems switching to unavailability mode is minimized, a high level of security of the enterprise information infrastructure is ensured. Complex traffic and information processes management at all levels of interaction achieve improving the reliability, fault tolerance and quality of technical systems.

Keywords: *system analysis, intelligent adaptive management, processing, network traffic, local information processes, abnormal impacts, information security, TCP/IP, threats, attacks*

DOI: 10.17587/it.26.185-191

References

1. **Kudryavcev M. E., Kalugina O. B.** Intrusion Detection Signatures: IDS Signature Basics, *Aktual'ny'e Problemy' Sovremennoj Nauki, Texniki i Obrazovaniya*, 2019, vol. 10, no. 1, pp. 80—83 (in Russian).

2. **Troshina S. M., Shtuller N. V.** Attack detection system, *Vestnik Ural'skogo finansovo-yuridicheskogo instituta*, 2016, no. 4 (6), pp. 109—112 (in Russian).

3. **Sung-Ho L., Jun-Sang P., Sung-Ho Y., Myung-Sup K.** High performance payload signature-based Internet traffic classification system, *Proceedings of the 17th Asia-Pacific Network Operations and Management Symposium (APNOMS)*, Busan, South Korea, 2015, pp. 491—494.

4. **Woo-Suk J., Jun-Sang P., Myung-Sup K., Jae-Hyun H.** Efficient payload signature structure for performance improvement of traffic identification, *Proceedings of the 17th Asia-Pacific Network Operations and Management Symposium (APNOMS)*, Busan, South Korea, 2015, pp. 180—185.

5. **He L., Cuibo Y., Xuerong G.** Analysis of traffic model and self-similarity for QQ in 3G mobile networks, *Proceedings of the International Conference on Advanced Intelligence and Awareness Internet (AIAI)*, Shenzhen, China, 2011, pp. 131—135.

6. **Faizullin R. R., Yaushev S. T., Insarov A. Y.** Modeling and Self-Similarity Analysis of Non-Poissonian Traffic Represented by Multimodal Non-Typical Pascal and Rice Distributions, *Proceedings of the International Conference on Systems of Signals Generating and Processing in the Field of on Board Communications*, Moscow, Russia, 2019, pp. 1—4.

7. **Jieying H., Zhang J. Z.** Network traffic anomaly detection using weighted self-similarity based on EMD, *Proceedings of the IEEE Southeastcon*, Jacksonville, FL, USA, 2013, pp. 1—5.

8. **Ye T., Dongqi H., Lishi L., Yu F.** A self-similar traffic generation model based on time, *Proceedings of the 7th IEEE International Symposium on Microwave, Antenna, Propagation, and EMC Technologies (MAPE)*, Xi'an, China, 2017, pp. 160—163.

9. **Franczuzova G. A., Gunko A. V., Basinya E. A.** Self-organizing computer network traffic management system: a method to counteract network threats, *Programmnaya inzheneriya*, 2014, no. 3, pp. 16—20 (in Russian).

10. **Basinya E. A.** Distributed system of collecting, processing and analysis of security information events of the enterprise network infrastructure, *Bezopasnost' informacionny'x texnologij*, 2018, vol. 25, no. 4, pp. 43—52 (in Russian).

3-я Международная научно-техническая конференция

"СОВРЕМЕННЫЕ СЕТЕВЫЕ ТЕХНОЛОГИИ"
"Modern Network Technologies, MoNeTec-2020"

27—29 октября 2020 г., г. Москва

Уважаемые Коллеги!

Приглашаем принять участие в 3-й Международной научно-технической конференции "Современные сетевые технологии" (MoNeTec-2020), 27—29 октября

http://monetec.ru

Конференция собирает представителей международного научного сообщества, исследовательских подразделений корпораций, стартапов, промышленности и бизнеса, институтов развития и органов государственной власти для обсуждения перспективных и актуальных технологий в сфере компьютерных сетей, виртуализации сетевых ресурсов и облачных вычислений, использования методов искусственного интеллекта.

На конференции планируется выступление с пленарными докладами ряда зарубежных и отечественных ученых по перспективным направлениям развития современных сетей передачи данных и их приложений. В программе конференции также предусмотрены секционные доклады по тематике конференции, стендовая сессия для студентов, индустриальная секция для представителей промышленности.

В рамках программы конференции запланировано проведение нескольких школ по сетевым технологиям и применению отечественных решений по тематике конференции для молодых ученых, студентов старших курсов и аспирантов, что будет способствовать расширению профессионального круга специалистов, способных поддерживать и развивать эти технологии и решения.

Труды Конференции будут опубликованы в библиотеке IEEE Xplore (Scopus, возможно, WoS).

Организаторами конференции выступают некоммерческое партнерство **"Центр прикладных исследований компьютерных сетей"** (НП "ЦПИ КС") и **Консорциум "Современные сетевые технологии".**

Приглашаем заинтересованные лица к участию. Подробная информация по условиям участия на сайте конференции http://www.monetec.ru

Адрес редакции:

107076, Москва, Стромынский пер., 4

Телефон редакции журнала **(499) 269-5510**
E-mail: it@novtex.ru

Технический редактор *Е. В. Конова*.
Корректор *Е. В. Комиссарова*.
Сдано в набор 09.01.2020. Подписано в печать 26.02.2020. Формат 60×88 1/8. Бумага офсетная.
Усл. печ. л. 8,86. Заказ IT320. 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**

Рисунки к статье И. В. Лобова, В. Г. Готмана

«АДАПТИВНАЯ БЕСШОВНАЯ ПОТОКОВАЯ ТРАНСЛЯЦИЯ В РЕАЛЬНОМ ВРЕМЕНИ НАД ПРОТОКОЛОМ HTTP МЕТОДОМ ОПЕРЕЖАЮЩЕЙ ЗАГРУЗКИ»

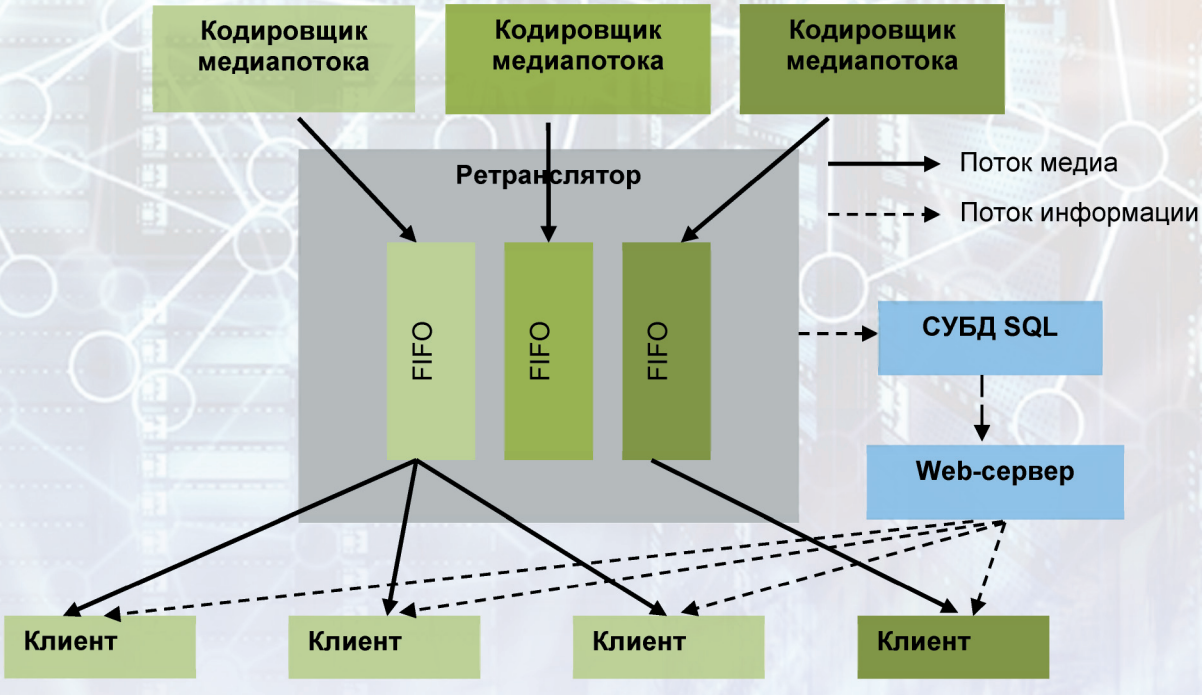


Рис. 4. Схема взаимодействия компонентов диспетчерской системы [10]

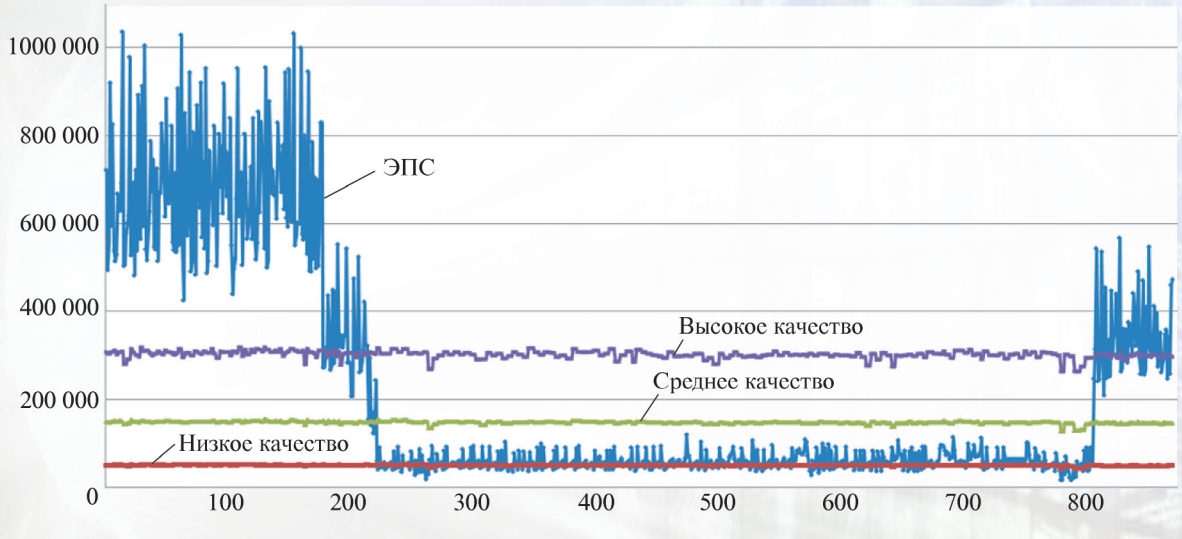


Рис. 5. Пример временного тренда эффективной пропускной способности клиента (байт/с) при тестировании системы

Рисунки к статье Е. А. Басыни

«МЕТОД ИНТЕЛЛЕКТУАЛЬНО-АДАПТИВНОГО УПРАВЛЕНИЯ ИНФОРМАЦИОННОЙ ИНФРАСТРУКТУРОЙ ПРЕДПРИЯТИЯ»

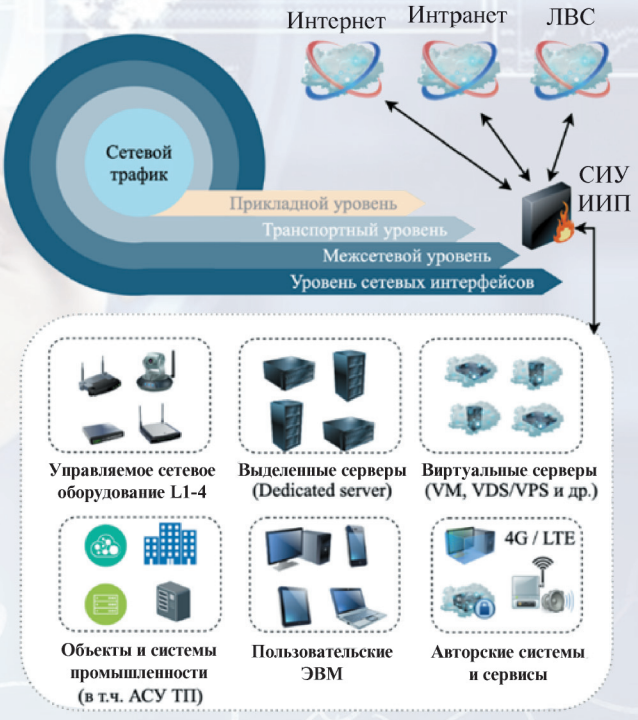


Рис. 1. Объекты управления СИУА ИИП

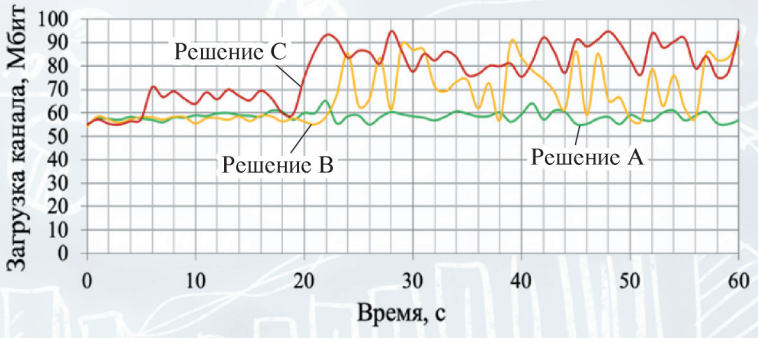


Рис. 2. Диаграмма загрузки канала связи при различных решениях по обработке трафика

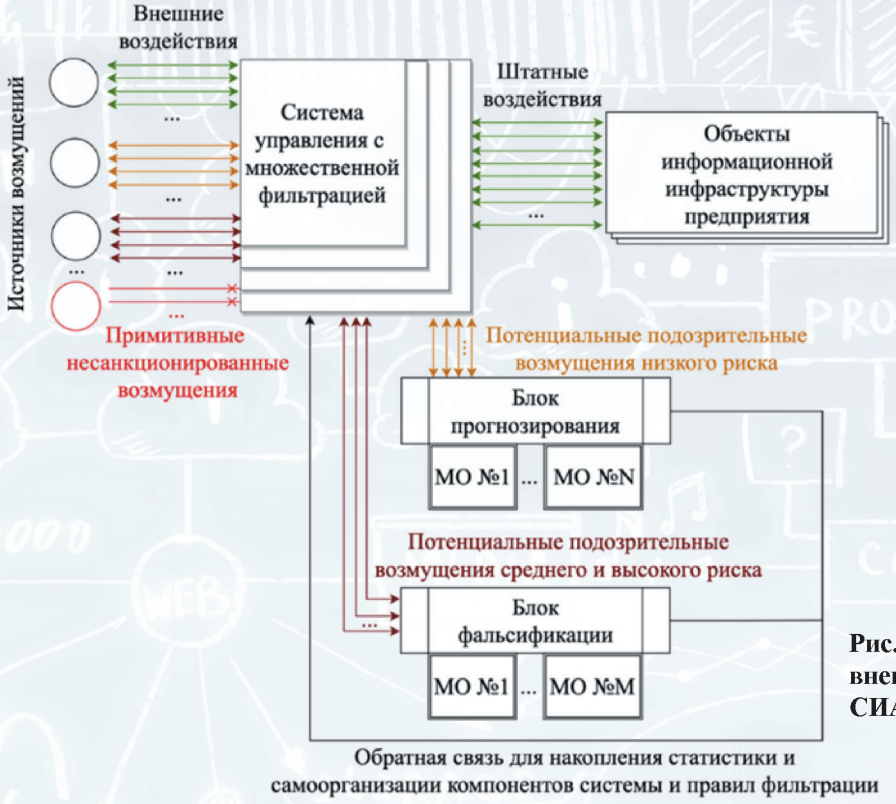


Рис. 5. Общая схема обработки внешних воздействий СИАУ ИИП

ISSN 1684-6400. Информационные технологии. 2020. Том 26. № 3. 129–192. Индекс: 72656 (Объединенный каталог «Пресса России»)