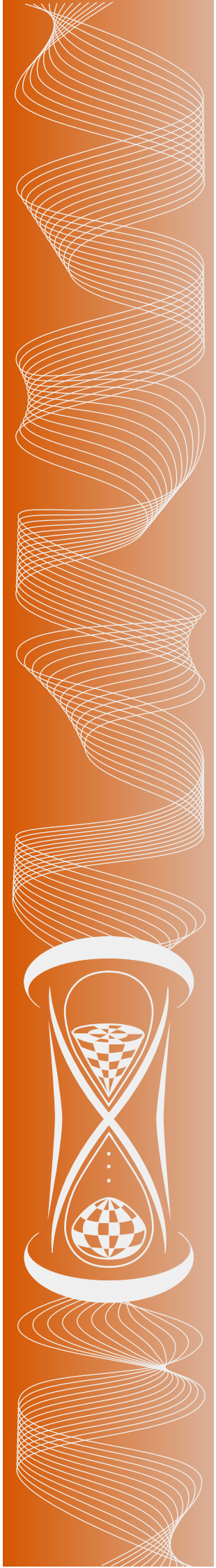
ISSN 2222-8772



Математические Структуры и Моделирование

№ 4(48)
2018

МИНИСТЕРСТВО НАУКИ
И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«ОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМ. Ф.М. ДОСТОЕВСКОГО»

МАТЕМАТИЧЕСКИЕ СТРУКТУРЫ И МОДЕЛИРОВАНИЕ

№ 4(48)

Омск
2018

Математические структуры и моделирование. — Омск : Омский государственный университет, 2018. — № 4(48). – 165 с.
ISSN 2222-8772 (print)
ISSN 2222-8799 (online)

Учредитель

Федеральное государственное бюджетное образовательное учреждение высшего образования «Омский государственный университет им. Ф. М. Достоевского».
Свидетельство о регистрации средства массовой информации ПИ № ФС77-72200 от 15 января 2018 г. выдано Роскомнадзором.

Адрес редакции, издателя и типографии

644077, Омская обл., г. Омск, пр-т Мира, д. 55а.

Дата выхода в свет: 29.12.2018.

Тираж 100 экз.

Свободная цена.

МАТЕМАТИЧЕСКИЕ СТРУКТУРЫ и МОДЕЛИРОВАНИЕ

Журнал основан в 1998 году. В журнале публикуются статьи, в которых излагаются результаты исследований по фундаментальной и прикладной математике, теоретической физике, компьютерным наукам, философии и истории математики и информатики, а также размышления, касающиеся окружающей нас природы и общества. Объекты исследования должны быть представлены в форме некоторых математических структур и моделей.

Все статьи журнала проходят обязательное рецензирование. Рефераты статей журнала опубликованы в «Реферативном журнале» и «Mathematical Reviews» (США). Журнал индексируется в РИНЦ (elibrary.ru) и «Zentralblat für Mathematik» (Германия). Журнал входит в Перечень рецензируемых научных изданий ВАК РФ, в которых должны быть опубликованы основные результаты диссертаций на соискание учёных степеней (Приказ Минобрнауки России от 25 июля 2014 г. № 793).

Электронная версия журнала представлена в сети:

http://msm.univer.omsk.su
http://msm.omsu.ru

Подписной индекс по каталогу «Пресса России»: 94082

Электронная почта главного редактора:

guts@omsu.ru

Электронная почта выпускающего редактора:

lavrov@omsu.ru

СОДЕРЖАНИЕ

Фундаментальная математика и физика

Прикладная математика и моделирование

Компьютерные науки

Информационная безопасность

Продолжение на следующей странице

Наши публикации

Математические структуры и моделирование 2018. № 4(48). С. 5–13

УДК 519.214
DOI: 10.25513/2222-8772.2018.4.5-13

СХОДИМОСТЬ РАСПРЕДЕЛЕНИЙ КАЛИБРОВОЧНЫХ ФУНКЦИЙ ОТ ЗАВИСИМЫХ ВЕЛИЧИН К MAX-УСТОЙЧИВЫМ ЗАКОНАМ

А.Г. Гринь
профессор, д.ф.-м.н., e-mail: griniran@gmail.com

Омский государственный университет им. Ф.М. Достоевского, Омск, Россия

Аннотация. Получены необходимые и достаточные условия для сходимости распределений симметрических калибровочных функций от зависимых случайных величин к max-устойчивым законам. Эти условия включают в себя и так называемые минимальные условия слабой зависимости.

Ключевые слова: калибровочные функции от случайных величин, max-устойчивые распределения, минимальные условия слабой зависимости.

Пусть $\{\xi_n\}$ — последовательность независимых одинаково распределённых величин, $X_n = \max\limits_{1\leqslant k\leqslant n} \xi_n$, $F_n(x) = \mathbf{P}\{X_n < x\}$, $F_1(x) < 1$, $x > 0$, а $F_n \Rightarrow F$ означает, что $\{F_n\}$ слабо сходится к F. Следуя [1], назовём $\{a_n,\ n = 1, 2, ...\}$ правильно меняющейся последовательностью порядка ρ, если $a_{[x]}$, $x > 0$ является правильно меняющейся функцией порядка ρ, где $[x]$ — целая часть x.

Для того чтобы при некотором выборе нормирующих констант a_n имело место соотношение $F_n(xa_n) \Rightarrow F_\xi(x)$, $n \to \infty$, где ξ — невырожденная случайная величина, необходимо и достаточно, чтобы $\mathbf{P}\{\xi_1 \geqslant x\}$ являлась правильно меняющейся функцией порядка $-\rho$, $\rho > 0$. При этом предельное распределение (их называют max-устойчивыми) имеет вид $F_\xi(x) = G_\rho(x) = \exp\{-cx^{-\rho}\}$, $x > 0$, $c > 0$, а нормирующие постоянные a_n можно найти из соотношения

$$a_n = \sup\{x : n\mathbf{P}\{|\xi_1| \geqslant x\} \geqslant 1\}. \tag{1}$$

Такая последовательность $\{a_n\}$ существует и является правильно меняющейся порядка $1/\rho$ [1, с. 29] и

$$n\mathbf{P}\{|\xi_1| \geqslant xa_n\} \to x^{-\rho},\ n \to \infty \tag{2}$$

[2, с. 319].

Символ $n + m \to \infty$ в каком-либо соотношении будет означать, что указанное соотношение выполняется при $n \to \infty$ и при любой последовательности натуральных чисел $m = m(n)$. В [3] получен следующий результат.

Теорема 1. *Пусть $\{\xi_n\}$, $n = 1, 2, ...\}$ — стационарная последовательность, у которой $\mathbf{P}\{\xi_1 \geqslant x\}$ является правильно меняющейся функцией*

порядка $-\rho,\ \rho > 0$ *и пусть* $X_n = \max\limits_{1\leqslant k\leqslant n} \xi_n$, *а последовательность* $\{a_n\}$ *определяется из соотношения* $n\mathbf{P}\{\xi_1 \geqslant a_n\} \to 1, n \to \infty$. *Для того чтобы* $F_n(xa_n) \to G_\rho(x),\ \ n \to \infty,\ x > 0$, *необходимо и достаточно, чтобы выполнялись следующие утверждения:*

а)

$$F_{n+m}(xa_{n+m}) - F_n(xa_{n+m})F_m(xa_{n+m}) \to 0,\quad n+m \to \infty; \tag{R_1}$$

б) при любом $x > 0$ *и при любой достаточно медленно растущей последовательности* $k = k(n) \to \infty,\ n \to \infty$

$$\mathbf{P}\{X_n > xa_{kn}\} \sim n\mathbf{P}\{\xi_1 > xa_{kn}\},\ n \to \infty. \tag{R_2}$$

Замечание 1. Теорему 1 можно интерпретировать так: условия (R_1) и (R_2) являются минимальными условиями слабой зависимости, при которых выполняются предельные теоремы для максимумов с той же нормировкой, что и в предельных теоремах для независимых величин.

В настоящей работе результат теоремы 1 обобщается на случай, когда X_n является функцией специального вида (так называемой калибровочной функцией) от величин $\xi_1, ..., \xi_n$ и приводится «общеупотребительное» условие слабой зависимости, обеспечивающее выполнение аналога условия (R_2).

Пусть при каждом $n \in \mathbb{N}$ определено отображение $f : \mathbb{R}^n \to \mathbb{R}$, удовлетворяющее следующим условиям:

f1. $f(\mathbf{x}) > 0,\ \mathbf{x} \neq 0$;

f2. $f(\gamma\mathbf{x}) = |\gamma| f(\mathbf{x}),\ \gamma \in \mathbb{R}$;

f3. $f(\mathbf{x}+\mathbf{y}) \leqslant f(\mathbf{x}) + f(\mathbf{y})$;

f4. $f(x_1, ..., x_n) = f(\varepsilon_1 x_{i_1}, ..., \varepsilon_n x_{i_n})$ где $\varepsilon_i = \pm 1$, а $(i_1, ..., i_n)$ – перестановка множества $(1, ..., n)$.

Функция f (на самом деле последовательность функций, но чтобы не загромождать рассуждений, мы не будем подчёркивать зависимость f от n какими-либо индексами и называть f последовательностью) называется *симметрической калибровочной функцией* (см., например, [4, с.107]).

Будем также предполагать, что

f5. $f(x_1, x_2, ..., x_{n-1}, 0) = f(x_1, x_2, ..., x_{n-1})$.

В [5] можно посмотреть примеры функций, удовлетворяющих свойствам f1-f5. В силу f3 с $\mathbf{x} = (x_1, x_2, ..., x_k, 0..., 0),\ \mathbf{y} = (0, ..., 0, x_{k+1}, ..., x_n)$ при любом $k < n$

$$-f(x_{k+1}, x_2, ..., x_n) \leqslant f(x_1, x_2, ..., x_n) - f(x_1, x_2, ..., x_k) \leqslant f(x_{k+1}, x_2, ..., x_n),$$

так что для любого $1 \leqslant k \leqslant n$

$$|f(x_1, x_2, ..., x_n) - f(x_1, x_2, ..., x_k)| \leqslant f(x_{k+1}, , ..., x_n). \tag{3}$$

Пусть $\{\xi_n\}$ — стационарная в узком смысле последовательность и пусть $X_n = f(\xi_1, ..., \xi_n),\ \ F_n(x) = \mathbf{P}\{X_n < x\}$.

Теорема 2. *Пусть* $\{\xi_n\}$, $n = 1, 2, ...\}$ — *стационарная последовательность, у которой* $\mathbf{P}\{|\xi_1| \geqslant x\}$ *является правильно меняющейся функцией порядка* $-\rho$, $\rho > 0$, $X_n = f(\xi_1, ..., \xi_n)$, *а последовательность* $\{a_n\}$ *определяется соотношением (1). Для того чтобы* $F_n(xa_n) \to G_\rho(x) = \exp\{-cx^{-\rho}\}$, $c = f^\rho(1)$, $x > 0$ $n \to \infty$: *необходимо и достаточно, чтобы выполнялись следующие утверждения*

а)

$$F_{n+m}(xa_{n+m}) - F_n(xa_{n+m})F_m(xa_{n+m}) \to 0, \quad n + m \to \infty; \tag{$\mathrm{R}_1(\mathrm{f})$}$$

б) при любом $x > 0$ *и при любой достаточно медленно растущей последовательности* $k = k(n) \to \infty$, $n \to \infty$

$$\mathbf{P}\{X_n > xka_n\} \sim n\mathbf{P}\{X_1 > xka_n\}, \ n \to \infty. \tag{$\mathrm{R}_2(\mathrm{f})$}$$

Если же для некоторой последовательности $\{\xi_n\}$ $F_n(xa_n) \to G_\rho(x)$, $x > 0$ *и выполняются условия* $(\mathrm{R}_1(\mathrm{f}))$ *и* $(\mathrm{R}_2(\mathrm{f}))$, *то* $\mathbf{P}\{|\xi_1| \geqslant x\}$ *является правильно меняющейся функцией порядка* $-\rho$.

Замечание 2. Условие R_2 интерпретировалось в [3] как одно из минимальных условий слабой зависимости, при которых распределения величин $X_n = \max\limits_{1 \leqslant k \leqslant n} \xi_n$ сходятся к max-устойчивым законам. В настоящей работе условие $\mathrm{R}_2(\mathrm{f})$ является не только условием слабой зависимости, но и накладывает значительные ограничения на вид функции f, заключающиеся по сути в том, что распределения функций $f(\xi_1, ..., \xi_n)$ слабо эквивалентны распределениям максимумов некоторых независимых случайных величин.

Лемма 1. . *Последовательность* $\{a_n^\rho\}$ *является правильно меняющейся последовательностью порядка 1 (а* a_n *— правильно меняющейся последовательностью порядка* $1/\rho$, $\rho > 0$) *тогда и только тогда, когда*

$$a_{n+m}^\rho \sim a_n^\rho + a_m^\rho, \quad n + m \to \infty.$$

Доказательство, по существу, повторяет доказательство леммы 1 в [3].

Доказательство теоремы 2.

Необходимость. Пусть $F_n(xa_n) \Rightarrow G_\rho(x)$, $n \to \infty$. Функция $G_\rho(x)$ непрерывна при $x > 0$, поэтому слабая сходимость равносильна поточечной:

$$F_n(xa_n) \to G_\rho(x), \ x > 0. \tag{4}$$

Пусть $m = m(n)$ — произвольная последовательность натуральных чисел. Обозначим

$$\Delta(n) = |F_{n+m}(xa_{n+m}) - F_n(xa_{n+m})F_m(xa_{n+m})|.$$

Поскольку a_n^ρ — правильно меняющаяся последовательность порядка 1, то в силу леммы 1

$$a_{n+m}^\rho \sim a_n^\rho + a_m^\rho, \ n \to \infty,$$

так что для любой последовательности натуральных чисел $\{n_1\}$ существуют $0 \leqslant a \leqslant 1$ и подпоследовательность $\{n_2\} \subseteq \{n_1\}$ такие, что

$$a_{n_2+m_2}^{-\rho} a_{n_2}^{\rho} \to a, \quad a_{n_2+m_2}^{-\rho} a_{m_2}^{\rho} \to 1-a, \quad n \to \infty,$$

где $m_2 = m(n_2)$. Пусть сначала $0 < a < 1$. С помощью (4) получаем

$$\Delta(n_2) = \left| F_{n_2+m_2}(x a_{n_2+m_2}) - F_{n_2}\left(x a_{n_2} \frac{a_{n_2+m_2}}{a_{n_2}}\right) F_{m_2}\left(x a_{m_2} \frac{a_{n_2+m_2}}{a_{m_2}}\right)\right| \to$$

$$\to \left| G_\rho(x) - G_\rho\left(x a^{-\frac{1}{\rho}}\right) G_\rho\left(x(1-a)^{-\frac{1}{\rho}}\right)\right| =$$

$$= \left|\exp\left\{-cx^{-\rho}\right\} - \exp\left\{-acx^{-\rho}\right\} \exp\left\{-(1-a)cx^{-\rho}\right\}\right| = 0. \tag{5}$$

Если же $a = 0$ $(a = 1)$, то при $n \to \infty$ $a_{n_2+m_2}^{-1} X_{n_2} \to 0$ $(a_{n_2+m_2}^{-1} X_{m_2} \to 0)$ по вероятности, следовательно, при $x > 0$ $F_{n_2}(x a_{n_2+m_2}) \to 1$ $(F_{m_2}(x a_{n_2+m_2}) \to 1)$, и с помощью (4) легко выводится, что

$$|F_{n_2+m_2}(x a_{n_2+m_2}) - F_{m_2}(x a_{n_2+m_2})| \to 0 \ (|F_{n_2+m_2}(x a_{n_2+m_2}) - F_{n_2}(x a_{n_2+m_2})| \to 0),$$

то есть $\Delta(n_2) \to 0,\ n \to \infty$. Вместе с (5) это означает, что из любой последовательности $\{\Delta(n_1)\}$ можно выделить сходящуюся к нулю подпоследовательность. Следовательно, что $\Delta(n) \to 0,\ n \to \infty$, и мы показали, что выполнено условие $(\mathrm{R}_1(\mathrm{f}))$.

Докажем $(\mathrm{R}_2(\mathrm{f}))$. В силу условия f_2 $X_1 = c^{1/\rho}|\xi_1|$. Так как $\{a_n^\rho\}$ — правильно меняющаяся последовательность порядка 1, то $a_{kn}^\rho \sim k a_n^\rho$, и если $k = k(n) \to \infty$ растёт достаточно медленно, то в силу (2)

$$n\mathbf{P}\{X_1 > x a_{kn}\} = n\mathbf{P}\{c^{1/\rho}|\xi_1| > x a_{kn}\} \sim \frac{c}{k x^\rho},\ n \to \infty. \tag{6}$$

По предположению теоремы

$$\mathbf{P}\{X_n > x a_{kn}\} = 1 - F_n(x a_{kn}) \sim 1 - \exp\left\{-\frac{c}{k x^\rho}\right\} \sim \frac{c}{k x^\rho},$$

что вместе с (6) даёт нам условие $\mathrm{R}_2(\mathrm{f})$.

Достаточность.

Пусть выполнены условия $(\mathrm{R}_1(\mathrm{f}))$ и $(\mathrm{R}_2(\mathrm{f}))$, $k = k(n) \to \infty$, $n = km + r$, $0 \leqslant r < m$. С помощью условия $(\mathrm{R}_1(\mathrm{f}))$ при k, растущем достаточно медленно, получаем

$$F_n(x a_n) \sim F_m^k(x a_n) F_r(x a_n),\ n \to \infty. \tag{7}$$

Правильно меняющаяся функция положительного порядка a_n эквивалентна неубывающей функции [1, с.26], и мы в дальнейшем будем считать её таковой. Если $r = r(n) \to \infty$, то из условия $(\mathrm{R}_1(\mathrm{f}))$ и (6) следует

$$1 - F_r(x a_n) \leqslant \mathbf{P}\{X_r \geqslant a_{kr}\} \sim r\mathbf{P}\{X_1 \geqslant a_{kr}\} \sim \frac{c}{k} \to 0,\ n \to \infty. \tag{8}$$

Если же $r = r(n)$ — ограниченная последовательность, то $r\mathbf{P}\{X_1 \geqslant a_{kr}\} \to 0$ просто потому, что $a_{kr} \to \infty,\ n \to \infty$. Вместе с (7) и (8) это означает, что

$$F_n(xa_n) \sim F_m^k(xa_n) = (1 - \mathbf{P}\{X_m > xa_n\})^k \ \ n \to \infty. \tag{9}$$

Из правильного изменения последовательности $\{a_n\}$ легко выводится, что $a_n \sim a_{km}$, и в силу условия $(\mathrm{R}_2(\mathrm{f}))$

$$\mathbf{P}\{X_m > xa_n\} \sim \mathbf{P}\{X_m > xa_{km}\} \sim m\mathbf{P}\{X_1 > xa_{km}\} \sim \frac{c}{kx^\rho}. \tag{10}$$

Из (9) и (10) выводим

$$F_n(xa_n) \sim \left(1 - \frac{c}{kx^\rho}(1 + o_n(1))\right)^k \to \exp\left\{-cx^{-\rho}\right\},\ n \to \infty.$$

Пусть теперь $F_n(xa_n) \to G_\rho(x) = \exp\{-cx^{-\rho}\},\ c = f^\rho(1),\ x > 0$ и $k = k(n) \to \infty,\ n \to \infty$. Тогда если $k(n)$ растет достаточно медленно, то

$$\mathbf{P}\{X_n > xka_n\} \sim \frac{c}{k^\rho x^\rho},$$

а в силу $(\mathrm{R}_2(\mathrm{f}))$

$$\mathbf{P}\{X_n > xka_n\} \sim n\mathbf{P}\{X_1 > xka_n\} = n\mathbf{P}\{c^{1/\rho}|\xi_1| > xka_n\} \sim \frac{c}{k^\rho x^\rho},\ n \to \infty.$$

Отсюда следует, что $\mathbf{P}\{\xi_1 \geqslant x\}$ является правильно меняющейся функцией порядка $-\rho$ [2, с.318]. Теорема доказана.

Приведём пример условия слабой зависимости, обеспечивающего выполнение условия $(\mathrm{R}_2(\mathrm{f}))$.

Пусть $\mathcal{F}_{\leqslant n}$ и $\mathcal{F}_{\geqslant n}$ – σ-алгебры, порождённые семействами $\{\xi_i : i \leqslant n\}$ и $\{\xi_i : i \geqslant n\}$. Если для некоторой функции $\lambda(x) > 0$ такой, что $\lambda(x) \downarrow 0,\ x \to 0$

$$\sup\left\{\frac{\mathbf{P}(AB)}{\mathbf{P}(A)\lambda(\mathbf{P}(B))} : A \in \mathcal{F}_{\leqslant 0}, B \in \mathcal{F}_{\geqslant 1} \text{ или } A \in \mathcal{F}_{\geqslant 1}, B \in \mathcal{F}_{\leqslant 0}\right\} < 1,$$

то говорят, что последовательность $\{\xi_n\}$ удовлетворяет *условию λ-перемешивания* (см.[6]).

Пусть $\{c_n\}$ — последовательность положительных чисел. Обозначим

$$X_{k,l} = f(\xi_k, ..., \xi_l), k < l,\ \ \overline{X}_n = \max_{1 \leqslant k \leqslant n} X_k,\ \ \delta_n = \lambda\left(2 \max_{1 \leqslant k \leqslant n} \mathbf{P}\{X_k \geqslant \varepsilon c_n\}\right).$$

Лемма 2. *Пусть $\varepsilon > 0,\ x > 0$ и $m \leqslant n$, а функция f удовлетворяет условиям $f_1 - f_5$. Если последовательность $\{c_n\}$ такова, что $\delta_n < 1$, то*

$$\mathbf{P}\{\overline{X}_{m-1} \geqslant (x + \varepsilon)c_n\} \leqslant (1 - \delta_n)^{-1}\mathbf{P}\{X_m \geqslant xc_n\}.$$

Доказательство. Пусть $E_k = \{\overline{X}_{k-1} < (x+\varepsilon)c_n \leqslant X_k\},\ k = 1, ..., m,\ \varepsilon > 0.$ Тогда $E_iE_j = \varnothing,\ i \neq j,\ \bigcup\limits_{k=1}^{m-1} E_k = \{\overline{X}_{m-1} \geqslant (x+\varepsilon)c_n\}$, а в силу (3)

$$\{X_k \geqslant (x+\varepsilon)c_n,\ X_{k+1,m} < \varepsilon c_n\} \subseteq \{X_m \geqslant xc_n\},$$

то есть

$$\{X_m < xc_n\} \subseteq \{X_k < (x+\varepsilon)c_n\} \cup \{X_{k+1,m} \geqslant \varepsilon c_n\},$$

откуда

$$\{X_m < xc_n,\ E_k\} \subseteq \{X_{k+1,m} \geqslant \varepsilon c_n,\ E_k\},\ k = 1, ..., m-1. \tag{11}$$

С помощью (11) и условия λ-перемешивания получаем

$$\mathbf{P}\{\overline{X}_{m-1} \geqslant (x+\varepsilon)c_n\} \leqslant \mathbf{P}\{X_m \geqslant xc_n\} + \sum_{k=1}^{m-1} \mathbf{P}\{X_m < xc_n,\ E_k\} \leqslant$$

$$\leqslant \mathbf{P}\{X_m \geqslant xc_n\} + \sum_{k=1}^{m-1} \mathbf{P}\{X_{k+1,m} \geqslant \varepsilon c_n,\ E_k\} \leqslant$$

$$\leqslant \mathbf{P}\{X_m \geqslant xc_n\} + \lambda \left(\max_{1 \leqslant k \leqslant n} \mathbf{P}\{X_k \geqslant \varepsilon c_n\} \right) \sum_{k=1}^{m-1} \mathbf{P}\{E_k\} \leqslant$$

$$\leqslant \mathbf{P}\{X_m \geqslant xc_n\} + \delta_n \cdot \mathbf{P}\{\overline{X}_{m-1} \geqslant (x+\varepsilon)c_n\},$$

откуда следует утверждение леммы. ■

Лемма 3. *Если функция f удовлетворяет условиям $f_1 - f_5$, последовательность $\{c_n\}$ такова, что $\delta_n < 1/2$, то при любом $x > 0$ и $0 < \varepsilon < x$*

$$\mathbf{P}\{X_n \geqslant xc_n\} \geqslant n\mathbf{P}\{X_1 \geqslant (x+3\varepsilon)c_n\}(1 - 3\delta_n).$$

Доказательство. Пусть $A_n = \{X_{n-1} < 2\varepsilon,\ f(\xi_n) \geqslant (x+3\varepsilon)c_n\}$

$$A_k = \{X_{k-1} < 2\varepsilon c_n,\ f(\xi_k) \geqslant (x+3\varepsilon)c_n,\ X_{k+1,n} < \varepsilon c_n\},\ 1 \leqslant k \leqslant n-1.$$

В силу (3)

$$|X_n - f(\xi_k)| \leqslant X_{k-1} + X_{k+1,n}, \tag{12}$$

так что

$$\mathbf{P}\{X_n \geqslant xc_n\} \geqslant \mathbf{P}\left\{\bigcup_{k=1}^{n} A_k\right\} = \sum_{k=1}^{n} \mathbf{P}\{\overline{A}_1 \cdot ... \cdot \overline{A}_{k-1}A_k\} =$$

$$= \sum_{k=1}^{n} \mathbf{P}\{A_k\} - \sum_{k=1}^{n} \mathbf{P}\left\{A_k \cdot \bigcup_{j=1}^{k-1} A_j\right\}. \tag{13}$$

При $1 \leqslant k \leqslant n-1$ получаем

$$\mathbf{P}\{A_k\} = \mathbf{P}\{f(\xi_k) \geqslant (x+3\varepsilon)c_n\} -$$

$$-\mathbf{P}\left\{f\left(\xi_k\right) \geqslant (x+3\varepsilon)c_n,\ \left(\{X_{k-1} \geqslant 2\varepsilon c_n\} \cup \{X_{k+1,n} \geqslant \varepsilon c_n\}\right)\right\} \geqslant$$

$$\geqslant \mathbf{P}\left\{f\left(\xi_k\right) \geqslant (x+3\varepsilon)c_n\right\}\left(1-\lambda\left(\mathbf{P}\{X_{k+1,n} \geqslant \varepsilon c_n\}\right)-\lambda\left(\mathbf{P}\{X_{k-1} \geqslant 2\varepsilon c_n\}\right) \geqslant\right.$$

$$\geqslant \mathbf{P}\left\{f\left(\xi_k\right) \geqslant (x+3\varepsilon)c_n\right\}(1-2\delta_n). \tag{14}$$

$\mathbf{P}\{A_n\}$ оценивается аналогично. Далее

$$\{X_{j-1} < 2\varepsilon c_n,\ f\left(\xi_j\right) \geqslant (x+3\varepsilon)c_n\} \subseteq \{X_{j-1} < 2\varepsilon c_n,\ X_j \geqslant (x+\varepsilon)c_n\},$$

так что если $\varepsilon < x$, то $\mathbf{P}\left\{A_k \cdot \bigcup\limits_{j=1}^{k-1} A_j\right\} \leqslant$

$$\leqslant \mathbf{P}\left\{f\left(\xi_k\right) \geqslant (x+3\varepsilon)c_n,\ \bigcup_{j=1}^{k-1}\left(X_{j-1} < 2\varepsilon c_n, f\left(\xi_j\right) \geqslant (x+3\varepsilon)c_n\right)\right\} \leqslant$$

$$\leqslant \mathbf{P}\left\{f\left(\xi_k\right) \geqslant (x+3\varepsilon)c_n, \overline{X}_{k-1} \geqslant 2\varepsilon c_n\right\})$$

$$\leqslant \mathbf{P}\left\{f\left(\xi_k\right) \geqslant (x+3\varepsilon)c_n\right\}\lambda\left(\mathbf{P}\{\overline{X}_{k-1} \geqslant 2\varepsilon c_n\}\right). \tag{15}$$

и тогда из (13), (14) и (15) и леммы 2 с $\delta_n < 1/2$ следует

$$\mathbf{P}\{X_n \geqslant xc_n\} \geqslant \sum_{k=1}^{n} \mathbf{P}\left\{f\left(\xi_k\right) \geqslant (x+\varepsilon)c_n\right\}(1-3\delta_n).$$

Лемма доказана.

■

Следующее предложение – это модификация леммы 3.1 из [7].

Лемма 4. *Пусть функция f удовлетворяет условиям $f_1 - f_5$, $\varepsilon > 0$ и последовательность $\{c_n\}$ такова, что $\delta_n < 1$. Тогда*

$$\mathbf{P}\{X_n \geqslant (x+3\varepsilon)c_n\} \leqslant \delta_n(1-\delta_n)^{-1}\mathbf{P}\{X_n \geqslant \varepsilon c_n\} + n\mathbf{P}\left\{X_1 \geqslant xc_n\right\}.$$

Доказательство. Пусть $E_k = \{\overline{X}_{k-1} < 2\varepsilon c_n \leqslant X_k\}, k = 1, ..., n$. Тогда $E_iE_j = \varnothing,\ i \neq j,\ \bigcup\limits_{k=1}^{n-1} E_k = \{\overline{X}_{n-1} \geqslant 2\varepsilon c_n\}$. В силу (3) при $1 \leqslant k \leqslant n-1$

$$\left\{X_{k+1,n} < \varepsilon c_n,\ E_k,\ \max_{1\leqslant k\leqslant n} f\left(\xi_k\right) < xc_n\right\} \subseteq$$

$$\subseteq \left\{X_n < (x+3\varepsilon)c_n,\ E_k,\ \max_{1\leqslant k\leqslant n} f\left(\xi_k\right) < xc_n\right\},$$

откуда

$$\left\{X_n \geqslant (x+3\varepsilon)c_n,\ E_k,\ \max_{1\leqslant k\leqslant n} f\left(\xi_k\right) < xc_n\right\} \subseteq \{E_k,\ X_{k+1,n} \geqslant \varepsilon c_n\}. \tag{16}$$

Аналогично выводится

$$\left\{X_n \geqslant (x+3\varepsilon)c_n,\ \max\limits_{1\leqslant k\leqslant n} f(\xi_k) < xc_n\right\} \subseteq \left\{\overline{X}_{n-1} \geqslant 2\varepsilon c_n,\ \max\limits_{1\leqslant k\leqslant n} f(\xi_k) < xc_n\right\}.$$

Отсюда

$$\left\{X_n \geqslant (x+3\varepsilon)c_n,\ \max\limits_{1\leqslant kn} f(\xi_k) < xc_n\right\} =$$

$$= \left\{X_n \geqslant (x+3\varepsilon)c_n,\ \overline{X}_{n-1} \geqslant 2\varepsilon c_n,\ \max\limits_{1\leqslant k\leqslant n} f(\xi_k) < xc_n\right\}. \tag{17}$$

С помощью (16) и (17) получаем $\mathbf{P}\{X_n \geqslant (x+3\varepsilon)c_n\} \leqslant$

$$\leqslant \mathbf{P}\left\{X_n \geqslant (x+3\varepsilon)c_n, \max\limits_{1\leqslant k\leqslant n} f(\xi_k) < xc_n\right\} + \mathbf{P}\left\{\max\limits_{1\leqslant k\leqslant n} f(\xi_k) \geqslant xc_n\right\} =$$

$$= \mathbf{P}\left\{X_n \geqslant (x+3\varepsilon)c_n, \overline{X}_{n-1} \geqslant 2\varepsilon c_n, \max\limits_{1\leqslant k\leqslant n} f(\xi_k) < xc_n\right\} + \mathbf{P}\left\{\max\limits_{1\leqslant k\leqslant n} f(\xi_k) \geqslant xc_n\right\} =$$

$$= \sum_{k=1}^{n-1} \mathbf{P}\left\{X_n \geqslant (x+3\varepsilon)c_n, E_k, \max\limits_{1\leqslant k\leqslant n} f(\xi_k) < xc_n\right\} + \mathbf{P}\left\{\max\limits_{1\leqslant k\leqslant n} f(\xi_k) \geqslant xc_n\right\}. \tag{18}$$

Из соотношения (3) следует

$$X_{k+1,n} \geqslant X_n - f(\xi_k) - X_{k-1},$$

и из (18) выводим

$$\mathbf{P}\{X_n \geqslant (x+3\varepsilon)c_n\} \leqslant \sum_{k=1}^{n-1} \mathbf{P}\{X_{k+1,n} \geqslant \varepsilon c_n, E_k\} + \mathbf{P}\left\{\max\limits_{1\leqslant k\leqslant n} f(\xi_k) \geqslant xc_n\right\} \leqslant$$

$$\leqslant n\mathbf{P}\{X_1 \geqslant xc_n\} + \lambda\left(\max\limits_{1\leqslant k\leqslant n} \mathbf{P}\{X_k \geqslant \varepsilon c_n\}\right) \sum_{k=1}^{n-1} \mathbf{P}\{E_k\} =$$

$$= \delta_n \mathbf{P}\{\overline{X}_{n-1} \geqslant 2\varepsilon c_n\} + n\mathbf{P}\{X_1 \geqslant xc_n\}.$$

Из этого соотношения с помощью Леммы 1 выводим утверждение леммы. ■

Замечание 3. Из лемм 2 и 3 вытекает следующее утверждение: если последовательность положительных чисел $\{c_n\}$ такова, что при любом $\varepsilon > 0$

$$\delta_n = \lambda\left(\max\limits_{1\leqslant k\leqslant n} \mathbf{P}\{X_k \geqslant \varepsilon c_n\}\right) \to 0,\ n \to \infty$$

и при любых $x > 0,\ \varepsilon > 0$ выполняется одно из следующих предположений:

$$\mathbf{P}\{X_n \geqslant (x+3\varepsilon)c_n\} = O\left(\mathbf{P}\{X_n \geqslant \varepsilon c_n\}\right)\ n \to \infty, \tag{19}$$

$$\delta_n \mathbf{P}\{X_n \geqslant \varepsilon c_n\} = o\left(n\mathbf{P}\{X_1 \geqslant xc_n\}\right),\ n \to \infty, \tag{20}$$

то при $n \to \infty$

$$\mathbf{P}\{X_n \geqslant xc_n\} \sim n\mathbf{P}\{X_1 \geqslant xc_n\}, \tag{21}$$

то есть, имеет место $(\mathrm{R}_2(\mathrm{f}))$.

Если для некоторой последовательности $\{\xi_n\}$ выполняется условие λ-перемешивания, $F_n(xa_n) \to G_\rho(x) = \exp\{-cx^{-\rho}\}$, $x > 0$, а $k = k(n) \to \infty$ растёт достаточно медленно, то

$$\frac{\mathbf{P}\{X_n \geqslant (x+3\varepsilon)ka_n\}}{\mathbf{P}\{X_n \geqslant \varepsilon ka_n\}} \sim \frac{1-\exp\{-ck^{-\rho}(x+3\varepsilon)^{-\rho}\}}{1-\exp\{-c(\varepsilon k)^{-\rho}\}} \to \left(\frac{\varepsilon}{x+3\varepsilon}\right)^{\rho}, \ n \to \infty.$$

Это означает, что выполняется (19) и, следовательно, (21), то есть $(\mathrm{R}_2(\mathrm{f}))$.

Литература

1. Сенета Е. Правильно меняющиеся функции. М. : Наука, 1985.
2. Феллер В. Введение в теорию вероятностей и её приложения. Т. 2. М. : Мир, 1984.
3. Гринь А.Г. Минимальные условия слабой зависимости в предельных теоремах для максимумов // Математические структуры и моделирование. 2006. Вып. 16. С. 21–25.
4. Маршалл А., Олкин И. Неравенства: теория мажоризации и её приложения. М. : Мир, 1983. 574 с.
5. Гринь А.Г. О предельных теоремах для функций от независимых случайных величин // Математические структуры и моделирование. 2016. № 2(38). С. 5–15.
6. Гринь А.Г. Области притяжения для последовательностей с перемешиванием // Сибирский математический журнал. 1990. Т. 31, № 1. С. 53–63.
7. Peligrad M. An invariance principle for φ– mixing sequences // Ann. Probab. 1985. V. 13, No. 4. P. 1304–1313.

THE CONVERGENCE OF THE DISTRIBUTIONS OF THE CALIBRATION FUNCTIONS FROM THE DEPENDENT VARIABLES TO MAX-STABLE LAWS

A.G. Grin

Dr.Sc. (Phys.-Math.), Professor, e-mail: griniran@gmail.com

Dostoevsky Omsk State University, Omsk, Russia

Abstract. The necessary and sufficient conditions for convergence of the distributions of symmetric calibration functions from dependent random variables to max-stable distributions are obtained in this article. These conditions include the so-called minimal conditions of the weak dependence.

Keywords: calibration functions of random variables, max-stable distributions, minimal conditions of the weak dependence.

Дата поступления в редакцию: 15.10.2018

Математические структуры и моделирование 2018. № 4(48). С. 14–20

УДК 537.622.5
DOI: 10.25513/2222-8772.2018.4.14-20

ВЛИЯНИЕ МАГНИТНОГО ПОЛЯ НА ФАЗОВЫЕ ПЕРЕХОДЫ В ПОЛУОГРАНИЧЕННОЙ АНТИФЕРРОМАГНИТНОЙ МОДЕЛИ ИЗИНГА

С.В. Белим
д.ф.-м.н., профессор, e-mail: sbelim@mail.ru

Омский государственный университет им. Ф.М. Достоевского, Омск, Россия

Аннотация. В статье проведено исследование критического поведения трёхмерной полуограниченной антиферромагнитной модели Изинга в магнитном поле методом компьютерного моделирования. Получены значения температуры Нееля для поверхностного и объёмного фазовых переходов при различных значениях напряжённости магнитного поля. Построена фазовая диаграмма системы. Определено положение трикритической точки специального фазового перехода в зависимости от напряжённости магнитного поля.

Ключевые слова: антиферромагнитная модель Изинга, поверхностный фазовый переход, специальный фазовый переход.

Введение

Ферромагнитная модель Изинга на двумерной решётке была изучена уже в 1936 году [1]. К середине шестидесятых годов прошлого века было опубликовано достаточно много статей о поведении модели Изинга в нулевом магнитном поле. Работы, посвящённые ферромагнитной модели Изинга, продолжают активно публиковаться до настоящего времени. Антиферромагнитной модели Изинга посвящено значительно меньше работ. В статье [2] была получена кривая поведения антиферромагнитной модели Изинга в магнитном поле в грубом приближении для двумерного и трёхмерного случаев. Авторы работы показали, что зависимость температуры фазового перехода от напряжённости магнитного поля описывается параболой. Позже в статье [3] было получено аналитическое выражение для температуры Нееля T_N в слабом магнитном поле H:

$$T_N(H) = T_N \left[1 - 0.012(mH/J)^2 + O(H^4)\right].$$

В работе [4] было исследовано поведение в сильных магнитных полях. В результате была получена зависимость магнитного поля от температуры фазового перехода:

$$H = H_C - T_N \ln 2 + O(T_N).$$

Далее в работе [5] было получено выражение для линии вблизи нулевого значения магнитного поля $H = 0$:

$$T_N(H) = T_N^0 \left(1 - 0.038023259H^2\right).$$

Для восприимчивости было получено выражение

$$\chi = 0.014718006H^2 \ln(1/t).$$

В этом выражении введено обозначение для приведённой температуры $t = T/T_N$.

Фазовая диаграмма двумерной антиферромагнитной системы в магнитном поле по результатам компьютерного моделирования [4–11] представлена на рисунке 1. Антиферромагнитная фаза системы находится в заштрихованной области.

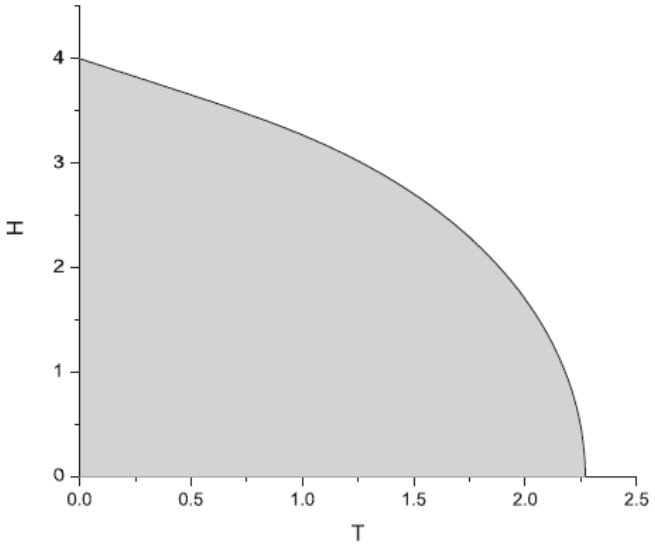


Рис. 1. Фазовая диаграмма двумерной антиферромагнитной системы в магнитном поле

Компьютерное моделирование фазовых переходов в полуограниченных антиферромагнитных системах было осуществлено в работах [12–14] и показало, что в системе может наблюдаться поверхностный фазовый переход при температурах отличных от фазового перехода в основном объёме системы.

1. Описание системы

Гамильтониан антиферромагнитной полуограниченной модели Изинга может быть записан в следующем виде:

$$H = -J_B \sum_B S_i S_j - J_S \sum_S S_i S_j - J_{SB} \sum_{SB} S_i S_j + \sum S_i H.$$

Значение спиновых переменных S_i может принимать одно из двух значений ($+1/2$ или $-1/2$). Суммирование в первых трёх слагаемых берётся только по парам ближайших соседей. В первом слагаемом учитываются только спины, расположенные внутри системы, но не на поверхности. Во втором слагаемом суммирование осуществляется только по спинам, расположенным на свободной поверхности системы. В третьем слагаемом суммирование осуществляется по парам спинов, один из которых расположен на свободной поверхности, а второй — в первом подповерхностном слое. В последнем слагаемом суммирование осуществляется по всем спинам системы. H — напряжённость магнитного поля. J_B, J_S и J_{SB} — значения обменных интегралов взаимодействия между спинами в объёме системы, на свободной поверхности и между спинами поверхности и первым подповерхностным слоем. Как показывает реальный эксперимент [15, 16] и расчёты из первых принципов [17, 18], значение обменного интеграла на поверхности системы J_S может отличаться от обменного интеграла в основном объёме системы J_B, причём возможен как вариант $J_S \geqslant J_B$, так и $J_S < J_B$. Обменный интеграл J_{SB} принимает значения между значениями J_S и J_B. Введём две величины, показывающих отношения обменных интегралов:

$$R = J_S/J_B, \qquad R_1 = J_{SB}/J_B.$$

Будем рассматривать полуограниченные системы. Система будет расположена в полупространстве $z \geqslant 0$. Исследовались системы размером $L \times L \times 2L$. По двум направлениям OX и OY использовались периодические граничные условия. В положительном направлении оси OZ использовались также периодические граничные условия, но для слоя с номером $2L-1$ соседним считался слой с номером $L-1$.

Введём два параметра порядка. Первый параметр порядка m описывает антиферромагнитное упорядочение в основном объёме системы и будет вычисляться как шахматная намагниченность спинов, не расположенных ни на одной из свободных поверхностей:

$$m = \frac{M_1 - M_2}{L^2(2L-1)}, \quad M_1 = \sum_{i=0}^{L^2(2L-1)/2} S_{2i}, \quad M_2 = \sum_{i=0}^{L^2(2L-1)/2} S_{2i+1}.$$

Для описания поверхностного фазового перехода использовался параметр порядка m_S, вычисляемый как шахматная намагниченность спинов, расположенных на поверхности $z = 0$:

$$m_S = \frac{M_{S1} - M_{S2}}{L^2}, \quad M_{S1} = \sum_{i=0}^{L^2/2} S_{2i}, \quad M_{S2} = \sum_{i=0}^{L^2} S_{2i+1}.$$

Для определения критической температуры использовалась теория конечно размерного скейлинга [19], согласно которой куммулянты Биндера четвёртого порядка для систем различного размера пересекаются в одной точке. Эта точка соответствует температуре фазового перехода. Для определения температуры

объёмного фазового перехода T_N использовались куммулянты Биндера для параметра порядка m:

$$U = 1 - \frac{\langle m^4 \rangle}{3\langle m^2 \rangle^2}.$$

Угловые скобки использованы для обозначения термодинамического усреднения. Для нахождения температуры поверхностного фазового перехода использовались куммулянты Биндера, вычисляемые на основе параметра порядка m_S:

$$U_S = 1 - \frac{\langle m_S^4 \rangle}{3\langle m_S^2 \rangle^2}.$$

Для определения точек фазового перехода необходимо найти зависимость куммулянтов Биндера от температуры системы для различных значений линейного размера системы L.

2. Результаты компьютерного моделирования

Компьютерное моделирование осуществлялось для антиферромагнитных полуограниченных систем с линейными размерами от $L = 16$ до $L = 48$ с шагом $\Delta L = 4$. Величина отношения поверхностного обменного интеграла к объёмному была выбрана равной $R_S = 1.6$. При данном значении в системе наблюдается поверхностный фазовый переход. Для второго отношения обменных интегралов было выбрано значение $R_{BS} = R_S$. В компьютерном эксперименте вычислялась температура Нееля T_N и температура поверхностного фазового перехода T_S. Напряжённость магнитного поля изменялась от $H = 0$ до $H = 4.0$ с шагом $\Delta H = 0.5$.

Ранее в работах [12–14] было показано, что в отсутствии магнитного поля в системе реализуются четыре фазы, связанные с упорядочиванием спинов на свободной поверхности системы и в основном объёме системы: полностью неупорядоченная фаза (SD/BD), поверхностно-упорядоченная объёмно-неупорядоченная фаза (SO/BD), поверхностно-упорядоченная объёмно-упорядоченная фаза (SO/BO), поверхностно-неупорядоченная объёмно-упорядоченная фаза (SD/BO). Между этими фазами возможны четыре линии фазовых переходов: из SD/BD в SO/BD – поверхностный фазовый переход, из SO/BD в SO/BO — экстраординарный фазовый переход, из SD/BD в SO/BO — обычный фазовый переход, из SD/BD в SD/BO — подповерхностный фазовый переход. Кривые этих четырёх переходов пересекаются в одной тетракритической точке, фазовый переход в которой получил название специального. Указанные виды фазовых переходов в рамках теоретико-полевого подхода были изучены в работах [20–22].

В данной работе основное внимание было уделено исследованию влияния магнитного поля на обычный и поверхностный фазовые переходы. Графики зависимости температуры фазового перехода для неограниченной системы, температуры обычного фазового перехода при $R_S = R_{BS} = 1.6$ и температуры поверхностного фазового перехода при $R_S = R_{BS} = 1.6$ представлены на рисунке 2. Как и для двумерных систем, зависимость температуры фазового перехода

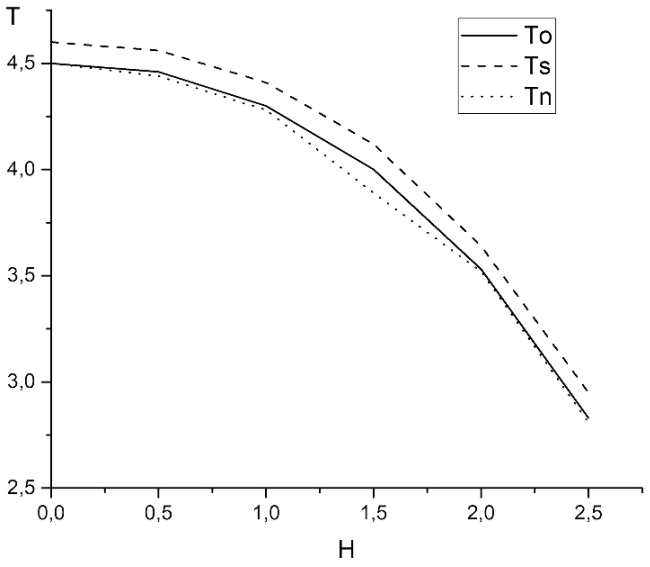


Рис. 2. Графики зависимости температуры фазового перехода для неограниченной системы, температуры обычного фазового перехода при $R_S = R_{BS} = 1.6$ и температуры поверхностного фазового перехода при $R_S = R_{BS} = 1.6$.

от напряжённости магнитного поля может быть с хорошей точностью аппроксимирована квадратичной функцией. Для температуры Нееля неограниченной системы:

$$T_N = T_N(0)\left(1 - (0.059 \pm 0.002)H^2\right),$$

где $T_N(0) = 4.51$ — температура Нееля при нулевом магнитном поле.

Для экстраординарного фазового перехода:

$$T_O = T_O(0)\left(1 - (0.059 \pm 0.002)H^2\right),$$

где $T_O(0) = 4.51$ — температура экстраординарного фазового перехода при нулевом магнитном поле.

Для поверхностного фазового перехода:

$$T_S = T_S(0)\left(1 - (0.057 \pm 0.002)H^2\right),$$

где $T_S(0) = 4.62$ — температура поверхностного фазового перехода при нулевом магнитном поле.

3. Выводы

Как видно из графиков, с точностью до погрешностей вычислений температура экстраординарного фазового перехода совпадает с температурой Нееля для бесконечных систем при всех значениях напряжённости магнитного поля. Также остаётся постоянной разность между температурой экстраординарного фазового перехода и температурой поверхностного фазового перехода. Зависимость температуры от напряжённости магнитного поля носит квадратичный

характер для всех видов фазовых переходов.Температура обычного и экстраординарного фазовых переходов становится нулевой при $H_O = 4.12$. Температура поверхностного фазового перехода принимает нулевое значение при $H_S = 4.19$. Оба значения близки к результату, полученному ранее для двумерных систем $H_{2D} = 4$.

Литература

1. Peirls R.E. On Ising's model of ferromagnetism // Proc. Camb. Philos. Soc. 1936. No. 32. P. 477–481.
2. Domb C., Green M.S. Phase Transitions and Critical Phenomena. V. 3. Academic Press. : London, 1974.
3. Rapaport D.C., Domb C. The smoothness postulate and the Ising antiferromagnet // J. Phys.C. 1971. No. 4(16). P. 2684–2694.
4. Mouller-Hartmann E., Zittartz J. Interface free energy and transition temperature of the square-lattice Ising antiferromagnet at finite magnetic field // Z. Physik B. 1971. No. 27(3). P. 261–266.
5. Monroe J.L. Systematic approximation method for the critical properties of lattice spin systems // Phys. Rev. E. 2001. No. 64. P. 016126.
6. Bulirsh R., Stoer J. Fehlerabschatzungen and Extrapoletion met rationalen Funktionen bei Verfahren vom Richardson-Typus // Numer. Math. 1964. No. 6. P. 413–427.
7. Vanden Broeck J.M., Schwartz L.W. A one-parameter family of sequence transformations // J. Math. Anal. 1979. No. 10. P. 658–666.
8. Wu X.N., Wu F.Y. Critical line of the square-lattice Ising model // Phys. Lett. A. 1990. No. 144. P. 123–126.
9. Blote H.W.J., Wu X.N. Accurate determination of the critical line of the square Ising antiferromagnet in a field // J. Phys. A. 1990. No. 23. P. L627–L629.
10. Wang X.-Z., Kim J.S. The Critical Line of an Ising Antiferromagnet on Square and Honeycomb Lattices // Phys. Rev. Lett. 1997. No. 78. P. 413–416.
11. Tarasenko A.A., Jastrabik L., Nieto F., Uebing C. Adatom diffusion on a square lattice: Comparison of real-space renormalization group and Monte Carlo approaches // Phys. Rev. B. 1999. No. 59. P. 8252–8261.
12. Belim S.V., Trushnikova E.V. Computer Modeling of Phase Transitions of Semibounded Antiferromagnets // Journal of Physics: Conf. Series. 2017. No. 944. P. 012011(1–7).
13. Белим С.В., Трушникова Е.В. Исследование критического поведения полуограниченных антиферромагнетиков методами компьютерного моделирования // Физика металлов и металловедение. 2018. Т. 119, Вып. 5. С. 465–471.
14. Белим С.В., Трушникова Е.В. Исследование поверхностного фазового перехода полуограниченных антиферромагнитных систем методом компьютерного моделирования // Поверхность. Рентгеновские, синхротронные и нейтронные исследования. 2018. № 9. С. 102–105.
15. Ruiz-Diaz P., Stepanyuk V.S. Effects of surface charge doping on magnetic anisotropy in capping 3d-5d(4d) multilayers deposited on highly polarizable substrates // J. Phys. D. Appl.Phys. 2014. No. 47. P. 105006.

16. Brovko O.O., Ruiz-Dнaz P., Dasa T.R., Stepanyuk V.S. Controlling magnetism on metal surfaces with non-magnetic means: electric fields and surface charging // J. Phys. Condens. Matter. 2014. No. 26. P. 093001.
17. Lin C.-Yu., Li J.-L., Hsieh Y.-H., Ou K.-L., Jones B.A. Magnetic interaction between surface-engineered rare-earth atomic spins // Phys. Rev. X. 2012. No 2. P. 021012.
18. Ruiz-Diaz P., Dasa T.R., Stepanyuk V.S. Tuning Magnetic Anisotropy in Metallic Multilayers by Surface Charging: An Ab Initio Study.// Phys. Rev. Lett. 2013. No. 110. P. 267203.
19. Landau D.P., Binder K. Phase diagrams and multicritical behavior of a three-dimensional anisotropic Heisenberg antiferromagnet // Phys. Rev. B. 1978. No. 17. P. 2328–2342.
20. Diehl H.W., Shpot M. Massive field-theory approach to surface critical behavior in three-dimensional systems // Nucl. Phys. B. 1998. No. 528. P. 595–647.
21. Белим С.В. Критическое поведение неупорядоченных систем со свободной поверхностью // ЖЭТФ. 2006. Т. 130, Вып. 4(10). С. 702–714.
22. Белим С.В. Мультикритическое поведение систем со свободной поверхностью // ЖЭТФ. 2008. Т. 133, Вып. 4. С. 884–891.

EFFECT OF A MAGNETIC FIELD ON PHASE TRANSITIONS IN A SEMI-BOUNDED ANTIFERROMAGNETIC ISING MODEL

S.V. Belim
Dr.Sc. (Phys.-Math.), Professor, e-mail: sbelim@mail.ru

Dostoevsky Omsk State University, Omsk Russia

Abstract. The article investigated the critical behavior of three-dimensional semi-infinite antiferromagnetic Ising model in a magnetic field by computer simulation. The values of the Neel temperature for surface and bulk phase transitions at different magnetic field values are received. The phase diagram of the system is built. The position of tricritical point of a special phase transition depending on the strength of the magnetic fields is determined.

Keywords: antiferromagnetic Ising model, surface phase transition, special phase transition.

Дата поступления в редакцию: 17.11.2018

Математические структуры и моделирование 2018. № 4(48). С. 21–26

УДК 514.764.7+517.926.7
DOI: 10.25513/2222-8772.2018.4.21-26

О КРИВЫХ С ПОСТОЯННЫМИ КРИВИЗНАМИ В ПСЕВДОЕВКЛИДОВОМ ПРОСТРАНСТВЕ

И.А. Зубарева
к.ф.-м.н., доцент, e-mail: i_gribanova@mail.ru

Институт математики им. С.Л. Соболева СО РАН, Омск, Россия

Аннотация. Автор доказала, что все кривизны регулярной кривой в n–мерном псевдоевклидовом пространстве $\mathbb{E}^n_l$, $n \geqslant 2$, произвольного индекса l постоянны тогда и только тогда, когда эта кривая есть орбита некоторой однопараметрической подгруппы группы всех движений пространства $\mathbb{E}^n_l$.

Ключевые слова: кривизна, репер Френе, орбита однопараметрической группы изометрий, псевдоевклидово пространство.

Предварительные сведения и вспомогательное утверждение

Кривые с постоянными кривизнами в n–мерном евклидовом пространстве, $n \geqslant 3$, бегло рассмотрены в книге Ю.А. Аминова [1]. Там показано, что вид кривых с постоянными ненулевыми кривизнами в чётномерных и нечётномерных евклидовых пространствах существенно различается. Именно в чётномерном евклидовом пространстве такая кривая ограничена и является сферической, т. е. лежит на некоторой сфере, а в нечётномерном она уходит по одному направлению в бесконечность. Несколько более расширенное утверждение другим способом было получено в книге С.В. Сизого [2].

В этой статье доказано, что регулярные кривые с постоянными кривизнами в n–мерном псевдоевклидовом пространстве $\mathbb{E}^n_l$, $n \geqslant 2$, произвольного индекса l есть в точности орбиты однопараметрических подгрупп группы всех движений этого пространства.

Псевдоевклидово пространство $\mathbb{E}^n_l$ индекса l, где n, l — целые числа, $n \geqslant 2$, $0 \leqslant l \leqslant n/2$, есть n–мерное векторное пространство *с псевдоскалярным произведением*

$$\{(x_1, \ldots, x_n), (y_1, \ldots, y_n)\} := -x_1y_1 - \cdots - x_ly_l + x_{l+1}y_{l+1} + \ldots x_ny_n.$$

Очевидно, что $\mathbb{E}^n_0$ — n–мерное евклидово пространство. *Скалярный квадрат* вектора $x = (x_1, \ldots, x_n)$ в $\mathbb{E}^n_l$ имеет вид

$$x^2 := \{x, x\} = -x_1^2 - \cdots - x_l^2 + x_{l+1}^2 + \ldots x_n^2$$

и может быть положительным, отрицательным (при $l > 0$) или равным нулю. Число $\|x\| := \sqrt{|\ x^2\ |}$ называется *длиной* вектора x.

Пусть γ — регулярная кривая в $\mathbb{E}_l^n$, $r = r(s)$, $s \in \mathbb{R}$, — её естественная параметризация, т. е. $\|r'(s)\| \equiv 1$. Определим $q_0(s) = r'(s)$. Если $\|q_{m-1}(s)\| \neq 0$, $m = 1, \ldots, n$, то положим $e_m(s) = q_{m-1}(s)/\|q_{m-1}(s)\|$, $\varepsilon_m(s) = e_m^2(s)$,

$$q_m(s) = e_m'(s) - \sum_{i=1}^{m} \varepsilon_i(s)\{e_m'(s), e_i(s)\}e_i(s).$$

Если $\|q_{m-1}(s)\| = 0$, то векторы $e_i(s)$, $i = m, \ldots, n$, не определены.

Пусть $\|q_i(s)\| \neq 0$, $i = 1, \ldots, n-1$. Нетрудно показать, что система векторов $\{e_i(s)\}_{i=1,\ldots,n}$ ортонормальна, т. е. $\|e_i(s)\| = 1$, $\{e_i(s), e_j(s)\} = 0$ при $i \neq j$, $i, j = 1, \ldots, n$. Эта система называется базисом Френе кривой γ в точке $r(s)$. При этом выполнены следующие формулы Френе (см. [3]):

$$\begin{cases} e_1'(s) = \kappa_1(s)e_2(s), \\ e_i'(s) = -\varepsilon_{i-1}(s)\varepsilon_i(s)\kappa_{i-1}(s)e_{i-1}(s) + \kappa_i(s)e_{i+1}(s), \quad i = 2, \ldots, n-1, \\ e_n'(s) = -\varepsilon_{n-1}(s)\varepsilon_n(s)\kappa_{n-1}(s)e_{n-1}(s). \end{cases} \tag{1}$$

Коэффициенты $\kappa_1(s)$, $\kappa_2(s), \ldots, \kappa_{n-1}(s)$ называются первой, второй, ..., $(n-1)$–ой кривизной кривой γ в точке $r(s)$, причём все они положительны.

Пусть теперь $\|q_m(s)\| = 0$ для некоторого $m \in \{1, \ldots, n-1\}$. В этом случае система векторов Френе $\{e_i(s)\}_{i=1,\ldots,m}$ ортонормальна, и выполнены следующие формулы Френе:

$$\begin{cases} e_1'(s) = \kappa_1(s)e_2(s), \\ e_i'(s) = -\varepsilon_{i-1}(s)\varepsilon_i(s)\kappa_{i-1}(s)e_{i-1}(s) + \kappa_i(s)e_{i+1}(s), \quad i = 2, \ldots, m-1, \\ e_m'(s) = -\varepsilon_{m-1}(s)\varepsilon_m(s)\kappa_{m-1}(s)e_{m-1}(s). \end{cases} \tag{2}$$

Коэффициенты $\kappa_1(s)$, $\kappa_2(s), \ldots, \kappa_{m-1}(s)$ называются первой, второй, ..., $(m-1)$–ой кривизной кривой γ в точке $r(s)$, причём все они положительны. При этом m–ая кривизна $\kappa_m(s)$ равна нулю, а остальные кривизны $\kappa_{m+1}(s), \ldots, \kappa_{n-1}(s)$ не определены.

Предложение 1. *Функции* $\varepsilon_i(s) := e_i^2(s)$, $s \in \mathbb{R}$, $i = 1, \ldots, m$, $m \leqslant n$, *постоянны, т. е.* $\varepsilon_i(s) \equiv \varepsilon_i$.

Доказательство. Из (2), определения функций $\varepsilon_i(s)$ и ортонормальности системы векторов $e_i(s)$, $i = 1, \ldots, m$, следует, что для любого $s \in \mathbb{R}$

$$\varepsilon_1'(s) = 2\{e_1'(s), e_1(s)\} = 2\kappa_1(s)\{e_2(s), e_1(s)\} \equiv 0,$$

$$\varepsilon_i'(s) = 2\{e_i'(s), e_i(s)\} = -\varepsilon_{i-1}(s)\varepsilon_i(s)\kappa_{i-1}(s)\{e_{i-1}(s), e_i(s)\} +$$
$$+\kappa_i(s)\{e_{i+1}(s), e_i(s)\} \equiv 0, \quad i = 2, \ldots, m-1,$$

$$\varepsilon_m'(s) = -\varepsilon_{m-1}(s)\varepsilon_m(s)\kappa_{m-1}(s)\{e_{m-1}(s), e_m(s)\} \equiv 0.$$

Поэтому функции $\varepsilon_i(s)$, $i = 1, \ldots, m$, постоянны. ■

1. Основной результат

Основной результат работы составляет следующая теорема.

Теорема 1. *Все кривизны регулярной кривой в n–мерном псевдоевклидовом пространстве $\mathbb{E}_l^n$, где n, l — целые числа, $n \geqslant 2$, $0 \leqslant l \leqslant n/2$, постоянны тогда и только тогда, когда эта кривая есть орбита некоторой однопараметрической подгруппы группы всех движений пространства $\mathbb{E}_l^n$.*

Доказательство. Достаточность теоремы 1 очевидна. Докажем необходимость.

1. Пусть γ — регулярная кривая в псевдоевклидовом пространстве $\mathbb{E}_l^n$ с естественной параметризацией $r(s)$, $s \in \mathbb{R}$, имеющая постоянные ненулевые кривизны $\kappa_1, \ldots, \kappa_{n-1}$.

Систему дифференциальных уравнений Френе (1) для векторов $e_1(s)$,…,$e_n(s)$ можно записать в виде $X'(s) = AX(s)$, где $X(s)$ — квадратная матрица порядка n, i–ая строчка которой есть декартовы координаты вектора $e_i(s)$, $i = 1, \ldots, n$, A — трёхдиагональная матрица системы (1). Тогда

$$X(s) = \exp(sA)X(0). \tag{3}$$

Обозначим через I диагональную матрицу порядка n такую, что $I_{ii} = -1$ при $i = 1, \ldots, l$ и $I_{ii} = 1$ при $i = l+1, \ldots, n$.

Лемма 1. *Для каждого $s \in \mathbb{R}$ матрица $X(s)IX^T(s)$ есть диагональная матрица n–го порядка, причём $(X(s)IX^T(s))_{ii} = \varepsilon_i$, $i = 1, \ldots, n$.*

Доказательство. Из определений матриц $X(s)$, I, псевдоскалярного произведения в $\mathbb{E}_l^n$, ортонормальности системы векторов $e_1(s), \ldots, e_n(s)$ и предложения 1 следует, что для любых $i, j = 1, \ldots, n$

$$(X(s)IX^T(s))_{ij} = \{e_i(s), e_j(s)\} = \begin{cases} 0, \text{ если } i \neq j, \\ \varepsilon_i, \text{ если } i = j. \end{cases}$$

■

Положим

$$C(s) = (X(0)I)^{-1} \exp(-sA)\,(X(0)I)\,, \quad s \in \mathbb{R}. \tag{4}$$

Лемма 2. *Для любого $s \in \mathbb{R}$, $C^T(s) = X^{-1}(0)X(s)$.*

Доказательство. В силу определения матрицы I выполнено $I = I^{-1} = I^T$. На основании (4), (3), леммы 1 и равенства $\varepsilon_i^2 = 1$, $i = 1, \ldots, n$, следует, что

$$X(0)C^T(s) = \left(X(0)IX^T(0)\right)\left(\exp(-sA)\right)^T \left((X(0)I)^{-1}\right)^T =$$

$$= \left(X(0)IX^T(0)\right)\left((\exp(sA)X(0)I)^{-1}\right)^T =$$

$$= \left(X(0)IX^T(0)\right)\left(\left(X(s)IX^T(s)\right)^{-1}\right)^T X(s) = X(s).$$

■

Определим вектор $d(s) = (d_1(s), \dots, d_n(s))$, $s \in \mathbb{R}$, формулой

$$d(s) = r(s) - r(0)C^T(s). \tag{5}$$

Тогда $r(s) = \Psi(s)\,(r(0))$, где $\Psi(s)$, $s \in \mathbb{R}$, есть аффинное преобразование псевдоевклидова пространства $\mathbb{E}_l^n$, задаваемое формулой

$$\Psi(s)(x) = xC^T(s) + d(s), \quad x = (x_1, \dots, x_n) \in \mathbb{E}_l^n. \tag{6}$$

Нам понадобится следующая лемма.

Лемма 3. *Множество $\{\Psi(s),\ s \in \mathbb{R}\}$, определённое формулой (6), есть однопараметрическая подгруппа группы всех движений псевдоевклидова пространства $\mathbb{E}_l^n$.*

Доказательство. Покажем сначала, что аффинное преобразование $\Psi(s)$, $s \in \mathbb{R}$, есть движение псевдоевклидова пространства $\mathbb{E}_l^n$. Вследствие (6) достаточно доказать, что для любых $a = (a_1, \dots, a_n)$, $b = (b_1, \dots, b_n) \in \mathbb{E}_l^n$ выполнено равенство $\{\alpha(s), \beta(s)\} = \{a, b\}$, где $\alpha(s)$, $\beta(s) \in \mathbb{E}_l^n$ определены формулами

$$\alpha(s) = aC^T(s), \quad \beta(s) = bC^T(s).$$

На основании (3), (4) последние равенства можно записать в виде

$$\alpha(s)\,(X(s)I)^T = a\,(X(0)I)^T, \quad \beta(s)\,(X(s)I)^T = b\,(X(0)I)^T.$$

Из определений псевдоскалярного произведения в $\mathbb{E}_l^n$, матриц $X(s)$, $s \in \mathbb{R}$, и I следует, что последние равенства равносильны равенствам

$$\{e_i(s), \alpha(s)\} = \{e_i(0), a\}, \quad \{e_i(s), \beta(s)\} = \{e_i(0), b\}, \quad i = 1, \dots, n. \tag{7}$$

Так как для каждого $s \in \mathbb{R}$ система векторов $\{e_i(s)\}_{i=1,\dots,n}$ — ортонормальный базис псевдоевклидова пространства $\mathbb{E}_l^n$, $e_i^2(s) = \varepsilon_i$, $i = 1, \dots, n$, (см. предложение 1), то

$$a = \sum_{i=1}^{n} \varepsilon_i \{e_i(0), a\} e_i(0), \ \ b = \sum_{i=1}^{n} \varepsilon_i \{e_i(0), b\} e_i(0),$$

$$\alpha(s) = \sum_{i=1}^{n} \varepsilon_i \{e_i(s), \alpha(s)\} e_i(s), \ \ \beta(s) = \sum_{i=1}^{n} \varepsilon_i \{e_i(s), \beta(s)\} e_i(s).$$

Отсюда и из (7) следует, что

$$\{a, b\} = \sum_{i=1}^{n} \varepsilon_i \{e_i(0), a\}\{e_i(0), b\} = \sum_{i=1}^{n} \varepsilon_i \{e_i(s), \alpha(s)\}\{e_i(s), \beta(s)\} = \{\alpha(s), \beta(s)\},$$

т. е. всякое преобразование $\Psi(s)$, $s \in \mathbb{R}$, есть движение псевдоевклидова пространства $\mathbb{E}_l^n$.

Осталось доказать, что множество движений $\{\Psi(s),\, s \in \mathbb{R}\}$ псевдоевклидова пространства $\mathbb{E}_l^n$ образует однопараметрическую подгруппу, т. е. $\Psi(s+t)(x) = \Psi(t)(\Psi(s)(x))$ для любых $s,\, t \in \mathbb{R}$, $x \in \mathbb{E}_l^n$. На основании (6) это эквивалентно тому, что

$$C(s+t) = C(t)C(s), \quad d(s+t) = d(s)C^T(t) + d(t), \quad s,t \in \mathbb{R}. \tag{8}$$

Первое равенство (8) непосредственно следует из (4) и свойств матричной экспоненты. Тогда на основании (5) второе равенство (8) равносильно равенству

$$r(s+t) - r(s)C^T(t) = r(t) - r(0)C^T(t), \quad s,t \in \mathbb{R}.$$

Таким образом, достаточно доказать, что при каждом фиксированном $t \in \mathbb{R}$ вектор–функция $R(s) := r(s+t) - r(s)C^T(t)$ переменного s постоянна.

Обозначим через $\vec{i}$ вектор в $\mathbb{E}_l^n$, первая координата которого равна 1, а остальные координаты равны нулю. Из определений вектора $e_1(s)$ и матрицы $X(s)$, (3) следует, что производная вектор–функции $R(s)$ равна

$$R'(s) = e_1(s+t) - e_1(s)C^T(t) = \vec{i}X(s+t) - \left(\vec{i}X(s)\right)C^T(t) =$$

$$= \vec{i}\,(\exp((s+t)A)X(0)) - \left(\vec{i}(\exp(sA)X(0))\right)C^T(t) =$$

$$= \left(\vec{i}\exp(sA)\right)\left(\exp(tA)X(0) - X(0)C^T(t)\right).$$

Тогда, вследствие (3) и леммы 2, $R'(s)$ есть нулевой вектор, и второе равенство (8) доказано.

■

2. Пусть теперь γ — регулярная кривая в псевдоевклидовом пространстве $\mathbb{E}_l^n$ с естественной параметризацией $r(s)$, $s \in \mathbb{R}$, имеющая постоянные кривизны $\kappa_1, \ldots, \kappa_m$, причём $\kappa_m = 0$, $m < n-1$. Дополним ортонормальную систему векторов $e_i(0)$, $i = 1, \ldots, m$, до ортонормального базиса $e_1(0), \ldots, e_m(0)$, $e_{m+1}, \ldots, e_n$ псевдоевклидова пространства $\mathbb{E}_l^n$. Фиксируем число $k \in \{m+1, \ldots, n\}$ и определим функции $f_i(s) = \{e_i(s), e_k\}$, $i = 1, \ldots, m$. Вследствие (2) и предложения 1 эти функции удовлетворяют линейной системе дифференциальных уравнений с постоянными коэффициентами

$$\begin{cases} f_1'(s) = \kappa_1 f_2(s), \\ f_i'(s) = -\varepsilon_{i-1}\varepsilon_i\kappa_{i-1}f_{i-1}(s) + \kappa_i f_{i+1}(s), \quad i = 2, \ldots, m-1, \\ f_m'(s) = -\varepsilon_{m-1}\varepsilon_m\kappa_{m-1}f_{m-1}(s), \end{cases} \tag{9}$$

причём $f_i(0) = 0$, $i = 1, \ldots, m$. Система (9) имеет единственное решение $f_i(s) \equiv 0$, $i = 1, \ldots, m$. Следовательно, для каждого $s \in \mathbb{R}$ векторы $e_1(s), \ldots, e_m(s)$, $e_{m+1}(s) := e_{m+1}, \ldots, e_n(s) := e_n$ задают ортонормальный базис псевдоевклидова пространства $\mathbb{E}_l^n$. Доопределим $\varepsilon_i = e_i^2$, $i = m+1, \ldots, n$.

Систему дифференциальных уравнений Френе (2) для векторов $e_1(s),\dots,e_n(s)$ можно записать в виде $X'(s) = AX(s)$, поэтому выполнено равенство (3). Здесь $X(s)$ — квадратная матрица порядка n, i–ая строчка которой есть декартовы координаты вектора $e_i(s)$, $i = 1,\dots,n$, A — блочно–диагональная матрица, состоящая из двух блоков, верхний — матрица системы уравнений (2), нижний — нулевая матрица порядка $n - m$.

Пусть снова I — диагональная матрица порядка n такая, что $I_{ii} = -1$ при $i = 1,\dots,l$ и $I_{ii} = 1$ при $i = l+1,\dots,n$. Определим аффинное преобразование $\Psi(s)$, $s \in \mathbb{R}$, псевдоевклидова пространства $\mathbb{E}^n_l$ формулой (6), где матрица $C(s)$ и вектор $d(s)$ заданы формулами (4) и (5) соответственно. Тогда выполнена лемма 3, и кривая $r(s)$ является орбитой точки $r(0)$ относительно однопараметрической подгруппы $\{\Psi(s),\ s \in \mathbb{R}\}$ группы движений псевдоевклидова пространства $\mathbb{E}^n_l$.

Теорема 1 доказана.

■

2. Благодарности

Автор благодарит профессора В.Н. Берестовского за постановку задачи и полезные замечания.

Литература

1. Аминов Ю.А. Дифференциальная геометрия и топология кривых. М. : Наука. Гл. ред. физ.–мат. лит., 1987.
2. Сизый С.В. Лекции по дифференциальной геометрии. М. : Физматлит, 2007.
3. Борисов Ю.Ф. Снятие априорных ограничений в теореме о полной системе инвариантов кривой в $\mathbb{E}^n_l$ // Сиб. мат. журн. 1997. Т. 38, № 3. С. 485–503.

ON CURVES WITH CONSTANT CURVATURES IN THE PSEUDO–EUCLIDEAN SPACE

I.A. Zubareva
Ph.D. (Phys.-Math.), Associate Professor, e-mail: i_gribanova@mail.ru

Institute of Mathematics S.L. Sobolev SB RAS, Omsk, Russia

Abstract. The author proved all curvatures of a regular curve in n–dimensional pseudo–Euclidean space of an arbitrary index, $n \geqslant 2$, are constant if and only if the curve is an orbit of a one–parameter isometry group of the space.

Keywords: curvature, Frenet frame, orbit of a one-parameter isometry group, pseudo–Euclidean space.

Дата поступления в редакцию: 02.11.2018

Математические структуры и моделирование 2018. № 4(48). С. 27–34

УДК 519.852
DOI: 10.25513/2222-8772.2018.4.27-34

$B\mathcal{H}$–БАЗИСЫ ДЛЯ ОДНОГО КЛАССА ФАСЕТ МНОГОГРАННИКА РАЗБИЕНИЯ НА КЛИКИ

Р.Ю. Симанчев
к.ф.-м.н., доцент, e-mail: osiman@rambler.ru
П.В. Соловьева
магистрант ИМИТ, e-mail: polinochka.chervonnykh@mail.ru

Омский государственный университет им. Ф.М. Достоевского, Омск, Россия

Аннотация. Пусть $K_n = (V, E)$ — полный неориентированный n–вершинный граф без петель и кратных рёбер. Остовный подграф $H \subset K_n$ называется M–графом, если каждая его компонента связности (возможно одновершинная) является кликой. Иными словами, всякий M–граф является правильным разбиением графа K_n на вершинно-непересекающиеся клики. Семейство всех M–графов в K_n обозначим через $\mathcal{H}$. Это семейство является множеством допустимых решений задачи о разбиении на клики, заключающейся в нахождении в полном рёберно-взвешенном графе M–графа минимального веса [2–4]. В упомянутых работах рассматриваются полиэдральные свойства [1] этой задачи, а именно строятся классы неравенств, порождающих грани многогранника задачи, на базе которых разработаны алгоритмы ветвей и отсечений. В настоящей работе, применяя технику, предложенную в [7], мы доказываем фасетность неравенств специального класса относительно многогранника задачи разбиения на клики.
Работа выполнена при поддержке РФФИ (проект 18-07-00599).

Ключевые слова: многогранник, фасета, задача разбиения на клики.

Введение

Для любого графа $D \subset K_n$ через VD и ED будем обозначать множества его вершин и рёбер соответственно. Для ребра $e \in E$ будем также использовать запись uv, где u, v — вершины из V, инцидентные ребру e. Множество рёбер, инцидентных вершине u будем обозначать как $\delta(u)$. Каждое множество $R \subset E$ индуцирует некоторый подграф T, в котором $ET = R$ и VT — множество вершин из V, инцидентных рёбрам из R. Граф, индуцированный множеством рёбер R, иногда будем обозначать через R. Для подграфов D, F из K_n положим

$$D \cup F = (VD \cup VF, ED \cup EF), D \cap F = ED \cap EF,$$

и если $F \subseteq D$, то $D \backslash F = (VD, ED \backslash EF)$. Кликой в K_n называется подграф, вершины которого попарно смежны. Одновершинный граф также является кликой.

С графом K_n свяжем евклидово пространство R^E размерности $\frac{n^2-n}{2}$, поставив в соответствие каждому ребру ось координат в R^E. Это пространство может рассматриваться как множество вектор-столбцов, компоненты которых индексируются элементами из E. Если $x \in R^E$ и $R \subset E$, то через $x(R)$ обозначим линейную форму $\sum\limits_{e \in R} x_e$. Вектором инциденций произвольного подграфа $D \subset K_n$ называется вектор $x^D \in R^E$ с компонентами $x_e^D = 1$ при $e \in ED$ и $x_e^D = 0$ при $e \notin ED$.

Множество $P \subset R^E$ называется многогранником, если P является выпуклой оболочкой конечного числа точек. Под размерностью ($\dim P$) многогранника P будем понимать уменьшенную на 1 мощность максимального по включению аффинно независимого семейства его точек. Если $\dim P = |E|$, то будем называть P многогранником полной размерности.

Линейное неравенство $a^t x \leqslant a_0$ $(a, x \in R^E,\ a \neq 0,\ a_0 \in R)$ называется правильным относительно многогранника P, если $a^t x \leqslant a_0$ для любого $x \in P$. Правильное неравенство $a^t x \leqslant a_0$ называется опорным к P, если существуют $x', x'' \in P$ такие, что $a^t x' = a_0$ и $a^t x'' < a_0$. Всякое опорное к P неравенство порождает множество $\{x \in P | a^t x = a_0\}$, которое называется гранью многогранника P. Грани размерности 0 будем называть вершинами, а грани размерности $(\dim P - 1)$ — фасетами многогранника P. Неравенство, порождающее фасету многогранника P, называется фасетным относительно этого многогранника. Многогранником M–графов или, что то же, многогранником задачи разбиения на клики называется множество

$$P_{\mathcal{H}} = conv\{x^H \in R^E | H \in \mathcal{H}\}.$$

1. Класс правильных неравенств

Будем рассматривать множество E рёбер графа K_n в качестве основного множества. В качестве семейства подмножеств $\mathcal{H} \subseteq 2^E$ возьмём семейство всех M–графов в графе K_n. Пусть $W = \{v_1, v_2, ..., v_p\}$ — упорядоченное подмножество множества V, p — нечётно. Через F обозначим звезду в K_n с центром в вершине $u \notin W$ и лучами uv_j, а через C — цикл с множеством вершин W и множеством рёбер $\{v_1v_2, ..., v_{p-1}v_p, v_pv_1\}$. С графом $F \cup C$ свяжем линейное неравенство

$$x(EF) - x(EC) \leqslant \left\lfloor \frac{p}{2} \right\rfloor$$

или, что то же,

$$(x^F - x^C)^t x \leqslant \left\lfloor \frac{p}{2} \right\rfloor. \tag{1}$$

Лемма 1. *Неравенство (2) является опорным относительно $P_{\mathcal{H}}$.*

Доказательство. Пусть множество $EF \cap EH = \{uv | v \in U\}$, $U \subseteq W$, тогда $|EF \cap EH| = |U| = s$. Если $v_i, v_{i+1} \in U$, то $v_iv_{i+1} \in EC \cap EH$ (вершины из W индексируются по модулю p). Следовательно, множество $EC \cap EH$ будет являться набором цепей. При этом между двумя соседними цепями есть, по

крайней мере, одна вершина, не лежащая в U. Обозначим эти цепи $P_1, P_2, ..., P_t$, причём $|VP_i| \geqslant 1$, $i = 1, 2, ..., t$. В связи с этим $s = \sum\limits_{i=1}^{t} s_i$, где s_i — количество вершин в цепи P_i. Пусть $v(P_i) \in V \setminus U$ — вершина, непосредственно следующая за последней вершиной цепи P_i в цикле C. Тогда множество $VP_i \cup \{v(P_i)\}$ назовём блоком, $i = 1, 2, ..., t$. Нетрудно заметить, что $|VP_i \cup \{v(P_i)\}| \geqslant 2$.

Найдём верхнюю оценку числа t. Очевидно, что максимально возможное количество цепей будет при условии минимальности длин этих цепей. Соответственно, это будут цепи длины 0, т. е. цепи, состоящие из 1-ой вершины. Так как p — нечётно и в данном случае $|VP_i \cup \{v(P_i)\}| = 2$, то количество одновершинных цепей будет не больше, чем количество всех вершин в W, без учёта вершины, не вошедшей в блок, уменьшенное в 2 раза. Иначе говоря, $t \leqslant \frac{p-1}{2} = \lfloor \frac{p}{2} \rfloor$.

Легко заметить, что $x^H(EC) = |EC \cap EH| = \sum\limits_{i=1}^{t} |P_i| = \sum\limits_{i=1}^{t} (s_i - 1) = \sum\limits_{i=1}^{t} s_i - t = s - t$, где $|P_i|$ — длина цепи P_i, $i = 1, 2, ..., t$. Следовательно, $x^H(EF) - x^H(EC) = |EH \cap EF| - |EH \cap EC| = s - s + t = t \leqslant \lfloor \frac{p}{2} \rfloor$. Правильность неравенства (2) относительно $P_{\mathcal{H}}$ доказана.

Опорность этого неравенства к многограннику $P_{\mathcal{H}}$ следует из того, что вектор инциденций, например, клики на вершинах $\{u, v_1, v_3, v_5, ..., v_{p-2}\}$, обращает его в равенство.

Лемма доказана. ■

2. Фасетность

В работе [3] представлена техника доказательства фасетности опорного неравенства. Применительно к многограннику полной размерности эта техника имеет следующий вид.

Пусть $b^t x \leqslant b_0$ — опорное к $P_{\mathcal{H}}$ неравенство.

Определение 1. Непустое множество $S \subset E$ будем называть $b\mathcal{H}$–переключением, если существуют такие $H_1, H_2 \in \mathcal{H}$, что:

1) $S = H_1 \Delta H_2$,
2) $b^t x^{H_1} = b^t x^{H_2} = b_0$,

где $H_1 \Delta H_2 = (H_1 \setminus H_2) \cup (H_2 \setminus H_1)$ — симметрическая разность множеств H_1 и H_2.

Определение 2. Элемент $e_0 \in E$ называется $b\mathcal{H}$–базисом, если выполняются следующие условия:

(i) $b_{e_0} \neq 0$,

(ii) для всякого $e \in E \setminus \{e_0\}$ существует такая упорядоченная последовательность $e_1, e_2, ..., e_t = e$ элементов из E, что при любом $i \in \{1, 2, ..., t\}$ элемент e_i принадлежит некоторому $b\mathcal{H}$–переключению, лежащему в $\{e_0, e_1, e_2, ..., e_i\}$.

Теорема 1. *[3]. Для того чтобы опорное к $P_{\mathcal{H}}$ неравенство $b^t x \leqslant b_0$ было фасетным, достаточно существования $b\mathcal{H}$–базиса $e_0 \in E$.*

Следуя описанной технике доказательства фасетности опорного неравенства относительно многогранника $P_{\mathcal{H}}$, сформулируем утверждение.

Предложение 1. *Пусть $\mathcal{H}$ — семейство всех M–графов в графе K_n, $(x^F - x^C)^t x \leqslant \lfloor \frac{p}{2} \rfloor$ — неравенство вида (2), индуцированное графом $F \cup C$. Следующие множества рёбер являются $(x^F - x^C)\mathcal{H}$–переключениями:*

a) одноэлементные множества рёбер $\{st\}$, при $s, t \notin V(F \cup C)$;

b) множество вида $\{su, sv_i, sv_{i+2}, sv_{i+4}, ..., sv_{i+(p-3)}\}$, при $s \notin V(F \cup C)$, $i = 1, 2, ..., p$ (индексы берутся по модулю p);

c) хорды цикла C, т. е. множества рёбер вида $\{v_i v_j\}$, $2 \leqslant |i - j| \leqslant p - 2$;

d) множество вида $\{uv_i, v_i v_{i+1}, v_i v_{i+3}, ..., v_i v_{i+(p-2)}\}$, $i = 1, 2, ..., p$ (индексы берутся по модулю p);

e) одноэлементные множества рёбер $\{ v_i s \}$, при $s \notin V(F \cup C)$, $i = 1, 2, ..., p$ (индексы берутся по модулю p).

Доказательство. Случай a). Положим H_1 — клика на вершинах $\{u, v_1, v_3, ..., v_j, ..., v_{p-2}\}$, и $H_2 = H_1 \cup \{st\}$. Тогда $H_1 \Delta H_2 = \{st\}$ и $(x^F - x^C)^t x^{H_1} = (x^F - x^C)^t x^{H_2} = \lfloor \frac{p}{2} \rfloor$.

Случай b). H_1 — клика на вершинах $\{u, v_i, v_{i+2}, v_{i+4}, ..., v_{i+(p-3)}\}$ и H_2 — клика на вершинах $\{u, s, v_i, v_{i+2}, v_{i+4}, ..., v_{i+(p-3)}\}$, $i = 1, 2, ..., p$ (индексы берутся по модулю p).

Случай c). Пусть имеется хорда $v_i v_j$, $2 \leqslant |i - j| \leqslant p - 2$. Будем полагать, что $i < j$. Кроме того, без ограничения общности положим $i = 1$. Построим клику H_1 следующим образом. Если $j = 1(mod 2)$, то H_1 — клика на множестве вершин $\{u, v_2, v_4, \ldots, v_{p-1}\}$. Ясно, что в этом случае вершина v_j не принадлежит VH_1. Если же $j = 0(mod 2)$, то H_1 — клика на множестве вершин $\{u, v_2, v_4, \ldots, v_{j-2}, v_{j+1}, \ldots, v_p\}$. Вновь $v_j \notin VH_1$. Пусть $H_2 = H_1 \cup \{v_1 v_j\}$. Ясно, что H_1 и H_2 — M–графы, требуемые определением 1.

Случай d). H_1 — клика на вершинах $\{u, v_i, v_{i+1}, ..., v_{i+(p-2)}\}$, $i = 1, 2, ..., p$ (индексы берутся по модулю p) и $H_2 = H_1 \cup \{uv_i, v_i v_{i+1}, v_i v_{i+3}, ..., v_i v_{i+(p-2)}\}$, и $(x^F - x^C)^t x^{H_1} = (x^F - x^C)^t x^{H_2} = \lfloor \frac{p}{2} \rfloor$.

Случай e). H_1 — клика на вершинах $\{u, v_{i+1}, v_{i+3}, v_{i+5}, ..., v_{i+(p-2)}\}$, $i = 1, 2, ..., p$ (индексы берутся по модулю p) и $H_2 = H_1 \cup \{v_i s\}$.

Утверждение доказано. ■

Пример 1. В качестве иллюстрации к доказательству утверждения 1 рассмотрим граф $F \cup C$ с $p = 7$. Тогда:

Теорема 2. *Неравенство $(x^F - x^C)x \leqslant \lfloor \frac{p}{2} \rfloor$ порождает фасету многогранника M–графов.*

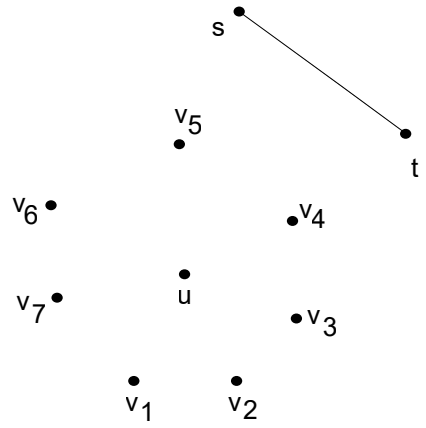


Рис. 1. $(x^F - x^C)\mathcal{H}$–переключение вида a)

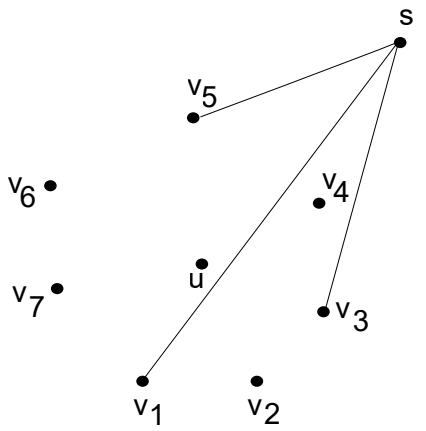


Рис. 2. $(x^F - x^C)\mathcal{H}$–переключение вида b)

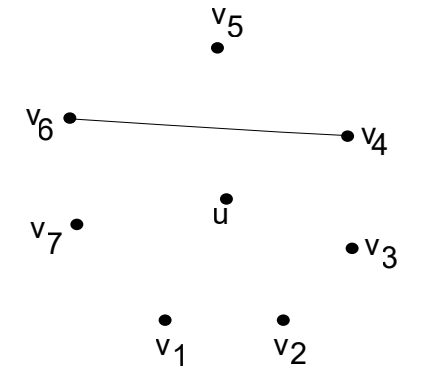


Рис. 3. $(x^F - x^C)\mathcal{H}$–переключение вида c)

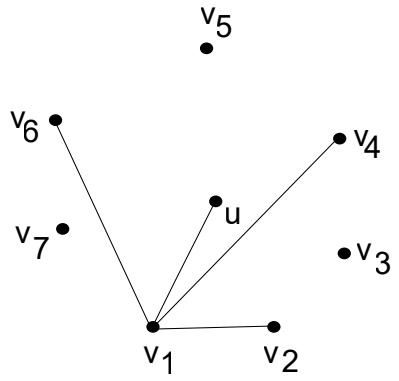


Рис. 4. $(x^F - x^C)\mathcal{H}$–переключение вида d)

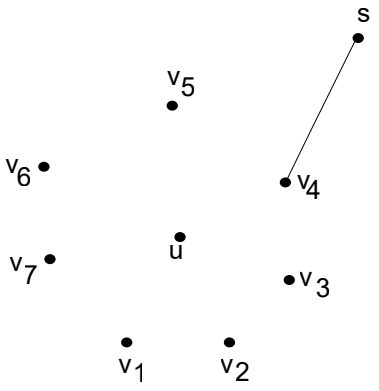


Рис. 5. $(x^F - x^C)\mathcal{H}$–переключение вида e)

Доказательство. Используя обозначения из формулировки утверждения 1, покажем, что $(x^F - x^C)\mathcal{H}$–базисом может являться любой луч uv звезды F. Зафиксируем луч uv_1. Условие (i) выполняется. Для проверки условия (ii) мы будем осуществлять переходы

$$\{uv_1\} \to \{uv_1\} \cup E_1 \to \{uv_1\} \cup E_1 \cup E_2 \to \ldots \to \{uv_1\} \cup E_1 \cup E_2 \cup \ldots E_t = E$$

так, что на каждом переходе s каждое ребро $e \in E_s$ получается из $\{uv_1\} \cup E_1 \cup \cup E_2 \cup \ldots E_{s-1}$ за один шаг с помощью $(x^F - x^C)\mathcal{H}$–переключений из утверждения 1. Эти переходы изобразим в виде таблицы, в левой колонке которой указано очередное множество E_s, а в правой — $(x^F - x^C)\mathcal{H}$–переключения, которому рёбра этого множества принадлежат (отметим, что в этой таблице важен порядок строк):

$$E_1 = \{st \mid s,t \notin V(F \cup C)\} \quad - \quad \{st\} \textit{ (утв. 1 a)};$$
$$E_2 = \{sv_i \mid s \notin V(F \cup C)\} \quad - \quad \{sv_i\} \textit{ (утв. 1 e)};$$
$$E_3 = \{v_iv_j \mid 2 \leqslant |i-j| \leqslant p-2\} \quad - \quad \{v_iv_j\} \textit{ (утв. 1 c)};$$

$$E_4 = \{su \mid s \notin V(F \cup C)\} \quad - \quad \{sv_1, sv_3, \ldots, sv_{p-2}, su\} \text{ (утв. 1 b)}.$$

Остаётся построить ребра множества $EF \cup EC$. Мы будем строить их в последовательности $v_1v_2, uv_2, v_2v_3, uv_3, \ldots, v_{p-1}v_p, uv_p, v_pv_1$, используя п. d) из утверждения 1. Первые четыре ребра в этой последовательности образуют множества E_5, E_6, E_7 и E_8:

$$E_5 = \{v_1v_2\} \quad - \quad \{uv_1, v_1v_2, v_1v_4, \ldots, v_1v_{p-2}\} \text{ (утв. 1 d)};$$

$$E_6 = \{uv_2\} \quad - \quad \{uv_2, v_1v_2, v_2v_{p-1}, v_2v_{p-3}, \ldots, v_2v_4\} \text{ (утв. 1 d)};$$

$$E_7 = \{v_2v_3\} \quad - \quad \{uv_2, v_2v_3, v_2v_5, \ldots, v_2v_{p-1}\} \text{ (утв 1 d)};$$

$$E_8 = \{uv_3\} \quad - \quad \{uv_3, v_2v_3, v_3v_p, v_3v_{p-2}, \ldots, v_3v_5\} \text{ (утв. 1 d)};$$

и так далее. Пары множеств E_5, E_6 и E_7, E_8 строятся с помощью одинакового приёма, который применяется далее для построения всех рёбер. Нетрудно заметить, что в итоге мы получим все рёбра множества E.

Теорема доказана. ■

Заключение

Основным результатом статьи является использование техники переключений, с помощью которой доказывается фасетность опорных неравенств к комбинаторным многогранникам. Аналогичные результаты, подтверждающие эффективность предлагаемой техники, можно найти в работах [7] и [2]. Полное теоретическое обоснование техники $b\mathcal{H}$–базисов содержится в [7].

Литература

1. Grotschel M., Padberg M.W. Polyhedral computations. The Travelling Salesman Problem / Ed. by E.L. Lawler etc. 1985.
2. Симанчёв Р.Ю., Уразова И.В. О гранях многогранника задачи аппроксимации графа // Дискрет. анализ и исследование операций. 2015. № 22(2). С. 86–101.
3. Simanchev R.Yu., Urazova I.V. On the Facets of Combinatorial Polytopes // DOOR-2016. LNCS, Springer, Heidelberg. 2016. No. 9869. P. 233–243 с.
4. Grotschel M., Wakabayashi Y. Facets of the clique partitioning polytope // Mathematical Programming. 1980. No. 47. P. 367–387.
5. Grotschel M., Wakabayashi Y. A cutting plane algorithm for a clustering problem // Mathematical Programming, (Series B). 1989. No. 45. P. 59–96.
6. Симанчёв Р.Ю. О неравенствах, порождающих фасеты комбинаторных многогранников // Дискретный анализ и исследование операций. 2017. № 24(4). С. 95–110.
7. Симанчев Р.Ю. О ранговых неравенствах, порождающих фасеты многогранника связных k–факторов // Дискретный анализ и исследование операций. 1996. Т. 3. С. 84–110.

$B\mathcal{H}$–BASES FOR A SOME CLASS OF FACET OF A CLIQUE PARTITIONING POLYTOPE

R.Yu. Simanchev
Ph.D. (Phys.-Math.), Associate Professor, e-mail: osiman@rambler.ru
P.V. Solov'eva
Master Student, e-mail: polinochka.chervonnykh@mail.ru

Dostoevsky Omsk State University, Omsk, Russia

Abstract. Let $K_n = (V, E)$ be a complete undirected n–vertex graph without loops and multiple edges. A spanning subgraph $H \subset K_n$ is called an M–graph if each of its connected components (possibly one-vertex) is a clique. We denote the family of all M–graphs in K_n by $\mathcal{H}$. This family is the set of feasible solutions of the clique partitioning problem, consisting in finding in the complete edge-weighted graph of an M–graph of minimal weight [2–4]. In the mentioned papers, the polyhedral properties [1] of this problem are considered. The classes of inequalities that generate the faces of the problem polytope are constructed, on the basis of which algorithms for branches and cuts are developed. In this paper, using the technique proposed in [7], we prove the facetness of inequalities of a special class with respect to the clique partitioning polytope.
The work was supported by the Russian Foundation for Basic Research (project 18-07-00599).

Keywords: polytope, facet, the clique partitioning problem.

Дата поступления в редакцию: 20.11.2018

Математические структуры и моделирование 2018. № 4(48). С. 35–45

УДК 519.852
DOI: 10.25513/2222-8772.2018.4.35-45

ЭВРИСТИКИ ДЛЯ ИДЕНТИФИКАЦИИ 1-ПАРАШЮТОВ В ЗАДАЧЕ АППРОКСИМАЦИИ ГРАФА

И.В. Уразова
к.ф.-м.н., доцент, e-mail: urazovainn@mail.ru

Омский государственный университет им. Ф.М. Достоевского, Омск, Россия

Аннотация. Задача аппроксимации графа заключается в нахождении в полном рёберно-взвешенном графе семейства попарно вершинно-непересекающихся клик минимального веса. В общем случае задача является NP–трудной. В работе (см. [1]) был предложен класс неравенств, опорных к многограннику данной задачи. Найдены условия, при которых построенные неравенства являются фасетными. При использовании этих неравенств в алгоритмах отсечений становится актуальной задача идентификации (Separation problem). В работе (см. [2]) было показано, что задача идентификации предложенных неравенств NP–трудна. В настоящей работе для идентификации отсечений разработана процедура локального поиска. Для анализа эффективности предлагаемых методов проведён вычислительный эксперимент.

Работа выполнена при поддержке РФФИ (проект 18-07-00599)

Ключевые слова: задача идентификации, полиэдр, фасета, аппроксимация графа.

Введение

Впервые задача аппроксимации графа была сформулирована Харари в 1955 году. В 60-70-х годах прошлого столетия в ряде работ были найдены нетривиальные классы графов, на которых задача является полиномиально разрешимой. В 1986 г. Криванек и Моравек рассмотрели задачу аппроксимации графа как частный случай задачи кластеризации деревьев и доказали, что она является NP–трудной. Систематическое изучение задачи аппроксимации графа началось в прошлом десятилетии, когда задача была переоткрыта под разными именами (Correlation clustering, Cluster editing) различными группами авторов. В частности была установлена NP–трудность различных её вариантов и для их решения предложены первые приближенные алгоритмы с гарантированной оценкой точности. Лучший из известных на сегодняшний день приближенный алгоритм для задачи аппроксимации графа гарантировано находит решение не более чем в 2.5 раза хуже оптимального.

Введём следующие обозначения и понятия. Для любого графа $D \subset K_n$ через VD и ED будем обозначать множества его вершин и рёбер соответственно. Для

ребра $e \in E$ будем также использовать запись uv, где u и v — вершины из V, инцидентные ребру e. Для $D \subseteq K_n$ и $u \in V$ через $\delta_D(u)$ обозначим множество рёбер графа D, инцидентных вершине u. Если $D = K_n$, то в этой записи индекс D будем опускать. Каждое множество рёбер $R \subset E$ индуцирует в K_n некоторый подграф T, в котором $ET = R$ и VT — множество вершин, инцидентных рёбрам из R. Там, где не возникает двусмысленности, граф, индуцированный множеством рёбер R, будем обозначать через R. Для подграфов D, F и K_n положим

$$D \cup F = ED \cup EF, D \cap F = ED \cap EF.$$

Пусть $\mathbf{K}_n = (V, E)$ — полный неориентированный граф без петель и кратных рёбер. Остовный подграф $H \subset \mathbf{K}_n$ называется M–графом, если каждая его компонента связности является кликой или одновершинным графом. Множество всех M–графов в $\mathbf{K}_n$ обозначим через $\mu(V)$.

Пусть $G \subset \mathbf{K}_n$ — некоторый априори заданный остовный подграф. Задача аппроксимации графа G заключается в нахождении M–графа H, минимизирующего на множестве $\mu(V)$ функционал

$$\rho_G(H) = |G \cup H| - |G \cap H|.$$

Иными словами, требуется найти такое множество попарно непересекающихся клик на V, которое как можно меньше (в рёберном смысле) отличается от графа G.

В работе рассматривается полиэдральная постановка задачи аппроксимации графа. Полиэдральный подход к решению экстремальных комбинаторных задач заключается в сопоставлении задаче специального многогранника, заданного как выпуклая оболочка векторов инциденций допустимых решений, и, как следствие, использование аппарата выпуклого анализа и целочисленного программирования. Особое место на этом пути занимает задача описания многогранника в виде множества решений системы линейных уравнений и неравенств. При наличии полного линейного описания многогранника экстремальная комбинаторная задача сводится к задаче линейного программирования (возможно с экспоненциальным числом ограничений), что нередко позволяет получить эффективные алгоритмы её решения.

Множество $P \subset R^E$ называется многогранником, если оно является выпуклой оболочкой конечного числа точек. Линейное неравенство $a^T x \leqslant a_0$ ($a, x \in R^E, a \neq 0, a_0 \in R$,"$T$" — знак транспонирования) называется правильным к многограннику P, оно не нарушается ни одной из точек могогранника. Правильное неравенство называется опорным к многограннику P, если оно выполняется для любой точки из P и существует, по крайней мере, одна точка из P, обращающая его в равенство. Всякое опорное к P неравенство порождает множество $\{x \in P | a^T x = a_0\}$, которое называется гранью многогранника P. Максимальные по включению грани многогранника называются фасетами. Ясно, что фасетой является та и только та грань, размерность которой на 1 меньше размерности самого многогранника. Опорное неравенство, порождающее фасету, назовём, соответственно, фасетным.

Полиэдром в R^E называется множество решений конечной системы линейных уравнений и неравенств с переменными x_e, $e \in E$, если оно ограничено. Согласно теореме Вейля–Минковского, для всякого многогранника существует совпадающий с ним в теоретико-множественном смысле полиэдр, и наоборот.

В [1, 4] было показано, что $(0,1)$–вектор $x \in R^E$ является вектором инциденций M–графа, если и только если он удовлетворяет системе

$$\begin{gathered} -x_{uv} + x_{uw} + x_{vw} \leqslant 1 \\ x_{uv} - x_{uw} + x_{vw} \leqslant 1 \\ x_{uv} + x_{uw} - x_{vw} \leqslant 1, \end{gathered} \tag{3}$$

где $u, v, w \in V$ — всевозможные тройки попарно различных вершин,

$$x_{uv} \geqslant 0, \text{ для всех } uv \in E. \tag{4}$$

Полиэдр, определяемый системой (3)–(4), обозначим через M_n. Таким образом, $P_n \subset M_n$, причём включение именно строгое (в [1] приведены примеры нецелочисленных вершин полиэдра M_n) и каждая целочисленная вершина M_n является вектором инциденций некоторого M–графа. Следуя работе [3], ограничения вида (3) будем называть треугольниками.

В [1] был введён и изучен новый класс опорных к P_n неравенств. Пусть $U = \{u_1, u_2, \ldots, u_k\}$ и $W = \{v_1, v_2, \ldots, v_p\}$ — непустые подмножества множества V, $U \cap W = \varnothing$, $k \geqslant 1$, $p \geqslant 2$. Через $T_i, i = 1, 2, \ldots, k$, обозначим звезду в K_n с центром в вершине u_i и лучами $u_i v_j, j = 1, 2, \ldots, p$ (см. рис. 1).

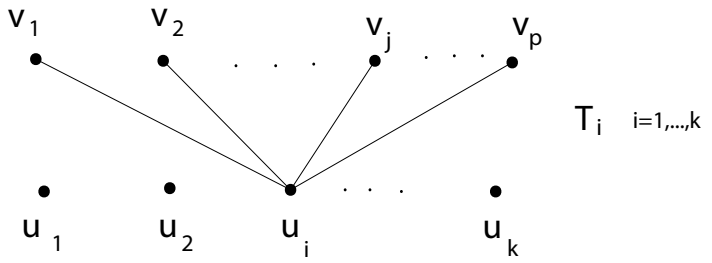


Рис. 1. Звезда в K_n с центром в вершине u_i и лучами $u_i v_j, j = 1, 2, \ldots, p$

Через K_p обозначим клику на множестве вершин W. Положим $T = \bigcup\limits_{i=1}^{k} T_i$. Граф $T \cup K_p$ называется k–парашютом (см. рис. 2).

С этим графом свяжем неравенство

$$x(ET) - x(EK_p) \leqslant \frac{k^2 + k}{2}.$$

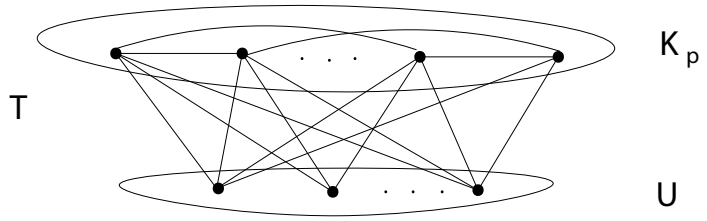


Рис. 2. k–парашют

В [1] было доказано, что такое неравенство, индуцированное k–парашютом $T \cup K_p$, является опорным к многограннику P_n тогда и только тогда, когда $p \geqslant k$, а фасетным — тогда и только тогда, когда $k = 1$. В этой связи особый интерес для нас будут представлять именно 1-парашюты (см. рис. 3).

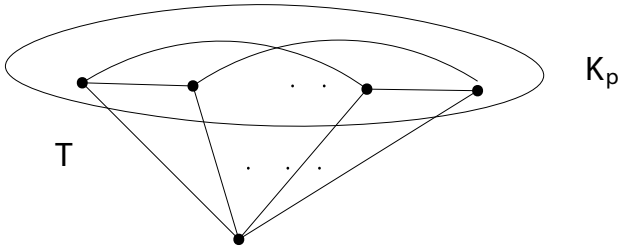


Рис. 3. 1-парашют

В [3, 4] был предложен класс опорных неравенств (2-дольных), отличающихся от k–парашютов тем, что на множестве $U = \{u_1, u_2, \ldots, u_k\}$ строится клика K_2 (см. рис. 4). Класс 2-дольных неравенств состоит из неравенств вида

$$x(ET) - x(EK_p \cup EK_2) \leqslant \min(p, k).$$

Кроме того, нетрудно заметить, что пересечением класса 2-дольных неравенств и класса k–парашютов является класс 1-парашютов.

Для заданного 1-парашюта $T \cup K_p$ вершину графа K_n, образующую множество U, будем иногда называть парашютистом, а клику K_p — куполом 1-парашюта.

При разработке процедур отсечения, использующих опорные неравенства как отсекающие плоскости, на передний план выходит задача идентификации (separation problem), которая заключается в следующем. Даны некоторый класс L неравенств, опорных к многограннику P, и точка $\bar{x} \in R^E$. Требуется найти

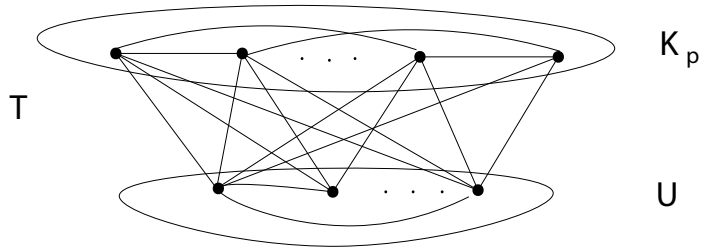


Рис. 4. 2-дольное неравенство

в классе L неравенство, строго отделяющее точку $\bar{x}$ от многогранника P, либо доказать, что в L такого неравенства нет. В 1982 году в работах Karp R.M. и Papadimitriu C.H. было показано, что при $NP \neq co - NP$ для полиэдра общего вида (не обязательно целочисленного) и класса неравенств, определяющих все фасеты выпуклой оболочки его целочисленных точек, не существует полиномиального алгоритма решения задачи идентификации. В связи с этим целесообразно рассматривать задачу идентификации применительно к конкретным классам неравенств относительно конкретных задач. Например, идентификация кликовых фасет для многогранника симметричной задачи коммивояжёра полиномиально разрешима, а идентификация гребневых неравенств и неравенств деревьев клик для той же задачи NP-трудна (см. также [2, 7]).

Очевидно, что поскольку число попарно различных троек вершин в K_n полиномиально по n, задача идентификации треугольников полиномиально разрешима. В [3] для поиска 2-дольного неравенства, отсекающего заданную точку, использовалась эвристическая процедура, поскольку сложностной статус задачи идентификации 2-дольных неравенств на тот момент был неизвестен. В 2001 году была показана NP–трудность задачи идентификации для этого класса. Однако важно иметь в виду следующее. Поскольку многогранник P_n целиком лежит в единичном кубе пространства R^E, то отсекаемую точку можно полагать лежащей в единичном кубе. Это несколько усиливает постановку задачи идентификации. Возможно ещё большее усиление. Поиск неравенства, отсекающего текущий нецелочисленный оптимум, начинается, как правило, с класса неравенств с полиномиально разрешимой задачей идентификации. Поэтому постановка задачи идентификации для класса 2-дольных неравенств может быть усилена до следующей: найдётся ли 2-дольное неравенство, отсекающее точку, удовлетворяющую всем треугольникам? В [2] дано новое строгое доказательство этого факта для класса 1-парашютов и, как следствие, 2-дольных неравенств и k–парашютов.

1. Идентификация 1-парашютов

Сформулируем задачу идентификации 1-парашютов относительно многогранника M–графов. При заданной точке $\bar{x} \in R^E$, $0 \leqslant \bar{x} \leqslant 1$, среди всех 1-парашютов $T \cup K_p$ графа K_n требуется найти такой, для которого $\sum\limits_{e \in ET} \bar{x}_e - \sum\limits_{e \in EK_p} \bar{x}_e > 1$. Эта задача может быть формализована в виде задачи целочисленного линейного программирования. Определим булевы переменные: $x_u \in \{0,1\}$, $u \in V$, — выбор парашютиста; $y_u \in \{0,1\}$, $u \in V$, — выбор вершин купола; $z_{uv} \in \{0,1\}$, $u,v \in V$, — выбор рёбер купола; $t_{uv} \in \{0,1\}$, $u,v \in V$, — выбор рёбер между парашютистом и куполом. Требуется максимизировать функцию

$$\frac{1}{2}\Big(\sum_{u,v \in V} \bar{x}_{uv} t_{uv} - \sum_{u,v \in V} \bar{x}_{uv} z_{uv}\Big) \tag{5}$$

при условиях:

$$z_{uv} \geqslant y_u + y_v - 1, \quad u,v \in V; \tag{6}$$

$$z_{uv} \leqslant y_u, \ z_{uv} \leqslant y_v, \quad u,v \in V; \tag{7}$$

$$t_{uv} \geqslant x_u + y_v - 1, \quad u,v \in V; \tag{8}$$

$$t_{uv} \leqslant x_u, \ t_{uv} \leqslant y_v, \quad u,v \in V; \tag{9}$$

$$x_u + y_u \leqslant 1, \quad u \in V; \tag{10}$$

$$\sum_{u \in V} x_u = 1; \tag{11}$$

$$\sum_{u \in V} y_u \geqslant 1; \tag{12}$$

$$x_u, y_u, z_{uv}, t_{uv} \in \{0,1\}, u,v \in V. \tag{13}$$

Ограничения (6), (7) определяют рёбра внутри купола и фактически гарантируют равенство $z_{uv} = y_u y_v$. Ограничения (8), (9) определяют рёбра между куполом и парашютистом и гарантируют равенство $t_{uv} = x_u y_v$. Ограничения (10) запрещают пересечение парашютиста с куполом. Ограничение (11) требует выбора ровно одного парашютиста. Ограничение (12) гарантирует непустой купол. В действительности для решения задачи идентификации достаточно проверить на совместность систему, образованную ограничениями (6)–(13) и неравенством $\sum\limits_{u,v \in V} \bar{x}_{uv} t_{uv} - \sum\limits_{u,v \in V} \bar{x}_{uv} z_{uv} > 1$, что в целом не облегчает задачу.

Как уже говорилось, задачу идентификации целесообразно рассматривать применительно к конкретным классам неравенств относительно конкретных задач. При такой стратегии, помимо точных процедур решения, часто используются эвристики (см., например, [3]). Так как задача идентификации 1-парашютов NP-трудна, мы разработали эвристическую процедуру для её решения.

Итак, рассматривается следующая задача. Дана точка $\bar{x} \in M_n \setminus P_n$. В графе K_n нужно найти такой 1-парашют $T_u \cup K_p$, где K_p, $p \geqslant 3$, — клика на множестве

вершин $\{v_1, v_2, \ldots, v_p\}$ и T_u — звезда с центром в вершине $u \notin \{v_1, v_2, \ldots, v_p\}$ и лучами uv_j, $j = 1, 2, \ldots, p$, что выполняется неравенство

$$\bar{x}(ET_u) - \bar{x}(EK_p) > 1.$$

Полагая величины $\bar{x}_e$ весами рёбер $e \in E$, а величину $\bar{x}(ET_u) - \bar{x}(EK_p)$ — весом 1-парашюта $T_u \cup K_p$, мы можем сформулировать рассматриваемую задачу идентификации как задачу поиска в полном рёберно-взвешенном графе 1-парашюта с весом больше 1.

Всякий 1-парашют $T_u \cup K_p$ является кликой порядка $p + 1$. В связи с этим, зафиксировав какое-либо множество вершин, мы можем простым перебором найти на этом множестве самый тяжёлый 1-парашют. Формально эта процедура выглядит так.

Процедура $\gamma(W)$. Эта процедура просматривает 1-парашюты на фиксированном множестве вершин $W = \{u_1, u_2, \ldots, u_l\} \subset V$, $l > 3$. Положим $\gamma(W) = 0$, T и K — графы с пустыми множествами вершин и рёбер.

Шаг i ($i = 1, 2, \ldots, l$). Пусть T_i — звезда с центром в вершине u_i и лучами u_iu_j, $j = 1, \ldots, i-1, i+1, \ldots l$, и K_i – клика на вершинах $u_1, \ldots, u_{i-1}, u_{i+1}, \ldots u_l$. Если $\bar{x}(ET_i) - \bar{x}(EK_i) > \gamma(W)$, то полагаем $\gamma(W) := \bar{x}(ET_i) - \bar{x}(EK_i)$, $T = T_i$ и $K = K_i$. Иначе оставляем всё неизменным. Переходим на шаг $(i + 1)$.

Если в результате этой процедуры получим $\gamma(W) > 1$, то $T \cup K$ — искомый 1-парашют на множестве вершин W. В процедуре $\gamma(W)$ не обязательно просматривать все вершины $u_1, u_2, \ldots, u_l$. Можно остановиться сразу, как только $\gamma(W) > 1$. Однако если размерность задачи относительно невелика, то просмотр лучше довести до конца, так как величину $\bar{x}(ET) - \bar{x}(EK) > \gamma$ можно в определённом смысле считать «глубиной» отсечения.

Вообще говоря, в результате применения процедуры γ к конкретному множеству W может получиться $\gamma(W) \leqslant 1$. В этой связи имеет смысл перейти к другому W и повторить процедуру. Мы рассматриваем три способа перехода к новому множеству вершин.

1) Множество W' получается из W удалением одной вершины $u \in W$, то есть $W' \subset W$ и $|W'| = |W| - 1$. Такой переход будем обозначать $W' = \varphi_1(W, u)$.

2) Множество W' получается из W добавлением одной вершины $u' \notin W$, то есть $W \subset W'$ и $|W'| = |W| + 1$. Такой переход будем обозначать $W' = \varphi_2(W, u')$.

3) Множество W' получается из W удалением одной вершины $u \in W$ и добавлением одной вершины $u' \notin W$, то есть $|W'| = |W|$ и $|W \cup W'| - |W \cap W'| = 2$. Для обозначения такого перехода будем использовать запись $W' = \varphi_3(W, u, u')$.

К случайно выбранному множеству W применяется каждый из указанных способов в рандомизированном режиме. А именно переход φ_1 осуществляется последовательным просмотром вершин множества W, вершина $u \in W$ выбирается с вероятностью p_1, запоминается такое W_1, что $\gamma(W_1) = \max\{\gamma(\varphi_1(W, u)), u \in W\}$; переход φ_2 — просмотр множества $V \setminus W$, вершина u' выбирается с вероятностью p_2, запоминается такое W_2, что $\gamma(W_2) = \max\{\gamma(\varphi_2(W, u')), u' \notin W\}$; переход φ_3 заключается в просмотре множества $W \times (V \setminus W)$, пара (u, u') выбирается с вероятностью p_3, запоминается

W_3, для которого $\gamma(W_3) = \max\{\gamma(\varphi_2(W, u, u')), u \in W, u' \notin W\}$. В качестве нового множества вершин W' выбирается то из множеств W_1, W_2 и W_3, на котором значение функции $\gamma(W)$ наибольшее.

Описанная процедура является одной итерацией эвристики, используемой нами для решения задачи идентификации 1-парашютов. Обозначим через $\varphi(W)$ множество вершин, которое получается из множества W в результате этой итерации. При построении отсекающих 1-парашютов итерации продолжаются то тех пор, пока не прекратится заметный рост функции γ. На рисунках 4 и 5 показан пример работы описанной процедуры для задач с $n = 100$.

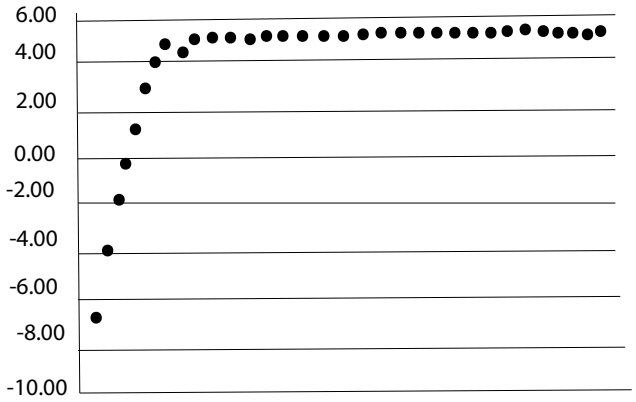


Рис. 5. Число вершин в клике 1-парашюта $|VK| = 15$, итераций 100, $\gamma(W) = 5$

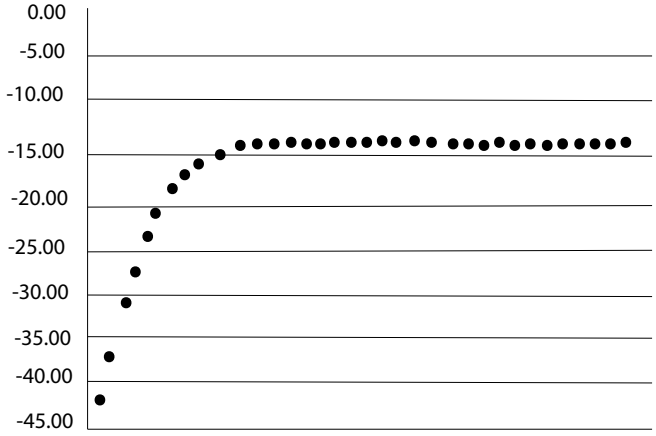


Рис. 6. Число вершин в клике 1-парашюта $|VK| = 30$, итераций 100, $\gamma(W) = -14$

В следующем параграфе показаны результаты вычислительного эксперимента, цель которого заключалась в проверке эффективности предложенной процедуры решения задачи идентификации для 1-парашютов. Следует отметить, что анализ результатов вычислительного эксперимента существенно осложняется отсутствием на сегодняшний день тестовых задач.

2. Вычислительный эксперимент

Алгоритм локального поиска 1-парашюта был реализован на языке Java и запускался на компьютере Intel Pentium (2,4 GHz). Сам эксперимент проводился по трём направлениям и был организован следующим образом. На первом этапе исследуется эвристика для построения 1-парашютов при одинаковых значениях параметров p_1, p_2 и p_3. Время было фиксированным и составляло 10 мин. Оценивалось значение целевой функции, полученное на заданных параметрах за 10 мин. Целью этого этапа было выделить, при каких параметрах рост значений целевой функции будет быстрее. В эксперименте рассматривались задачи размерности от 20 до 100, всего было по 10 задач на каждую размерность. Ниже представлен фрагмент таблицы 1 с результатами эксперимента. В первом столбце указано имя файла с решаемой задачей, в последнем — количество рёбер аппроксимируемого графа, в остальных столбцах стоят значения целевой функции, полученные за 10 мин при заданной вероятности.

Таблица 1. Результаты первого эксперимента

	$p_1 = p_2 = p_3$					
Задача	0.25	0.40	0.50	0.60	0.75	EG
$f1-40$	216.64	216.84	216.89	218.49	218.23	385
$f2-40$	223.81	225.28	223.54	222.70	225.35	394
$f3-40$	227.33	225.82	225.86	228.16	226.55	400
$f4-40$	217.30	218.20	218.27	219.05	219.42	379
$f5-40$	227.38	225.50	225.14	225.78	227.38	400
$f1-50$	185.78	180.30	178.89	179.01	190.46	313
$f2-50$	177.15	175.05	174.32	177.01	177.19	306

Из таблицы 1 видно, что при выбранной вероятности 0.6 — 0.75 рост значения целевой функции за фиксированное время происходит быстрее. Часто найденных с помощью локального поиска неравенств 1-парашютов оказывается очень много, и добавление их всех в качестве отсечений может привести к быстрому росту размерности задачи и, как следствие, к увеличению времени на каждой итерации.

Целью данного эксперимента было выяснить, сколько лучше добавить найденных 1-парашютов, чтобы рост значений функции происходил быстрее. Вероятности были взяты одинаковыми и равными, а именно $p_1 = p_2 = p_3 = 0.6$. Из всех найденных 1-парашютов выбирались неравенства с наибольшим значением $|a^T\bar{x} - 1|$. Здесь $\bar{x}$ — решение задачи ЛП, полученное на очередной итерации,

$a^T x \leqslant 1$ — неравенство 1-парашюта, найденное с помощью эвристики. Время работы программы было фиксированным и составляло 10 минут. На каждой итерации добавлялись найденные неравенства 1-парашютов в количестве 1, $n/10$, $n/5$ и $n/2$, где n — это число вершин исходного графа G. В эксперименте рассматривались задачи размерности от 20 до 100, всего было по 10 задач на каждую размерность. Ниже представлен фрагмент таблицы 2 с результатами эксперимента.

Таблица 2. Результаты второго эксперимента

	$q=1$	$n/10$		$n/5$		$n/2$		
Задача	f	$ax-1$	f	$ax-1$	f	$ax-1$	f	EG
$f1-40$	210.00	2.50	216.30	2.50	217.53	2.68	222.38	385
$f2-40$	221.07	2.50	227.50	2.50	228.45	2.21	234.69	394
$f3-40$	221.75	2.50	225.81	2.50	230.39	2.00	234.15	400
$f4-40$	211.06	2.50	217.27	2.50	220.41	2.19	225.44	379
$f5-40$	218.92	2.50	220.00	2.50	229.15	2.24	232.18	400
$f1-50$	174.93	3.89	173.99	4.88	175.42	2.50	177.04	313
$f2-50$	171.49	4.62	172.98	4.50	172.24	4.00	173.56	306
$f3-50$	180.50	3.50	178.70	2.50	181.25	3.50	183.58	325
$f4-50$	173.84	4.58	175.96	3.47	177.81	4.79	177.72	312

Из таблицы 2 видно, что если на каждой итерации добавлять $n/2$ найденных 1-парашютов, то за заданное время рост значений функции происходил быстрее.

Последний эксперимент повторяет предыдущий за исключением одного шага. Вместо значения $|a^T\bar{x}-1|$ рассматривалась величина $d=\frac{|a^T\bar{x}-1|}{|a|}$. Результаты представлены в таблице 3.

Таблица 3. Результаты третьего эксперимента

	$q=1$	$n/10$		$n/5$		$n/2$		
Задача	f	d	f	d	f	d	f	EG
$f1-40$	214.27	0.30	222.32	0.30	225.04	0.40	231.78	385
$f2-40$	225.68	0.30	229.33	0.29	240.59	0.5	244.23	394
$f3-40$	225.25	0.36	232.12	0.21	236.52	0.15	241.29	400
$f4-40$	216.18	0.32	224.61	0.26	230.36	0.21	234.73	379
$f5-40$	222.00	0.32	229.82	0.27	234.86	0.14	240.82	400
$f1-50$	182.18	0.45	186.14	0.37	185.23	0.43	186.43	313

Из таблицы 3 видно, что если выбирать найденные 1-парашюты по значению $d=\frac{|a^T\bar{x}-1|}{|a|}$ при параметрах $p_1=p_2=p_3=0.6$ и $n/2$, то значения целевой функции выше, чем при $|a^T\bar{x}-1|$.

Заключение

В заключении автор выражает благодарность Р.Ю. Симанчеву и Ю.А. Кочетову за ценные идеи и замечания в работе над данной статьёй.

Литература

1. Симанчев Р.Ю., Уразова И.В. О гранях многогранника задачи аппроксимации графов // Дискрет. анализ и исследование операций 2015. № 22(2). С. 86–101.
2. Simanchev R.Yu., Urazova I.V. Separation Problem for k-parashutes // Proc. DOOR 2016. Vladivostok, Russia, September 19-23. CEUR-WS. 2016. No. 1623. P. 109–114. URL: `http://ceur-ws.org/Vol-1623/paperco16.pdf` (дата обращения: 10.11.2018).
3. Grotschel M., Wakabayashi Y. A cutting plane algorithm for a clustering problem // Mathematical Programming, (Series B). 1989. No. 45. P. 59–96.
4. Grotschel M., Wakabayashi Y. Facets of the clique partitioning polytope // Mathematical Programming. 1990. No. 47. P. 367–387.
5. Simanchev R.Yu., Urazova I.V. Cutting Planes Algorithm for the Connected k-factor Problem Using the Facet Inequalities. Petrovac, Montenegro, October 2-7. CEUR-WS. 2017. URL: `http://ceur-ws.org/Vol-1987/paper74.pdf` (дата обращения: 10.11.2018).
6. Schrijver A. Combinatorial Optimization. Polyhedra and Efficiency. Springer. 2003. Vol. A.
7. Симанчёв Р.Ю. О неравенствах, порождающих фасеты комбинаторных многогранников // Дискретный анализ и исследование операций. 2017. № 24(4). С. 95–110.

HEURISTICS FOR THE SEPARATION PROBLEM OF 1-PARACHUTES IN THE GRAPH APPROXIMATION PROBLEM

I.V. Urazova
Ph.D. (Phys.-Math.), Associate Professor, e-mail: urazovainn@mail.ru

Dostoevsky Omsk State University, Omsk, Russia

Abstract. In work (see [1]) a class of inequalities supporting to the polyhedron of the given problem was proposed. Conditions are found under which the constructed inequalities are faceted. When these inequalities are used in cutting algorithms, the Separation problem becomes an actual problem. In the paper (see [2]) it was shown that the problem of identifying the proposed inequalities is NP–hard. In this paper, a local search procedure has been developed to identify clipping. To analysis the effectiveness of the proposed methods, a computer experiment was carried out.
This work was supported by the Russian Foundation for Basic Research (project 18-07-00599)

Keywords: separation problem, plyhedra, facet, approximation problem.

Дата поступления в редакцию: 20.11.2018

Mathematical Structures and Modeling
2018. N. 4(48). PP. 46–52

UDC 519.2
DOI: 10.25513/2222-8772.2018.4.46-52

IN THE DISCRETE CASE, AVERAGING CANNOT BE CONSISTENT

Olga Kosheleva
Ph.D. (Phys.-Math.), Associate Professor, e-mail: olgak@utep.edu
Vladik Kreinovich
Ph.D. (Phys.-Math.), Professor, e-mail: vladik@utep.edu

University of Texas at El Paso, El Paso, Texas 79968, USA

Abstract. When we have two estimates of the same quantity, it is desirable to combine them into a single more accurate estimate. In the usual case of continuous quantities, a natural idea is to take the arithmetic average of the two estimates. If we have four estimates, then we can divide them into two pairs, average each pair, and then average the resulting averages. Arithmetic average is *consistent* in the sense that the result does not depend on how we divide the original four estimates into two pairs. For discrete quantities — e.g., quantities described by integers — the arithmetic average of two integers is not always an integer. In this case, we need to select one of the two integers closest to the average. In this paper, we show that no matter how we select — even if we allow probabilistic selection — the resulting averaging cannot be always consistent.

Keywords: averaging, processing estimates, consistency, discrete case.

1. Formulation of the Problem

Need for averaging. In many practical situations, we have two (or more) estimates x_1 and x_2 of the same quantity x. In such situations, it is desirable to combine the two estimates and come up with a single — hopefully more accurate – estimate $x_1 * x_2$ of this quantity.

What operation $*$ should be use? In geometric terms, the pair (x_1, x_2) can be naturally represented by a point in a 2-D plane. If the estimates were exact, we would have the exact same number x in both components of this pair, i.e., we would have the pair (x, x). It is therefore reasonable to look for the value x for which the corresponding pair (x, x) is the closest to the pair (x_1, x_2).

The distance between the 2-D points (x, x) and (x_1, x_2) is equal to

$$\sqrt{(x - x_1)^2 + (x - x_2)^2}.$$

Minimizing this distance is equivalent to minimizing its square

$$(x - x_1)^2 + (x - x_2)^2.$$

Differentiating this expression with respect to x and equating the derivative to 0, we conclude that $x = \dfrac{x_1 + x_2}{2}$. Such averaging is indeed one of the main ways to combine two estimates; see, e.g., [1,5].

Averaging is consistent. If we have four estimates x_1, x_2, x_3, and x_4, then a natural idea is:

- to divide them into two pairs; for example, we can divide into pairs (x_1, x_2) and (x_3, x_4);
- average values from each pair, coming up with combined estimates $x_1 * x_2$ and $x_3 * x_4$, and
- then average the resulting averages, combing up with the value

$$(x_1 * x_2) * (x_3 * x_4).$$

It is reasonable to require that the averaging operation is *consistent* in the sense that the result of this operation should not change if, on the first stage, we use a different division into two pairs, i.e., if

$$(x_1 * x_2) * (x_2 * x_4) = (x_1 * x_3) * (x_2 * x_4).$$

What if the corresponding quantity is discrete? Some physical quantities — like electric change — are discrete, in the sense that they can take only values $\ldots, -2e, -e, 0, e, 2e, \ldots$ proportional to some fixed value e. To make our discussion simpler, let us select this value e as a measurement unit. In this case, possible values of the quantity x are integers.

If x_1 and x_2 have the same parity, i.e., if they are either both odd or both event, then the arithmetic average $\overline{x} = \dfrac{x_1 + x_2}{2}$ is also an integer. However, if one of the estimates is even, and another is odd — e.g., if $x_1 = 0$ and $x_1 = 1$ — then the arithmetic average is no longer an integer. In this case, as one can easily see, we have two different integers x for which the square $(x - x_1)^2 + (x - x_2)^2$ of the distance is the smallest: the floor $\lfloor \overline{x} \rfloor$ and the ceiling $\lceil \overline{x} \rceil$ of the corresponding fraction x. For example, for $x_1 = 0$ and $x_2 = 1$, we have $\overline{x} = 0.5$, so $\lfloor \overline{x} \rfloor = 0$ and $\lceil \overline{x} \rceil = 1$.

Formulation of the problem. We would like to select, for very pair of integers (x_1, x_2), one of the two possible averages. A natural question is: can we select it in such a way that the resulting operation is consistent?

What we prove. In this paper, we prove that in the discrete case, averaging cannot be consistent.

2. Definitions and the Main Result

Definition 1. *We say that an operation $* : Z \times Z \to Z$ that maps pairs of integers into an integer is a* discrete-case averaging *if for every pair (x_1, x_2), the*

result $x = x_1 * x_2$ *minimizes the sum* $(x - x_1)^2 + (x - x_2)^2$*:*

$$(x_1 * x_2 - x_1)^2 + (x_1 * x_2 - x_2)^2 = \min_{x \in Z}((x - x_1)^2 + (x - x_2)^2).$$

Definition 2. *We say that a discrete-case averaging* $*$ *is* consistent *if for every four integers* x_1*,* x_2*,* x_3*, and* x_4*, we have*

$$(x_1 * x_2) * (x_2 * x_4) = (x_1 * x_3) * (x_2 * x_4).$$

Proposition 1. *No discrete-case averaging is consistent.*

Proof. Let us assume that $*$ is a consistent discrete-case averaging, and let us get a contradiction out of this assumption.

1°. By definition of a discrete-case averaging, the value $1 * 2$ should be equal either to 1 or to 2. Let us show that in both cases, consistency is violated for some values x_1, x_2, x_3, and x_4.

2°. Let us first consider the case when $1 * 2 = 1$. Let us prove that in this case, $2 * 3 = 2$.

Indeed, by definition of a discrete-case averaging, we have $2 * 3 = 2$ or $2 * 3 = 3$. However, if $2 * 3 = 3$, then, due to consistency, we have

$$(1 * 2) * (1 * 3) = (1 * 1) * (2 * 3).$$

We consider the case when $1 * 2 = 1$; by definition, $1 * 3 = 2$, thus the left-hand side of the above formula has the form $(1 * 2) * (1 * 3) = 1 * 2$, and we already know that $1 * 2 = 1$.

On the other hand, if $2 * 3 = 3$, then the right-hand side has the form

$$(1 * 1) * (2 * 3) = 1 * 3 = 2.$$

So, if $2 * 3 = 3$, then the left-hand side and the right-hand side are different — and hence, the averaging $*$ is not consistent. Since we assumed that $*$ is consistent, this means that $2 * 3$ cannot be equal to 3 — and thus, it must be equal to 2.

Then, due to consistency, we should also have

$$(1 * 2) * (2 * 3) = (1 * 3) * (2 * 2).$$

Here, $1 * 2 = 1$ and $2 * 3 = 2$, so the left-hand side of this equality takes the form $(1 * 2) * (2 * 3) = 1 * 2 = 1$.

On the other hand, here $1 * 3 = 2$ and $2 * 2 = 2$, hence the right-hand side takes the form $(1 * 3) * (2 * 2) = 2 * 2 = 2$. So, the left-hand side and right-hand side are different — and thus, the averaging is not consistent.

3°. To complete the proof, let us consider the remaining case when $1 * 2 = 2$. Let us prove that in this case, $0 * 1 = 1$.

Indeed, by definition of a discrete-case averaging, we have $0*1=0$ or $0*1=1$. However, if $0*1=0$, then, due to consistency, we have

$$(0*2)*(1*2)=(0*1)*(2*2).$$

We consider the case when $1*2=2$; by definition, $0*2=1$, thus the left-hand side of the above formula has the form $(0*2)*(1*2)=1*2$, and we already know that $1*2=2$.

On the other hand, if $0*1=0$, then the right-hand side has the form

$$(0*1)*(2*2)=0*2=1.$$

So, if $0*1=0$, then the left-hand side and the right-hand side are different — and hence, the averaging $*$ is not consistent. Since we assumed that $*$ is consistent, this means that $0*1$ cannot be equal to 0 — and thus, it must be equal to 1.

Then, due to consistency, we should also have

$$(1*2)*(0*1)=(1*1)*(0*2).$$

Here, $1*2=2$ and $0*1=1$, so the left-hand side of this equality takes the form $(1*2)*(0*1)=2*1=2$.

On the other hand, here $1*1=1$ and $0*2=1$, hence the right-hand side takes the form $(1*1)*(0*2)=1*1=1$. So, the left-hand side and right-hand side are different — and thus, the averaging is not consistent.

The proposition is proven.

3. What If We Allow Probabilistic Averaging

Idea. The above result is about a *deterministic* averaging, when to every pair (x_1,x_2), we assign a single value x_1*x_2. For example, for $x_1=0$ and $x_2=1$, we have two possible values x, for which the sum $(x-x_1)^2+(x-x_2)^2$ is the smallest — namely, the values 0 and 1, and we pick one of these values.

But if 0 and 1 are equally good, why not select each of them with some probability, e.g., with probability 1/2 each? In this case, we get a *probabilistic* averaging, for which, for each x_1 and x_2, the value x_1*x_2 is a random variable.

Natural question. Will the resulting probabilistic averaging be consistent — in the sense that for every x_1, x_2, x_3, and x_4, the random variables $(x_1*x_2)*(x_2*x_4)$ and $(x_1*x_3)*(x_2*x_4)$ have the same distribution?

What we prove. We prove that the answer is still "no" — but at least the above two random variables have the same mean.

Definition 3. *By a* probabilistic averaging*, we mean an operation $*$ that assigns, to every pair of integers (x_1,x_2), the following random variable:*

- *when the sum x_1+x_2 is even, the random variable x_1*x_2 is equal to $\overline{x}\stackrel{\text{def}}{=}\dfrac{x_1+x_2}{2}$ with probability 1;*

- *when the sum $x_1 + x_2$ is odd, the random variable is equal either to $\lfloor \overline{x} \rfloor$ or to $\lceil \overline{x} \rceil$, with some probability.*

Definition 4. *We say that a probabilistic averaging is* consistent *if for every x_1, x_2, x_3, and x_4, the random variables*

$$(x_1 * x_2) * (x_3 * x_4) \text{ and } (x_1 * x_3) * (x_2 * x_4)$$

have the same distribution, where different $*$ operations are assumed to be independent.

Definition 5. *We say that a probabilistic averaging is* weakly consistent *if for every x_1, x_2, x_3, and x_4, the random variables $(x_1 * x_2) * (x_3 * x_4)$ and $(x_1 * x_3) * (x_2 * x_4)$ have the same mean.*

Proposition 2. *No probabilistic averaging is consistent.*

Proposition 3. *There exists a probabilistic averaging which is weakly consistent.*

Proof of Proposition 2. Let us assume that $*$ is a consistent probabilistic averaging, and let us get a contradiction out of this assumption.

1°. Let us first prove that for all pairs $(n, n+1)$, the probability p of selecting n as $n * (n+1)$ is equal to either 0, or 0.5, or 1.

Indeed, by definition of consistency, we should have

$$(n * n) * ((n+1) * (n+1)) = (n * (n+1)) * (n * (n+1)).$$

The left-hand side is equal to $n * (n+1)$ and is, thus, equal to n with probability p.

In the right-hand side, each of the two terms $n * (n+1)$ is equal to n with probability p and to $n+1$ with the remaining probability $1-p$. Since different $*$-operations are assumed independent, we therefore have four possible cases:

- the first case is when both terms $n * (n+1)$ are equal to n; the probability of this case is $p \cdot p = p^2$;
- the second case is when the first term is equal to n and the second term is equal to $n+1$; the probability of this case is equal to $p \cdot (1-p)$;
- the third case is when the first term is equal to $n+1$ and the second term is equal to n; the probability of this case is equal to $(1-p) \cdot p$;
- finally, the fourth case is when both terms $n * (n+1)$ are equal to $n+1$; the probability of this case is $(1-p) \cdot (1-p) = (1-p)^2$.

In the first case, the value $(n * (n+1)) * (n * (n+1))$ is always equal to n, and, as we recall, this case occurs with probability p^2. In the second and third cases, the value n appears with probability p; thus, the overall probability of getting n in these cases is $2p \cdot (1-p) \cdot p = 2p^2 \cdot (1-p)$. In the fourth case, we always get $n+1$. So, the overall probability of getting n is

$$p^2 + 2p^2 \cdot (1-p) = p^2 + 2p^2 - 2p^3 = 3p^2 - 2p^3.$$

Since the operation $*$ is consistent, the probability of getting n on both sides should be equal, so we must get $p = 3p^2 - 2p^3$. The first possibility to get this

equality is to have $p = 0$. If $p \neq 0$, then we can divide both sides by p and get $1 = 3p - 2 \cdot 2p^2$, i.e., a quadratic equation $2p^2 - 3p + 1 = 0$, whose solutions are $p = 0.5$ and $p = 1$.

2°. From Part 1 of this proof, it follows that the probability p_{12} of getting 1 as a result of $1 * 2$ is either 0, or 0.5, or 1. Let us first consider the case when $p_{12} > 0$.

In this case, let us consider another consistency requirement:

$$(1 * 2) * (1 * 3) = (1 * 1) * (2 * 3).$$

In the left-hand side, $1 * 2$ is equal to 1 with probability $p_{12} > 0$, and to 2 with the remaining probability $1 - p_{12}$. Here, $1 * 3 = 2$, so $(1 * 2) * (1 * 3)$ is equal to $1 * 2$ with probability p_{12} and to $2 * 2$ with probability $1 - p_{12}$. In the first case, we get 1 in p_{12} of the cases, so the overall probability that the left-hand side is 1 is equal to p_{12}^2.

In the right-hand side, $1 * 1 = 1$, and $2 * 3$ is equal to 2 with some probability p_{23} and to 3 with the remaining probability $1 - p_{23}$. Thus, the right-hand side is equal to $1 * 2$ with probability p_{23} and to $1 * 3 = 2$ with probability $1 - p_{23}$. In the first case, we get 1 in p_{12} of the cases, so the overall probability that the right-hand side is 1 is equal to $p_{12} \cdot p_{23}$.

Due to consistency, the probability that the left-hand side is 1 and that the right-hand side is 1 should be the same, so we get $p_{12}^2 = p_{12} \cdot p_{23}$. Since $p_{12} > 0$, we can conclude that $p_{12} = p_{23}$.

Now, let us consider yet another particular case of consistency:

$$(1 * 2) * (2 * 3) = (1 * 3) * (2 * 2).$$

The right-hand side is always equal to $2 * 2 = 2$, while in the left-hand side, we have $1 * 2 = 1$ with probability p_{12}, $2 * 3 = 2$ with probability $p_{23} = p_{12}$ and thus, $(1 * 2) * (2 * 3) = 1 * 2$ with probability p_{12}^2. Out of these cases, we get $(1 * 2) * (2 * 3) = 1$ with probability $p_{12} \cdot p_{12}^2 > 0$.

So, in the right-hand side, we never get 1, but in the left-hand side, we get 1 with positive probability — which contradicts to the consistency assumption.

3°. Thus, the case $p_{12} > 0$ is impossible, and so, we always have $1 * 2 = 2$.

In this case, consistency implies that $(1 * 2) * (0 * 2) = (0 * 1) * (2 * 2)$. Here, $1 * 2 = 2$ and $0 * 2 = 1$, and thus, the left-hand side is equal to $2 * 1 = 2$.

The value $0 * 1$ is equal to 0 with some probability p_{01} and to 1 with the remaining probability $1 - p_{01}$. Since $2 * 2 = 2$, the right-hand side is equal to $0 * 2 = 1$ with probability p_{01} and to $1 * 2 = 2$ with probability $1 - p_{01}$. The left-hand side is always equal to 2, hence the right-hand side cannot be equal to 1, and so $p_{01} = 0$.

Thus, we always have $0 * 1 = 1$. In this case, we can use another particular case of consistency: $(1 * 2) * (0 * 1) = (1 * 1) * (0 * 2)$. Here, since $1 * 2 = 2$ and $0 * 1 = 1$, the left-hand side is equal to $2 * 1 = 2$, while the right-hand side is equal to $1 * 1 = 1$ — a contradiction.

Thus, the proposition is proven.

Proof of Proposition 3. One can easily check that, as the desired probabilistic averaging, we can take the averaging in which, for each pair with non-integer $\overline{x}$, we return both the floor and the ceiling of $\overline{x}$ with equal probability 1/2. In this case, the mean is simply the usual arithmetic average, and we know that the arithmetic average is consistent.

Acknowledgments

This work was supported in part by the US National Science Foundation grant HRD-1242122 (Cyber-ShARE Center of Excellence).

References

1. Rabinovich S.G. Measurement Errors and Uncertainty: Theory and Practice. Springer Verlag, New York, 2005.
2. Sheskin D.J. Handbook of Parametric and Nonparametric Statistical Procedures. Chapman and Hall / CRC, Boca Raton, Florida, 2011.

В ДИСКРЕТНОМ СЛУЧАЕ УСРЕДНЕНИЕ НЕ МОЖЕТ БЫТЬ СОСТОЯТЕЛЬНЫМ

О. Кошелева
к.ф.-м.н., доцент, e-mail: olgak@utep.edu
В. Крейнович
к.ф.-м.н., профессор, e-mail: vladik@utep.edu

Техасский университет в Эль Пасо, США

Аннотация. Когда мы имеем две оценки одной и той же величины, желательно объединить их в одну более точную оценку. В обычном случае непрерывных величин естественной идеей является вычисление среднего арифметического двух оценок. Если мы имеем четыре оценки, то мы можем разделить их на две пары, усреднить каждую пару, а затем усреднить полученные средние значения. Среднее арифметическое *состоятельно* в том смысле, что результат не зависит от того, как мы разделим исходные четыре оценки на две пары. Для дискретных величин (например, величин, описываемых целыми числами) среднее арифметическое двух целых чисел не всегда является целым числом. В этом случае нам нужно выбрать одно из двух целых чисел, ближайших к среднему. В этой статье мы показываем, что независимо от того, как мы выбираем (даже если мы допустим вероятностный выбор), полученное усреднение не может быть всегда состоятельным.

Ключевые слова: усреднение, обработка оценок, состоятельность, дискретный случай.

Дата поступления в редакцию: 13.10.2018

Mathematical Structures and Modeling
2018. N. 4(48). PP. 53–60

UDC 510.52
DOI: 10.25513/2222-8772.2018.4.53-60

ALL MAXIMALLY COMPLEX PROBLEMS ALLOW SIMPLIFYING DIVIDE-AND-CONQUER APPROACH: INTUITIVE EXPLANATION OF A SOMEWHAT COUNTERINTUITIVE LADNER'S RESULT

Olga Kosheleva
Ph.D. (Phys.-Math.), Associate Professor, e-mail: olgak@utep.edu
Vladik Kreinovich
Ph.D. (Phys.-Math.), Professor, e-mail: vladik@utep.edu

University of Texas at El Paso, El Paso, Texas 79968, USA

Abstract. Ladner's 1975 result says that any NP-complete problem — i.e., in effect, any maximally complex problem — can be reduced to solving two easier problems. This result sounds counterintuitive: if a problem is maximally complex, how can it be reduced to simpler ones? In this paper, we provide an intuitive explanation for this result. Our main argument is that since complexity and easiness-to-divide are not perfectly correlated, it is natural to expect that a maximally complex problem is not maximally difficult to divide. Our related argument is that — as this result shows — NP-completeness is a sufficient but not a necessary condition for a problem to be maximally complex; how to come up with a more adequate notion of complexity is still an open problem.

Keywords: NP-complete, divide-and-conquer approach, Ladner's result, maximally complex problems.

1. Formulation of the Problem

Ladner's result: a brief description and why it is counterintuitive. Ladner's 1975 result (see, e.g., [1, 3]) says that any NP-complete problem — i.e., in effect, any maximally complex problem — can be reduced to solving two easier problems.

This result sounds counterintuitive: if a problem is maximally complex, how can it be reduced to simpler ones?

What we do in this paper. In this paper, we provide an intuitive explanation for this result.

To provide this explanation, we first need to recall what is NP-completeness and what exactly is Lander's result. This will be done in the remaining part of this section. The next section will contain our explanation.

Which algorithms are feasible: a brief reminder. The definition of NP-completeness is based on the notion of a feasible algorithm. Thus, in order to

explain what is NP-completeness, we first need to explain what is a feasible algorithm.

This notion formalizes the intuitive idea that while some algorithms are practically feasible, other algorithms require so much computation time that they are not practically possible. For example:

- an algorithm that requires computation time n^2, where n is the length of the input, is usually practical, while
- an algorithm that require computation time 2^n is not — since even for reasonable-size inputs $n \approx 500$, the resulting computation time would exceed the lifetime of the Universe.

Usually:

- polynomial-time algorithms — i.e., algorithms A whose running time $t_A(x)$ on each inputs x do not exceed some polynomial $P(n)$ of the length $n = \mathrm{len}(x)$ of the input – are feasible, while
- algorithms for which $t_A^w(n) \stackrel{\mathrm{def}}{=} \max\limits_{x:\mathrm{len}(x)\leqslant n} t_A(x)$ is not bounded by any polynomial are not feasible.

Because of this, usually, an algorithm is defined to be feasible if it is polynomial-time; see, e.g., [2,4].

Comment. It is well known that this definition is not perfect; e.g.:

- an algorithm that takes time $t^w(n) = 10^{500} \cdot n$ is polynomial-time but not practically feasible, while
- an algorithm that takes time $\exp(10^{-20} \cdot n)$ is practically feasible but not polynomial-time.

However, this is the most adequate definition we have.

What is a "problem". Another important notion needed to explain what is NP-complete is the notion of the class NP. This notion comes from the attempt to formally describe the intuitive idea of a (general) problem.

In all practical situations, when we formulate a problem, we expect that there is a clear and feasible way to check whether a given candidate for a solution is indeed what we want.

For example, in mathematics, the main activity is proving theorems. Once we have a formulation,

- finding a proof is difficult — it may take hundreds of years for the whole mathematical community — but
- once a detailed proof is presented, it is relative easy to check step-by-step that the proof is correct: e.g., computer programs for checking proofs were available already in the 1960s, when computers were much slower.

Similarly, in physics,

- finding a law that explains all the observations may be difficult, but
- once the explaining formula is found, checking that all the observations are consistent with this formula is straightforward.

In engineering,

- it is sometimes difficult to find a design that satisfies the given specifications — e.g., designing a compact antenna for a smart phone requires complex

computations — but

- once a design is presented, it is usually straightforward to check that this design satisfies all the desired specifications.

In all these cases, we are given some information x, and we want to find some object y that satisfies a feasible property $C(x, y)$.

In all these cases, an additional requirement is that the length of y should be feasible, i.e., similarly to time, that the length of y not exceed some polynomial of the length of x: $\text{len}(y) \leqslant P_\ell(\text{len}(x))$. Indeed:

- In mathematics, if a proof is too long, it is not possible to check it.
- In physics, if a formula is too complex, it is worthless: we could as well use, e.g., piece-wise linear interpolation of the experimental data.
- In engineering, if the design is too complicated, it is not feasible to complement, etc.

In all these cases, we have a feasible algorithm $C(x, y)$ and we have a polynomial $P_\ell(n)$. The problem is: given x, find y such that $C(x, y)$ and

$$\text{len}(y) \leqslant P_\ell(\text{len}(x)).$$

Such a solution y is not always possible. So, before we solve the problem of actually finding y, we need to check whether such a y exists — or, in set-theoretic terms, whether x belongs to the set S of such x's for which the corresponding y exists.

The class of all such checking problems is known as NP, for Non-deterministic Polynomial. This name came from the fact that in all such problem,

- once we guessed a solution y,
- we can check, feasibly (i.e., in polynomial time) whether this guess is indeed a solution.

In other words, we can solve this problem in polynomial time if, in addition to computations, we allow guesses — such general "computations" are known as *non-deterministic*.

Comment. Not all the problems belong to the class NP. For example, if we are looking for an optimal solution, then there is no easy general way to check that a given candidate is indeed an optimal solution: that would require comparing it with all other possible solutions, and there are usually exponentially many of them.

However, in practice, what we really want is, e.g., a solution and a proof that this solution is optimal — it is always feasibly checkable, otherwise this problem is not practically useful.

Is NP equal to P? Some problems from the class NP can be solved by feasible (polynomial-time) algorithms. The class of all such problems is usually denoted by P.

Most computer scientists believe that there are problems that cannot be solved by feasible algorithms, i.e., that NP $\neq$ P. However, no one has been able to prove or disprove this, it is still an open problem. In this paper, we will operate under the assumption that NP $\neq$ P.

The notion of reduction. The last auxiliary notion that we need to explain what is NP-completeness is the notion of reduction.

Often, one general problem can be reduced to another one, in the sense that for each particular instance of the first problem we can feasible compute one (or several) instances of the second problem so that, based on the solution(s) to the second problem, we can feasibly compute the solution to the original problem.

For example, equations $a \cdot x + \dfrac{b}{x} = c$ can be reduced — by multiplying by x — to quadratic equations. Similarly, the general problem of quadratic equations can be reduced to solving cubic equations: to perform this reduction, it is sufficient to add the term $0 \cdot x^3$ to the left-hand side of the quadratic equation $a \cdot x^2 + b \cdot x + c = 0$.

The notion of NP-completeness. If a problem A can be reduced to a problem B, this means, intuitively, that the problem B is:

- either more complex than the problem A (as in the case of quadratic vs. cubic equations)
- or of the same complexity (as in the first example of reduction).

Thus, if we have a problem from the class NP to which every other problem from NP can be reduced, then such a problem is clearly maximally complex. Such problems are called *NP-complete*.

Ladner's result. In 1975, Richard E. Ladner proved that, if NP $\neq$ P, then every NP-complete set S can be represented as a union $S = S_1 \cup S_2$ of two disjoint set sets S_1 and S_2 none of which is NP-complete; see, e.g., [1, 3].

Why this result is somewhat counterintuitive. What Ladner's result shows is that we can reduce the original NP-complete problem of checking whether a given input belongs to the set S to two easier problems of checking whether $x \in S_1$ or $x \in S_2$. Once we have a positive answer to one of the two new problems, this means that $x \in S$ — and vice versa, if $x \in S$, this means that either $x \in S_1$ or $x \in S_2$.

But if the original problem S is NP-complete — i.e., maximally complex — how can it be reduced to simpler ones? That would make this problem easier.

What we do in this paper. In this paper, we provide an intuitive explanation of Ladner's result — an explanation that, hopefully, makes it less counterintuitive.

2. Our Main Argument

Let us reformulate out situation in general terms. To describe our explanation, let us reformulate the situation in general terms. We have two functions:

- a function that describes the complexity of a problem A; we will denote this function by $f(A)$, and
- a function that describe to what extent the given problem is difficult to divide into two easier-to-handle ones; we will denote the second function by $g(A)$.

Ladner's result is that problems that maximize $f(A)$ do not maximize $g(A)$ — although these two functions are clearly strongly correlated. How can we explain this?

The two functions are different. First, let us notice that the functions $f(A)$ and $g(A)$ are different — in the sense that they lead to different order between problems.

A classical example of this difference comes from *linear programming (LP)* — i.e., checking whether a given finite set of linear inequalities $\sum\limits_{j=1}^{n} a_{ij} \cdot x_j \leqslant b_i$, with known a_{ij} and b_i and unknown x_j, is consistent. This problem is known to be maximally difficult to parallelize – P-hard — thus, maximally difficult to divide into two easier-to-handle cases. However, this is *not* a maximally complex problem at all: it is actually in the class P; see, e.g., [4].

Thus, we have a clear case when the function $g(A)$ attains its maximum, while the value of the first function $f(A)$ is much much smaller than its maximum value. Thus, the functions $f(A)$ and $g(A)$ are indeed different.

When do maxima of two functions always coincide? In general, if we have two functions on a set, when do these two function attains their maxima on exactly the same elements?

In general, we are talking about conditional maxima, i.e., maxima on a subset $\mathcal{A}$ of the set U of all the elements. One can easily check that the two functions leads to the maximal elements on each subset $\mathcal{A}$ of the set U of all elements if and only if they generate the same order:

Proposition 1. *Let $f, g : U \to L$ be two functions from a set U to a partially ordered set L. Then, the following two conditions are equivalent to each other:*

- *for every subset $\mathcal{A} \subset U$, the sets of all f-largest and g-largest elements of $\mathcal{A}$ coincide:*

$$\{A \in \mathcal{A} : \forall B \in \mathcal{A}(f(B) \leqslant f(A))\} = \{A \in \mathcal{A} : \forall B \in \mathcal{A}(g(B) \leqslant g(A))\};$$

- *the functions f and g lead to the same order, i.e., for all $A, B \in U$, we have $f(A) \leqslant f(B)$ if and only if $g(A) \leqslant g(B)$.*

Proof. Clearly, if f and g lead to the same order, then the set of f-largest and g-largest elements coincide.

Vice versa, if the sets of f-largest and g-largest elements always coincide, then, for all A and B, we can consider the set $\mathcal{A} = \{A, B\}$. If $f(A) \leqslant f(B)$, this means that B is in the set of f-largest elements. Thus, B is also in the set of g-largest elements, and therefore, $g(A) \leqslant g(B)$.

Same argument shows that if $g(A) \leqslant g(B)$ then $f(A) \leqslant f(B)$. The proposition is proven.

This is related to Kendall's tau. If the two functions $f(A)$ and $g(A)$ were perfectly aligned, we would always have $f(A) \leqslant f(B)$ if and only if $g(A) \leqslant g(B)$. We know that our two functions are strongly related but not perfectly aligned. Thus, for pairs (A, B), the f- and g-orders sometimes differ.

In statistics, such a situation is well-known, it is described by *Kendall's tau* (see, e.g., [5]), which is defined as $\tau = 2r - 1$, where r is the proportion of all the pairs (A, B) for which f- and g-orders coincide.

The fact that for some pairs, f- and g-orders differ means that in our case, we have $r < 1$.

What is the probability that an f-largest element is also g-largest: an intuitive estimate and the resulting explanation of Ladner's result. An element A is f-largest if $f(B) \leqslant f(A)$ for all B. What is the probability that this element is also g-largest, i.e., that we have $g(B) \leqslant g(A)$ for all B?

For each B, the probability that $g(B) \leqslant g(A)$ is equal to r — by definition of the quantity r. We have no reason to believe that these is a positive or negative correlation between inequalities corresponding to different elements B. Thus, it is reasonable to assume that these inequalities are independent.

Due to the independence assumption, the probability that we have $g(B) \leqslant g(A)$ for all B can be estimated as the product of the corresponding probabilities, i.e., as r^N, where N denotes the overall number of elements in the set U.

In our case, N is large, so r^N is practically 0. Thus, the probability that an f-largest (i.e., NP-complete) problem is also g-largest (i.e., maximally difficult to reduce to two easier-to-solve problems) is close to 0. This is perfectly in line with Ladner's result.

Comment. Of course, our intuitive explanation does not explain the whole result: we explained, in effect, why it is reasonable to believe that "almost all" NP-complete problems are not maximally difficult to divide, but this does not explain that this is true for *all* NP-complete problems. This additional explanation comes from the fact that, by definition, all NP-complete problems are (kind of) equivalent to each other — in the sense that every two problems can be reduced to each other. Thus, it is reasonable to expect that what is true for one NP-complete problem is also true for all of them.

3. Our Related Argument

Ladner's result is somewhat counterintuitive: a brief reminder. In theoretical computer science, for problems from the class NP, NP-completeness is usually identified with being maximally complex. So, if a problem is not NP-complete, this means that it is not as complex as the NP-complete problems.

From this viewpoint, the possibility to reduce an NP-complete problem to two non-NP-complete ones is counterintuitive: maximally complex problem is reduced to two less complex ones. Intuitively, if maximally complex problem is reduced to two problems, at least one of them should be of the same complexity as the original problem.

Ladner's result becomes even more counterintuitive if we consider its consequence proven in [1] about a similar notion of function complexity: there is a case when the composition of two function is of maximal possible complexity, while both composed functions are easier to compute.

But is the usual identification correct? Ladner's result becomes counterintuitive if we identify maximally complex problems with NP-complete ones. But let us recall where this identification comes from. It comes from the fact that *if* the

problem is NP-complete — i.e., if every problem from the class NP can be reduced to this problem — then this problem is clearly of the largest possible complexity. This is clear and intuitive.

However, there are no intuitive arguments for saying that if the problem is not NP-complete, then it must be easier. In fact, intuitively, what Ladner's result shows is that at least one of the sets S_1 or S_2, while not NP-complete, is, from the intuitive viewpoint, almost as complex as NP-complete problems.

Resulting explanation and the resulting open problem. So, it is not Ladner's result itself that is counterintuitive, what makes this result counterintuitive is the identification of NP-completeness and maximal complexity. What this result shows is that this not-very-justified association is counterintuitive — while every NP-complete problem is maximally complex, this result clearly shows that there are problems which are intuitively maximally complex but not NP-complete.

It is thus desirable to come up with a more intuitive definition of maximally complex problems. In the corresponding definition, if a maximally complex set S is a union $S = S_1 \cup \ldots \cup S_m$ of finitely many sets from the class NP, at least one of the sets S_i should be maximally complex in the same sense. How to come up with such a definition is an open problem.

Acknowledgments

This work was supported in part by the US National Science Foundation grant HRD-1242122 (Cyber-ShARE Center of Excellence).

References

1. Hemaspaandra L.A., Spakowski H. Team diagonalization // ACM SIGACT News. 2018. Vol. 48, No. 3. P. 51–61.
2. Kreinovich V., Lakeyev A., Rohn J., Kahl P. Computational complexity and Feasibility of Data Processing and Interval Computations. Kluwer, Dordrecht, 1997.
3. Ladner R. On the structure of polynomial time reducibility // Journal of the ACM. 1975. Vol. 22, No. 1. P. 155–171.
4. Papadimitriou C.H. Computational Complexity. Pearson, Boston, Massachusetts, 1993.
5. Sheskin D.J. Handbook of Parametric and Nonparametric Statistical Procedures. Chapman and Hall / CRC, Boca Raton, Florida, 2011.

ВСЕ МАКСИМАЛЬНО СЛОЖНЫЕ ПРОБЛЕМЫ ДОПУСКАЮТ УПРОЩЕНИЕ ПРИНЦИПОМ «РАЗДЕЛЯЙ И ВЛАСТВУЙ»: ИНТУИТИВНОЕ ОБЪЯСНЕНИЕ НЕСКОЛЬКО ПРОТИВОРЕЧИВОГО РЕЗУЛЬТАТА ЛАДНЕРА

О. Кошелева
к.ф.-м.н., доцент, e-mail: olgak@utep.edu
В. Крейнович
к.ф.-м.н., профессор, e-mail: vladik@utep.edu

Техасский университет в Эль Паco, США

Аннотация. Результат Ладнера 1975 года говорит о том, что любая NP-полная проблема, то есть, по сути, любая максимально сложная проблема, может быть сведена к решению двух более простых задач. Этот результат звучит парадоксально: если проблема максимально сложна, как её можно свести к более простым? В этой статье мы даём интуитивное объяснение этому результату. Наш главный аргумент состоит в том, что, поскольку сложность проблемы и лёгкость разделения проблемы не вполне коррелированы, естественно ожидать, что максимально сложную задачу не обязательно настолько же сложно разделить. Наш связанный с этим аргумент состоит в том, что, как показывает этот результат, NP-полнота является достаточным, но не необходимым условием, чтобы задача была максимально сложной; как придумать более адекватное понятие сложности — по-прежнему остаётся открытой проблемой.

Ключевые слова: NP-полный, принцип «разделяй и властвуй», результат Ладнера, максимально сложные проблемы.

Дата поступления в редакцию: 17.10.2018

Математические структуры и моделирование
2018. № 4(48). С. 61–77

УДК 550.37 : 550.837
DOI: 10.25513/2222-8772.2018.4.61-77

ИССЛЕДОВАНИЯ ОСОБЕННОСТЕЙ СПЕКТРАЛЬНОЙ ПЛОТНОСТИ ДЛЯ ЭЛЕКТРОМАГНИТНОГО ПОЛЯ В ВЕРТИКАЛЬНО НЕОДНОРОДНОЙ ПРОВОДЯЩЕЙ СРЕДЕ

С.А. Терентьев
к.ф.-м.н., доцент, e-mail: s.a.terentyev@gmail.com
А.К. Гуц
д.ф.-м.н., профессор, e-mail: guts@omsu.ru

Омский государственный университет им. Ф.М. Достоевского, Омск, Россия

Аннотация. Электромагнитное поле в задачах электроразведки часто представляется в виде интегралов с быстроосцилирущим ядром. При вычислении этих интегралов на ЭВМ приходится деформировать контур интегрирования в плоскость комплексного переменного. В статье изучена допустимая область деформации контура интегрирования в случае неоднородной среды. Источник поля — вертикальный гармонический диполь.

Ключевые слова: электроразведка, электромагнитное поле вертикального электрического диполя, быстроосцилирующие интегралы, деформация контура, комплексная плоскость, отсутствие особых точек, область деформации.

Введение

При аналитическом решении задач электроразведки очень часто компоненты электромагнитного поля могут быть выражены в виде интеграла

$$\int_0^{+\infty} u(\lambda, p)K(\lambda, r)d\lambda,$$

где $K(\lambda)$ — быстро осциллирующее по λ ядро. При вычислении таких интегралов на ЭВМ приходится деформировать контур интегрирования в комплексную область $\mathbb{C}$ изменения переменной λ. В связи с этим необходимо прежде всего определить область $D_\lambda \subset \mathbb{C}$, в которой подынтегральная функция $u(\lambda, p)$ не имеет особенностей по λ. В случае горизонтально-слоистой среды, состоящей из слоев с плоскими поверхностями раздела, указанная задача была решена С.И. Смагиным [1]. Трёхслойная среда была изучена в [2, с. 116-119].

В данной работе рассматривается более общий случай неоднородной среды с параметрами σ (проводимость), μ (магнитная проницаемость) и ε (электрическая проницаемость), зависящими от глубины z залегания слоя. Источником поля является гармонический электрический вертикальный диполь.

1. Постановка задачи

Пусть имеется неоднородная среда, ограниченная плоскими поверхностями раздела $z = z_0, z = z_1$, где $z_0 < z_1$ (ось z направлена вверх, рис. 1). Параметры среды σ, μ, ε будем считать функциями переменной z. При $z > z_1$ и $z < z_0$ среда предполагается однородной с $\sigma = \sigma_i$, $\mu = \mu_i$.

Источник электромагнитного поля находится в точке с декартовыми координатами $(0, 0, 0)$.

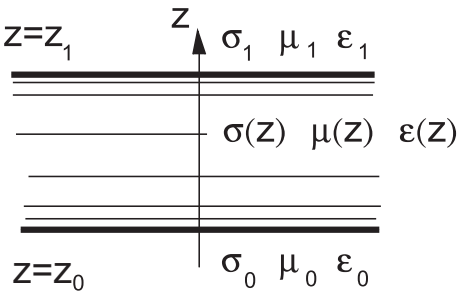


Рис. 1. Горизонтальная среда

На поверхностях раздела $z = z_i$ $(i = 0, 1)$ ставим граничные условия для электромагнитного поля $\mathbf{E}, \mathbf{H}$:

$$[\mathbf{E}_l]_{z=z_i} = 0, \qquad [\mathbf{H}_l]_{z=z_i} = 0$$

$$\left[\frac{\partial \mathbf{E}}{\partial l}\right]_{z=z_i} = 0, \quad \left[\frac{\partial \mathbf{H}}{\partial l}\right]_{z=z_i} = 0,$$

где индекс l означает горизонтальную составляющую поля, а $\partial/\partial l$ — производную по касательному к поверхности раздела направлению.

Если поле $(=\mathbf{E}$ или $\mathbf{H})$ задано интегралом

$$\int_0^{+\infty} K(x, y, \lambda) u(z, \lambda) d\lambda, \tag{1.1}$$

то будем K называть ядром интегрального оператора (1.1), а $u(z, \lambda)$ — плотностью.

Продолжим λ в комплексную плоскость

$$\mathbb{C} = \{\lambda = \lambda_x + i\lambda_y : \lambda_x, \lambda_y \in \mathbb{R}\}.$$

Область, лежащую в плоскости $\mathbb{C}$, в которой плотность $u(z, \lambda)$ не имеет особенностей по λ, будем обозначать через D_λ.

В этой статье мы определяем область D_λ для электромагнитного поля, создаваемого вертикальным гармоническим электрическим диполем.

2. Поле вертикального гармонического электрического диполя

В этом параграфе подробно изучим электромагнитное поле, создаваемое вертикальным гармоническим электрическим диполем, находящимся в неоднородной среде.

2.1. Исходные уравнения

Предположим для начала, что

$$\sigma, \mu, \varepsilon \in C^1(\mathbb{R}),$$

$$\sigma(z) \neq 0 \quad \text{при} \quad z \in \mathbb{R}.$$

Будем исходить из следующей системы уравнений Максвелла

$$rot\ \mathbf{E} = i\omega\mu\mathbf{H},$$

$$rot\ \mathbf{H} = (\sigma - i\omega\varepsilon)\mathbf{E} + \mathbf{j}, \tag{2.1}$$

$$div\ \varepsilon\mathbf{E} = \rho, \quad div\ \mathbf{B} = 0, \tag{2.2}$$

где $\mathbf{j}$ — сторонний электрический ток, ρ — электрические заряды, зависимость во времени определялась фазовым множителем $exp(-i\omega t)$, т. е. гармоничность источников означает следующую зависимость от времени

$$\mathbf{M} \to \mathbf{M}e^{-i\omega t}.$$

Полагаем

$$\mathbf{H} = rot\mathbf{A}, \tag{2.3}$$

$$\mathbf{E} = i\omega\mu\mathbf{A} - \nabla\varphi. \tag{2.4}$$

Подставляя (2.3), (2.4) в (2.2) получим

$$\nabla div\mathbf{A} - \triangle\mathbf{A} = (i\omega\mu\sigma + \omega^2\varepsilon\mu)\mathbf{A} - (\sigma - i\omega\varepsilon)\nabla\varphi + \mathbf{j}$$

или

$$\triangle\mathbf{A} + (i\omega\mu\sigma + \omega^2\varepsilon\mu)\mathbf{A} = -\mathbf{j} + \nabla[div\mathbf{A} + \varphi \cdot (\sigma - i\omega\varepsilon)] - \varphi \cdot \nabla(\sigma - i\omega\varepsilon).$$

Пусть

$$\varphi = -\frac{1}{\sigma - i\omega\varepsilon} div\mathbf{A}. \tag{2.5}$$

Тогда получаем уравнение для векторного потенциала

$$\triangle\mathbf{A} - \frac{\nabla(\sigma - i\omega\varepsilon)}{\sigma - i\omega\varepsilon} div\mathbf{A} + (i\omega\mu\sigma + \omega^2\varepsilon\mu)\mathbf{A} = -\mathbf{j}. \tag{2.6}$$

Будем рассматривать в качестве источника вертикальный электрический диполь, для которого

$$\mathbf{j} = (0, 0, \delta(x, y)\delta(z)),$$

т. е. предполагаем, что диполь находится в точке $(0, 0, 0)$ и, соответственно,

$$\mathbf{A} = (0, 0, A).$$

Решение уравнения (2.6) будем искать в виде

$$A = \frac{1}{2\pi} \int_0^{+\infty} \lambda I_0(\lambda r) u(z, \lambda) d\lambda, \tag{2.7}$$

где $r = \sqrt{x^2 + y^2}$, а I_0 — функция Бесселя первого рода нулевого порядка.

Подставив (2.7) в (2.6) и используя представление

$$\delta(x, y) = \frac{1}{2\pi} \int_0^{+\infty} \lambda I_0(\lambda r) d\lambda,$$

получим следующее дифференциальное уравнение

$$\frac{d^2u}{dz^2} - (\lambda^2 + k^2)u - \frac{(\sigma - i\omega\varepsilon)'}{\sigma - i\omega\varepsilon} \cdot \frac{du}{dz} = \delta(z), \tag{2.8}$$

где $k^2 = -(i\omega\mu\sigma + \omega^2\varepsilon\mu)$, а штрих $'$ означает дифференцирование по z.

Необходимо теперь указать соответствующие рассматриваемой задаче краевые условия. Непрерывность касательных составляющих полей $\mathbf{E}$ и $\mathbf{H}$, а также гладкость функций σ, ϵ, μ на поверхностях раздела $z = z_0$ и $z = z_1$ влекут условия:

$$[H_z]_{z=z_i} = 0, \qquad [E_z]_{z=z_i} = 0,$$
$$\left[\frac{\partial H_z}{\partial z}\right]_{z=z_i} = 0, \quad \left[\frac{\partial E_z}{\partial z}\right]_{z=z_i} = 0,$$

где квадратные скобки означают скачок

$$[f(z)]_{z=z_i} = f(z_i + 0) - f(z_i - 0).$$

Откуда

$$\left.\begin{array}{l} [u]_{z=z_i} = 0, \\ \left[\dfrac{du}{dz}\right]_{z=z_i} = 0, \quad i = 0, 1 \end{array}\right\}. \tag{2.9}$$

Кроме того, следует добавить условия излучения

$$\lim_{z\to\pm\infty} |u| = 0, \quad \lim_{z\to\pm\infty} \left|\frac{du}{dz}\right| = 0. \tag{2.10}$$

Краевую задачу (2.3), (2.9), (2.10) заменим эквивалентной краевой задачей

$$\frac{d^2u}{dz^2} - (\lambda^2 + k^2)u - \frac{(\sigma - i\omega\varepsilon)'}{\sigma - i\omega\varepsilon} \cdot \frac{du}{dz} = 0, \tag{2.11}$$

$$z \in (z_0, z_1) \setminus \{0\},$$

$$[u]_{z=z_i} = 0, \quad \left[\frac{du}{dz}\right]_{z=z_i} = 0, \tag{2.12}$$

$$\lim_{z\to\pm\infty} |u| = 0, \quad \lim_{z\to\pm\infty} \left|\frac{du}{dz}\right| = 0, \tag{2.13}$$

$$[u]_{z=0} = 0, \tag{2.14}$$

$$\left[\frac{du}{dz}\right]_{z=0} = 1, \tag{2.15}$$

в которой введена дополнительная (фиктивная) поверхность $z = 0$, содержащая источник. Задача (2.11)–(2.15) отличается от задачи (2.8)–(2.10) тем, что в ней отсутствуют сингулярные функции в качество коэффициентов.

Уравнение (2.11) перепишем в следующем виде

$$(\sigma - i\omega\varepsilon)\frac{d}{dz}\left(\frac{1}{\sigma - i\omega\varepsilon} \cdot \frac{du}{dz}\right) - (\lambda^2 + k^2)u = 0,$$

или, вводя $\lambda = \lambda_x + i\lambda_y$

$$\frac{d}{dz}\left[\frac{\sigma + i\omega\varepsilon}{\sigma^2 + \omega^2\varepsilon^2} \cdot \frac{du}{dz}\right] - \frac{\sigma + i\omega\varepsilon}{\sigma^2 + \omega^2\varepsilon^2}\left[\lambda_x^2 - \lambda_y^2 - \omega^2\varepsilon\mu + i(2\lambda_x\lambda_y - \omega\sigma\mu)\right]u = 0. \tag{2.16}$$

Пусть $u = u_1 + iu_2$.

Тогда уравнение (2.16) представляет собой систему двух дифференциальных уравнения

$$\frac{d}{dz}\left[\frac{\sigma}{\sigma^2 + \omega^2\varepsilon^2}\frac{du_1}{dz} - \frac{\omega\varepsilon}{\sigma^2 + \omega^2\varepsilon^2}\frac{du_2}{dz}\right] - \frac{\sigma(\lambda_x^2 - \lambda_y^2 - \omega^2\varepsilon\mu) - \omega\varepsilon(2\lambda_x\lambda_y - \sigma\mu)}{\sigma^2 + \omega^2\varepsilon^2}u_1 +$$
$$+ \frac{\omega\varepsilon(\lambda_x^2 - \lambda_y^2 - \omega^2\varepsilon\mu) + \sigma(2\lambda_x\lambda_y - \sigma\mu)}{\sigma^2 + \omega^2\varepsilon^2}u_2 = 0,$$
$$\frac{d}{dz}\left[\frac{\omega\varepsilon}{\sigma^2 + \omega^2\varepsilon^2}\frac{du_1}{dz} + \frac{\sigma}{\sigma^2 + \omega^2\varepsilon^2}\frac{du_2}{dz}\right] - \frac{\omega\varepsilon(\lambda_x^2 - \lambda_y^2 - \omega^2\varepsilon\mu) + \sigma(2\lambda_x\lambda_y - \sigma\mu)}{\sigma^2 + \omega^2\varepsilon^2}u_1 -$$
$$- \frac{\sigma(\lambda_x^2 - \lambda_y^2 - \omega^2\varepsilon\mu) - \omega\varepsilon(2\lambda_x\lambda_y - \sigma\mu)}{\sigma^2 + \omega^2\varepsilon^2}u_2 = 0.$$

Вводя матрицы

$$u = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix}$$

$$P = \begin{pmatrix} -\dfrac{\sigma}{\sigma^2+\omega^2\varepsilon^2} & \dfrac{\omega\varepsilon}{\sigma^2+\omega^2\varepsilon^2} \\ -\dfrac{\omega\varepsilon}{\sigma^2+\omega^2\varepsilon^2} & -\dfrac{\sigma}{\sigma^2+\omega^2\varepsilon^2} \end{pmatrix},$$

$$Q = \begin{pmatrix} \dfrac{\sigma(\lambda_x^2-\lambda_y^2-\omega^2\varepsilon\mu)-\omega\varepsilon(2\lambda_x\lambda_y-\sigma\mu)}{\sigma^2+\omega^2\varepsilon^2} & -\dfrac{\omega\varepsilon(\lambda_x^2-\lambda_y^2-\omega^2\varepsilon\mu)+\sigma(2\lambda_x\lambda_y-\sigma\mu)}{\sigma^2+\omega^2\varepsilon^2} \\ \dfrac{\omega\varepsilon(\lambda_x^2-\lambda_y^2-\omega^2\varepsilon\mu)+\sigma(2\lambda_x\lambda_y-\sigma\mu)}{\sigma^2+\omega^2\varepsilon^2} & \dfrac{\sigma(\lambda_x^2-\lambda_y^2-\omega^2\varepsilon\mu)-\omega\varepsilon(2\lambda_x\lambda_y-\sigma\mu)}{\sigma^2+\omega^2\varepsilon^2} \end{pmatrix},$$

можно изучаемую систему дифференциальных уравнений переписать в компактном виде

$$\frac{d}{dz}\left(P\frac{du}{dz}\right) + Qu = 0. \tag{2.17}$$

При этом краевые условия (2.12)–(2.15) примут следующий вид:

$$[u_k]_{z=z_i} = 0, \quad \left[\frac{du_k}{dz}\right]_{z=z_i} = 0, \quad k = 1,2; \quad i = 0,1, \tag{2.18}$$

$$\lim_{z\to\pm\infty}|u_k| = 0, \quad \lim_{z\to\pm\infty}\left|\frac{du_k}{dz}\right| = 0, \tag{2.19}$$

$$[u_k]_{z=0} = 0, \tag{2.20}$$

$$\left[\frac{du_1}{dz}\right]_{z=0} = 1, \quad \left[\frac{du_2}{dz}\right]_{z=0} = 0. \tag{2.21}$$

2.2. Пространство $W_2^{(1)}(\mathbb{R},\mathbb{R}^2)$

Пусть $u : \mathbb{R} \to \mathbb{R}^2$ — вектор-функция, компоненты которой $u_1(z), u_2(z)$, имеет первые производные по Соболеву, принадлежащие $L_2(\mathbb{R})$, т. е.

$$u_k \in W_2^{(1)}(\mathbb{R}), \ (k = 1,2).$$

Рассмотрим линейное пространство $W_2^{(1)}(\mathbb{R},\mathbb{R}^2)$ всех таких вектор-функций. Введём в $W_2^{(1)}(\mathbb{R},\mathbb{R}^2)$ скалярное произведение

$$(u,v)_{W_2^{(1)}} \equiv (u_1,v_1)_{W_2^{(1)}(\mathbb{R})} + (u_2,v_2)_{W_2^{(1)}(\mathbb{R})} \tag{2.22}$$

и норму

$$\| u \|_{W_2^{(1)}} = \sqrt{(u,u)_{W_2^{(1)}}},$$

т. е.

$$W_2^{(1)}(\mathbb{R},\mathbb{R}^2) = W_2^{(1)}(\mathbb{R}) \times W_2^{(1)}(\mathbb{R}^2)$$

и, следовательно, это банахово пространство.

Рассмотрим в $W_2^{(1)}(\mathbb{R},\mathbb{R}^2)$ билинейную форму

$$a(u,v)=\int\limits_{-\infty}^{+\infty}\frac{\sigma(z)}{\sigma^2(z)+\omega^2\varepsilon^2(z)}\sum_{i=1}^{2}\frac{du_i}{dz}\frac{dv_i}{dz}dz+$$

$$+\int\limits_{-\infty}^{+\infty}\frac{\sigma(z)[\lambda_x^2-\lambda_y^2-\omega^2\varepsilon(z)\mu(z)]-\omega\varepsilon(z)[2\lambda_x\lambda_y-\sigma(z)\mu(z)]}{\sigma^2(z)+\omega^2\varepsilon^2(z)}\sum_{i=1}^{2}u_iv_idz, \qquad (2.23)$$

где

$\omega=const>0$, $\sigma(z)>0$, $\mu(z)>0$, $\varepsilon(z)>0$ – гладкие функции,

$$(\lambda_x,\lambda_y)\in D_\lambda,$$

$$D_\lambda=\{(\lambda_x,\lambda_y)\in\mathbb{R}^2:\sigma[\lambda_x^2-\lambda_y^2-\omega^2\varepsilon\mu]>\omega\varepsilon[2\lambda_x\lambda_y-\sigma\mu]\ \text{для}\ \forall z\in\mathbb{R}\}.$$

Пусть

$$L(z)=\frac{\sigma}{\sigma^2+\omega^2\varepsilon^2},$$

$$M(z)=\frac{\sigma[\lambda_x^2-\lambda_y^2-\omega^2\varepsilon\mu]-\omega\varepsilon[2\lambda_x\lambda_y-\sigma\mu]}{\sigma^2+\omega^2\varepsilon^2}.$$

Теорема 1. *Если* $\alpha=\inf\limits_{z\in\mathbb{R}}\ L(z)>0$, $\beta_\lambda=\inf\limits_{z\in\mathbb{R}}\ M(z)>0$,

$$A_\lambda=\max\{\sup_{z\in\mathbb{R}}L(z),\sup_{z\in\mathbb{R}}M(z)\}<+\infty,$$

то билинейная форма (2.23) *задаёт в* $W_2^{(1)}(\mathbb{R},\mathbb{R}^2)$ *скалярное произведение, эквивалентное произведению* (2.22).

Доказательство. Достаточно показать, что имеют место следующие неравенства:

$$a(u,u)\leqslant const\cdot\parallel u\parallel^2_{W_2^{(1)}} \qquad (2.24)$$

$$\parallel u\parallel^2_{W_2^{(1)}}\leqslant const\cdot a(u,u) \qquad (2.25)$$

для любой $u\in W_2^{(1)}(\mathbb{R},\mathbb{R}^2)$.

Имеем

$$a(u,u)=\int\limits_{-\infty}^{+\infty}L(z)\sum_{i=1}^{2}\left(\frac{du_i}{dz}\right)^2dz+\int\limits_{-\infty}^{+\infty}M(z)\sum_{i=1}^{2}u_i^2dz.$$

Заметим, что $a(u,u)\geqslant 0$ при $\lambda=(\lambda_x,\lambda_y)\in D_\lambda$.

Имеем

$$\int\limits_{-\infty}^{+\infty} L(z) \sum_{i=1}^{2} \left(\frac{du_i}{dz} \right)^2 dz \leqslant A_\lambda \int\limits_{-\infty}^{+\infty} \sum_{i=1}^{2} \left(\frac{du_i}{dz} \right)^2 dz \leqslant A_\lambda \parallel u \parallel^2_{W_2^{(1)}} .$$

Далее

$$\int\limits_{-\infty}^{+\infty} M(z) \sum_{i=1}^{2} u_i^2 dz \leqslant A_\lambda \int\limits_{-\infty}^{+\infty} \sum_{i=1}^{2} u_i^2 dz \leqslant A_\lambda \parallel u \parallel^2_{W_2^{(1)}} .$$

Следовательно,

$$a(u,u) \leqslant 2A_\lambda \parallel u \parallel^2_{W_2^{(1)}},$$

и тем самым неравенство (2.24) установлено.

Докажем неравенство (2.25). Допустим, что оно не верно. Тогда для любого натурального $m \geqslant 1$ найдётся функция $u_m \in W_2^{(1)}(\mathbb{R}, \mathbb{R}^2)$, для которой

$$\parallel u_m \parallel^2_{W_2^{(1)}} > m \cdot a(u_m, u_m).$$

Положим

$$v_m(z) = \frac{u_m(z)}{\parallel u_m \parallel_{W_2^{(1)}}}.$$

Тогда

$$\parallel v_m \parallel_{W_2^{(1)}} = 1 \tag{2.26}$$

и

$$a(v_m, v_m) < \frac{1}{m}.$$

Откуда

$$\int\limits_{-\infty}^{+\infty} \sum_{i=1}^{2} \left(\frac{dv_{im}}{dz} \right)^2 dz < \frac{1}{m\alpha}, \tag{2.27}$$

$$\int\limits_{-\infty}^{+\infty} \sum_{i=1}^{2} v_{im}^2 dz < \frac{1}{m\beta_\lambda}. \tag{2.28}$$

Имеем, используя (2.27) и (2.28)

$$\parallel v_m - v_n \parallel^2_{W_2^{(1)}} = \parallel v_m - v_n \parallel^2_{L_2 \times L_2} + \parallel \frac{dv_m}{dz} - \frac{dv_n}{dz} \parallel^2_{L_2 \times L_2} \leqslant$$

$$\leqslant (\parallel v_m \parallel_{L_2 \times L_2} + \parallel v_n \parallel_{L_2 \times L_2})^2 + \left(\parallel \frac{dv_m}{dz} \parallel_{L_2 \times L_2} + \parallel \frac{dv_n}{dz} \parallel_{L_2 \times L_2} \right)^2$$

$$\leqslant 2 \parallel v_m \parallel^2_{L_2 \times L_2} + 2 \parallel v_n \parallel^2_{L_2 \times L_2} + 2 \parallel \frac{dv_m}{dz} \parallel^2_{L_2 \times L_2} + 2 \parallel \frac{dv_n}{dz} \parallel^2_{L_2 \times L_2} \leqslant$$

$$\leqslant \frac{2}{m\beta_\lambda} + \frac{2}{n\beta_\lambda} + \frac{2}{m\alpha} + \frac{2}{n\alpha}.$$

Тогда $\| v_m - v_n \|_{W_2^{(1)}} \to 0$ при $m, n \to \infty$, т. е. последовательность $\{v_m\}$ фундаментальна в $W_2^{(1)}(\mathbb{R}, \mathbb{R}^2)$. Поэтому она сходится в $W_2^{(1)}(\mathbb{R}, \mathbb{R}^2)$ к элементу $v \in W_2^{(1)}(\mathbb{R}, \mathbb{R}^2)$.

Переходя к пределу в (2.26), (2.28), получим

$$\| v \|_{W_2^{(1)}} = 1, \tag{2.29}$$

$$\int\limits_{-\infty}^{+\infty} \sum_{i=1}^{2} v_i^2 dz = 0. \tag{2.30}$$

Из (2.30) следует $v_i = 0$, т. е. $v \equiv 0$. Последнее противоречит (2.29).

Теорема 1 доказана.

■

Замечание 1. В практически важном случае обычно берут $\omega\varepsilon = 0$. Тогда

$$M(z) = \frac{\lambda_x^2 - \lambda_y^2}{\sigma(z)}.$$

Очевидно, условие $\beta_\lambda > 0$ означает ограниченность $\sigma(z)$.

2.3. Существование слабых решений в классе $W_2^{(1)}(\mathbb{R}, \mathbb{R}^2)$

Рассмотрим вопрос о существовании и единственности решения краевой задачи (2.17)–(2.21) в классе функций, допускающих первые обобщённые производные в смысла Соболева.

Домножим слева уравнение (2.17) на $\phi^* = (\phi_1, \phi_2)$, где ϕ_i – гладкие функции, и проинтегрируем по z от $z = 0$ до $z = z_i$ $(i = 0, 1)$:

$$\int\limits_0^{z_i} \phi^* \frac{d}{dz}\left(P\frac{du}{dz}\right) dz + \int\limits_0^{z_i} \phi^* Q u dz = 0$$

или

$$\phi^* P \frac{du}{dz}\Big|_0^{z_i} - \int\limits_0^{z_i} \frac{d\phi^*}{dz} P \frac{du}{dz} dz + \int\limits_0^{z_i} \phi^* Q u dz = 0.$$

Получаем два уравнения (при $i = 0, 1$):

$$\phi^*(z_1 - 0)P(z_0)\frac{du}{dz}(z_1 - 0) - \phi^*(0)P(0)\frac{du}{dz}(+0) - \int\limits_0^{z_1} \frac{d\phi^*}{dz} P \frac{du}{dz} dz + \int\limits_0^{z_1} \phi^* Q u dz = 0,$$

$$\phi^*(z_1 + 0)P(z_0)\frac{du}{dz}(z_1 + 0) - \phi^*(0)P(0)\frac{du}{dz}(-0) + \int\limits_{z_0}^{0} \frac{d\phi^*}{dz} P \frac{du}{dz} dz - \int\limits_{z_0}^{0} \phi^* Q u dz = 0.$$

Вычитая из первого уравнения второе, получаем

$$-\int\limits_{z_0}^{z_1}\frac{d\phi^*}{dz}P\frac{du}{dz}dz+\int\limits_{z_0}^{z_1}\phi^*Qudz+\phi^*(z_1-0)P(z_1)\frac{du}{dz}(z_1-0)-$$

$$-\phi^*(z_0+0)P(z_0)\frac{du}{dz}(z_0+0)-\phi^*(0)P(0)\left[\frac{du}{dz}\right]_{z=0}=0. \tag{2.31}$$

В силу непрерывности u и $\frac{du}{dz}$ на поверхностях раздела $z=z_i$ можно распространить интегрирование в (2.31) на всю вещественную ось z. В этом случае, учитывая условия излучения (2.19), мы вместо (2.31) будем иметь следующее тождество:

$$\int\limits_{-\infty}^{+\infty}\frac{d\phi^*}{dz}P\frac{du}{dz}dz+\int\limits_{-\infty}^{+\infty}\phi^*Qudz+$$

$$+\frac{\sigma(0)}{\sigma^2(0)+\omega^2\varepsilon^2(0)}\phi_1(0)+\frac{\omega\varepsilon(0)}{\sigma^2(0)+\omega^2\varepsilon^2(0)}\phi_2(0)=0. \tag{2.32}$$

Соотношение (2.32) распространяется на класс функций ϕ и u из $W_2^{(1)}(\mathbb{R},\mathbb{R}^2)$ с нормой

$$\parallel u \parallel_{W_2^{(1)}}=\sqrt{\parallel u_1 \parallel^2_{W_2^{(1)}(\mathbb{R})}+\parallel u_2 \parallel^2_{W_2^{(1)}(\mathbb{R})}}.$$

Введём следующую билинейную форму

$$a(u,\phi)=-\int\limits_{-\infty}^{+\infty}\frac{d\phi^*}{dz}P\frac{du}{dz}dz+\int\limits_{-\infty}^{+\infty}\phi^*Qudz \tag{2.33}$$

и линейную форму

$$l(\phi)=-\frac{\sigma(0)}{\sigma^2(0)+\omega^2\varepsilon^2(0)}\phi_1(0)-\frac{\omega\varepsilon(0)}{\sigma^2(0)+\omega^2\varepsilon^2(0)}\phi_2(0)=0. \tag{2.34}$$

Тогда (2.32) принимает вид

$$a(u,\phi)=l(\phi). \tag{2.35}$$

Имеем

$$a(u,u)=\int\limits_{-\infty}^{+\infty}L(z)\sum_{i=1}^{2}\left(\frac{du_i}{dz}\right)^2dz+\int\limits_{-\infty}^{+\infty}M(z)\sum_{i=1}^{2}{du_i}^2dz.$$

Допустим, что

$$\alpha=\inf_{z\in\mathbb{R}}L(z)>0 \tag{2.36}$$

и пара (λ_x,λ_y) принадлежит области $D_\lambda\subset\mathbb{R}^2$, определённой как

$$D_\lambda=\{(\lambda_x,\lambda_y)\in\mathbb{R}^2:\sigma[\lambda_x^2-\lambda_y^2-\omega^2\varepsilon\mu]>\omega\varepsilon[2\lambda_x\lambda_y-\sigma\mu]\text{ для }\forall z\in\mathbb{R}\}.$$

Зафиксируем $\lambda \in D_\lambda$. По теореме 1 билинейная форма $a(u,\phi)$ задаёт в $W_2^{(1)}(\mathbb{R},\mathbb{R}^2)$ скалярное произведение, эквивалентное стандартному скалярному произведению

$$(u,\phi)_{W_2^{(1)}} = (u_1,\phi_1)_{W_2^{(1)}} + (u_1,\phi_1)_{W_2^{(1)}},$$

если только

$$\beta_\lambda = \inf_{z\in\mathbb{R}} \frac{\sigma[\lambda_x^2 - \lambda_y^2 - \omega^2\varepsilon\mu] - \omega\varepsilon[2\lambda_x\lambda_y - \sigma\mu]}{\sigma^2 + \omega^2\varepsilon^2} > 0$$

и $A_\lambda < +\infty$ (см. 2.2).

Следовательно,

$$a(u,u) \geqslant const\cdot \parallel u \parallel^2_{W_2^{(1)}} . \tag{2.37}$$

Далее в области D_λ имеем

$$|a(u,\phi)| \leqslant A_\lambda \left[\sum_{i=1}^{2}\int_{-\infty}^{+\infty}\left|\frac{du_i}{dz}\frac{d\phi_i}{dz}\right|dz + \sum_{i=1}^{2}\int_{-\infty}^{+\infty}|u_i\phi_i|dz\right] \leqslant$$

$$\leqslant A_\lambda \left[\sum_{i=1}^{2}\left(\int_{-\infty}^{+\infty}\left|\frac{du_i}{dz}\right|^2 dz\right)^{1/2}\left(\int_{-\infty}^{+\infty}\left|\frac{d\phi_i}{dz}\right|^2 dz\right)^{1/2} + \sum_{i=1}^{2}\left(\int_{-\infty}^{+\infty}u_i^2 dz\right)^{1/2}\left(\int_{-\infty}^{+\infty}\phi_i^2 dz\right)^{1/2}\right] \leqslant$$

$$\leqslant A_\lambda \left[\sum_{i=1}^{2}\left(\parallel \frac{du_i}{dz} \parallel^2_{L_2(\mathbb{R})} + \parallel u_i \parallel^2_{L_2(\mathbb{R})}\right)^{1/2}\left(\parallel \frac{d\phi_i}{dz} \parallel^2_{L_2(\mathbb{R})} + \parallel \phi_i \parallel^2_{L_2(\mathbb{R})}\right)^{1/2} +\right.$$

$$\left. + \sum_{i=1}^{2}\left(\parallel u_i \parallel^2_{L_2(\mathbb{R})} + \parallel \frac{du_i}{dz} \parallel^2_{L_2(\mathbb{R})}\right)^{1/2}\left(\parallel \phi_i \parallel^2_{L_2(\mathbb{R})} + \parallel \frac{d\phi_i}{dz} \parallel^2_{L_2(\mathbb{R})}\right)^{1/2}\right] =$$

$$= 2A_\lambda \sum_{i=1}^{2} \parallel u_i \parallel_{W_2^{(1)}} \cdot \parallel \phi_i \parallel_{W_2^{(1)}} \leqslant$$

$$\leqslant A_\lambda \sqrt{\sum_{i=1}^{2} \parallel u_i \parallel_{W_2^{(1)}}} \cdot \sqrt{\sum_{i=1}^{2} \parallel \phi_i \parallel_{W_2^{(1)}}} = A_\lambda \parallel u \parallel_{W_2^{(1)}} \cdot \parallel \phi \parallel_{W_2^{(1)}},$$

т. е.

$$|a(u,\phi) \leqslant const\cdot \parallel u \parallel_{W_2^{(1)}} \cdot \parallel \phi \parallel_{W_2^{(1)}} . \tag{2.38}$$

Из (2.37), (2.38), согласно теореме Лакса–Мильграма [3, с. 180], следует однозначная разрешимость вариационной задачи (2.35) в классе $W_2^{(1)}(\mathbb{R},\mathbb{R}^2)$.

Итак, краевая задача (2.17)–(2.21) имеет единственное слабое решение. При этом достаточно требовать лишь интегрируемости функций $L(z)$ и $M(z)$, а не $\sigma,\mu,\varepsilon \in C^1(\mathbb{R})$.

2.4. Определение области D_λ

Пусть $u(z,\lambda)$ — классическое решение задачи (2.17)–(2.21). Покажем, что оно не имеет особенностей по λ при $\lambda \in D_\lambda$. Из (2.32) при $\phi = u$ имеем

$$\int\limits_{-\infty}^{+\infty} \frac{du^*}{dz} P \frac{du}{dz} dz + \int\limits_{-\infty}^{+\infty} u^* Q u dz +$$

$$+ \frac{\sigma(0)}{\sigma^2(0)+\omega^2\varepsilon^2(0)} u_1(0) + \frac{\omega\varepsilon(0)}{\sigma^2(0)+\omega^2\varepsilon^2(0)} u_2(0) = 0. \tag{2.39}$$

При $\lambda \in D_\lambda$ имеем

$$q(\lambda) \equiv -\frac{\sigma(0)}{\sigma^2(0)+\omega^2\varepsilon^2(0)} u_1(0,\lambda) - \frac{\omega\varepsilon(0)}{\sigma^2(0)+\omega^2\varepsilon^2(0)} u_2(0,\lambda) \geqslant 0. \tag{2.40}$$

Из (2.39) и (2.40) получаем неравенство

$$0 \leqslant \int\limits_{-\infty}^{+\infty} \left(\frac{du_2}{dz}\right)^2 dz \leqslant \alpha^{-1} q(\lambda). \tag{2.41}$$

Теорема 2. *При условии* (2.36) *классическое решение* $u(z,\lambda)$ *не имеет по переменной* $\lambda \in D_\lambda$ *особенностей.*

Доказательство. Пусть $u(z,\lambda)$ имеет полюс порядка m в точке $\lambda_0(z)$, т. е. предполагаем, что $u(z,\lambda)$ аналитична по λ, кроме точки $\lambda_0(z)$.

Тогда в некоторой окрестности точки $\lambda_0(z)$ имеем

$$u(z,\lambda) = \frac{a(z)}{[\lambda - \lambda_0(z)]^m}, \quad a(z) \neq 0, \ m \geqslant 1. \tag{2.42}$$

Откуда

$$u_2(z,\lambda) = [(\lambda_x - \lambda_{x0}(z))^2 + (\lambda_y - \lambda_{y0}(z))^2]^{-m} \times$$

$$\times Im\ a(z) \sum_{k=0}^{m} C_m^k \cdot (\lambda_x - \lambda_{x0}(z))^k (-i)^{m-k} (\lambda_y - \lambda_{y0}(z))^{m-k}$$

или

$$u_2(z,\lambda) = \frac{\sum\limits_{k=0}^{m} b_k(z)(\lambda_x - \lambda_{x0}(z))^k (\lambda_y - \lambda_{y0}(z))^{m-k}}{[(\lambda_x - \lambda_{x0}(z))^2 + (\lambda_y - \lambda_{y0}(z))^2]^m}. \tag{2.43}$$

Покажем, что

$$\left.\begin{array}{l} \lambda_0(z) \in C(\mathbb{R}) \cap C^1(\mathbb{R} \setminus \{0\}), \\ \text{производные } \lambda_0(z) \text{ справа и слева от } 0 \text{ конечны.} \end{array}\right\} \tag{2.44}$$

Действительно, из (2.42) видно, что если $\lambda \neq \lambda_0(z)$, то $u(z,\lambda)$ можно дважды дифференцировать по λ.

Тогда, дифференцируя по λ равенство

$$[\lambda - \lambda_0(z)]^m u(z,\lambda) = a(z),$$

получим

$$[\lambda - \lambda_0(z)]^m \frac{\partial u}{\partial \lambda} + m[\lambda - \lambda_0(z)]^{m-1} u = 0,$$

$$[\lambda - \lambda_0(z)] \frac{\partial u}{\partial \lambda} + mu = 0,$$

$$\lambda_0(z) = \lambda + \frac{mu}{\left(\dfrac{\partial u}{\partial \lambda}\right)}. \tag{2.45}$$

Так как $u \in C^1(\mathbb{R} \setminus \{0\})$, $u \in C(\mathbb{R})$ и существуют конечные производные у u по z в 0 справа и слева, то из (2.45) следует (2.44)[1].

Предположим, что $\lambda_0(z) \not\equiv \lambda_0(0)$. Возьмём z_0 так, что $\lambda_0(z_0) \neq \lambda_0(0)$. Из непрерывности $\lambda_0(z)$ заключаем, что существует сегмент $[\gamma, \delta]$, $z_0 \in (\gamma, \delta)$ такой, что разложение (2.42) справедливо для всех $z \in [\gamma, \delta]$ при любом λ, принадлежащем некоторому кругу θ с центром $\lambda_0(z_0)$. Тогда из (2.41) следует

$$0 \leqslant \int_\gamma^\delta \left(\frac{du_2}{dz}\right)^2 dz \leqslant \alpha^{-1} q(\lambda), \quad \lambda \in \theta,$$

куда можно подставить (2.43).

Получаем

$$0 \leqslant \int_\gamma^\delta \frac{[p(\lambda_x - \lambda_{x0}(z), \lambda_y - \lambda_{y0}(z))]^2}{[(\lambda_x - \lambda_{x0}(z))^2 + (\lambda_y - \lambda_{y0}(z))^2]^{2m+2}} dz \leqslant \alpha^{-1} q(\lambda), \tag{2.46}$$

где $p(u,v)$ — полином степени $\leqslant m+1$ с коэффициентами — полиномами от $a_x(z), a_y(z)$, $\frac{d\lambda_{x0}(z)}{dz}$, $\frac{d\lambda_{y0}(z)}{dz}$.

Возьмём λ так, что оно не есть особенность для $u(0,\lambda)$. Тогда $q(\lambda)$ — конечно, то есть (2.46) означает сходимость выписанного интеграла. Но он как раз расходящийся, если λ равно, например, $\lambda_0(z_1)$, где $z_1 \in [\gamma, \delta]$.

Получили противоречие.

Но мы предполагали, что $\lambda_0(z) \not\equiv \lambda_0(0)$. Пусть теперь $\lambda_0(z) \equiv \lambda_0(0)$.

При $\lambda_y = \lambda_{y0}(0)$ имеем

$$u_2(z,\lambda) = \frac{a_y(z)}{[\lambda_x - \lambda_{x0}(0)]^m}, \tag{2.47}$$

$$u_1(z,\lambda) = \frac{a_x(z)}{[\lambda_x - \lambda_{x0}(0)]^m}, \tag{2.48}$$

[1] $\partial u/\partial\lambda \in C(\mathbb{R} \setminus \{0\})$, поскольку дифференцируя интегральное тождество (2.32) по λ, получаем интегральное тождество для $\partial u/\partial\lambda$. Для него вариационная задача подобна задаче (2.32). Значит $\partial u/\partial\lambda \in W_2^{(1)}$. Откуда $\partial u/\partial\lambda \in C(\mathbb{R})$, благодаря вложению $W_2^{(1)} \subset C$.

Из (2.39), (2.40) имеем

$$0 \leqslant \int\limits_{-\infty}^{+\infty} \left[\left(\frac{du_1}{dz} \right)^2 + \left(\frac{du_2}{dz} \right)^2 \right] dz \leqslant \alpha^{-1} q(\lambda). \tag{2.49}$$

Подставляя (2.47), (2.48) в (2.49), получаем

$$0 \leqslant \int\limits_{-\infty}^{+\infty} \left[\left(\frac{du_1}{dz} \right)^2 + \left(\frac{du_2}{dz} \right)^2 \right] dz \leqslant \alpha^{-1} q(\lambda) [\lambda_x - \lambda_{x0}(0)]^{2m} =$$

$$= -\frac{\alpha^{-1}}{\sigma^2(0) + \omega^2 \varepsilon^2(0)} [a_x(0)\sigma(0) + \omega\varepsilon(0) a_y(0)] [\lambda_x - \lambda_{x0}(0)]^{2m}.$$

Устремляя λ_x к $\lambda_{x0}(0)$, получаем

$$\int\limits_{-\infty}^{+\infty} \left[\left(\frac{da_x}{dz} \right)^2 + \left(\frac{da_y}{dz} \right)^2 \right] dz = 0,$$

то есть

$$a_x(z) = const, \quad a_y(z) = const.$$

Учитывая условие на бесконечности (2.19), получаем

$$a(z) \equiv 0,$$

то есть

$$u(z, \lambda) \equiv 0.$$

Это противоречит граничному условию $[\frac{du}{dz}]_{z=0} = 1$. Следовательно, функция $u(z, \lambda)$ не имеет полюсов по λ в области D_λ.

Но $u(z, \lambda)$ не имеет и существенных особенностей. Действительно, по теореме Пикара [4, с. 141] $u(0, \lambda)$ будет принимать в окрестности существенно особых точек $\lambda_0(0)$ любые конечные значения, за исключением, быть может, одного. Тогда найдётся точка λ_1 такая, что $u(0, \lambda_1) = u_1(0, \lambda_1) > 0$. Но это противоречит неравенству (2.40).

Итак, особых точек по λ у решения $u(z, \lambda)$ задачи (2.17)–(2.21) быть не может.

Теорема 2 доказана.

■

2.5. Вычисление электромагнитного поля на ЭВМ

Имеем

$$\mathbf{H} = rot\ \mathbf{A},$$

$$\mathbf{E} = i\omega\mu\mathbf{A} + \nabla \left\{ \frac{1}{\sigma - i\omega\varepsilon} div\mathbf{A} \right\}, \quad \mathbf{A} = (0, 0, A),$$

где

$$A(x,y,z)=\frac{1}{2\pi}\int\limits_0^{+\infty}\lambda I_0(\lambda r)u(z,\lambda)d\lambda.$$

Следовательно, поля $\mathbf{H}$ и $\mathbf{E}$ выражаются в виде суммы интегралов вида

$$\int\limits_0^{+\infty}\mathcal{D}^\alpha I_0(\lambda r)\frac{d^k u(z,\lambda)}{dz^k}\lambda d\lambda,\quad |\alpha|,k\leqslant 2,$$

где $\mathcal{D}^\alpha$ — производная по x и y.

В силу теоремы 2 при $\lambda\in D_\lambda$ функция

$$\frac{d^k u(z,\lambda)}{dz^k}$$

не имеет особенностей по λ. Значит, можно деформировать контур интегрирования в комплексную область изменения переменной λ, не выходя за пределы области D_λ. Это позволяет избежать быстроосцилирующих функций в качестве

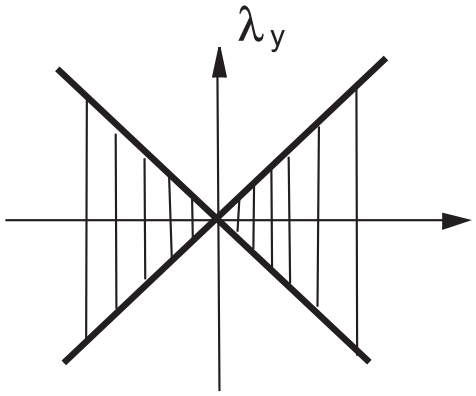


Рис. 2. Область D_λ

подынтегральных выражений и, следовательно, эффективно провести вычисление полей $\mathbf{H}$, $\mathbf{E}$ на ЭВМ.

В практически важном случае $\omega\varepsilon\approx 0$. Тогда

$$D_\lambda=\{(\lambda_x,\lambda_y)\in\mathbb{R}^2:|\lambda_x|>|\lambda_y|\},$$

т. е. область допустимой деформации контура интегрирования достаточно простая (рис. 2), в то же время весьма приемлемая с точки зрения процедуры вычисления интересующих нас интегралов на ЭВМ [2].

Заключение

Результат, представленный в данной статье, был анонсирован в [5], а его полное изложение было дано в научном отчёте [6], который находится в труднодоступном архиве. Там можно найти и решение аналогичной задачи для горизонтально-слоистой среды с горизонтальным (магнитным и электрическим) диполем в качестве источника.

Следует сказать, что случай вертикального электрического диполя является наиболее сложным. Сложнее только горизонтальные диполи. Эти случаи дают не одно, а два уравнения, аналогичные вертикальным электрическому и магнитному диполям (второе проще анализировать). Кроме того, в одном из них на горизонтали источника терпит разрыв не производная, а само решение. Это вынуждает изменить класс пространств, в котором ищется решение. Но ничего принципиально нового нет.

В одном из отчётов [7–11] приведена иная постановка задачи для случая всех диполей. В качестве неизвестных выбраны вертикальные компоненты полного (включая сторонний источник) электрического и магнитного «тока». Коэффициенты принадлежат пространству L_∞, а решение ищется в классе $W_2^{(1)}$. Более того, удалось дать определения вертикального и горизонтального диполей (и электрического, и магнтного), находящихся на границе разрыва параметров среды (но не наклонного). При этом содержание теорем не меняется.

Наконец, заметим, что в настоящее время модель вертикально неоднородной среды практически не актуальна. Тем не менее она используется:

а) как фоновая в методе объёмных интегральных уравнений. Именно для неё вычисляется функция (тензор) Грина;

б) в случае недостатка числа измеряемых сигналов, например, при инверсии для навигации во время бурения;

в) реже в других случаях электромагнитного каротажа или наземной электроразведки.

Литература

1. Смагин С.И. Расчёт функции Грина уравнения Гельмгольца с одномерным кусочно-постоянным волновым числом // Сб.: Условно-корректные задачи математической фишки в интерпретации геофизических наблюдений. Новосибирск, 1973.
2. Табаровский Л.А. Применение метода интегральных уравнений в задачах геоэлектрики. Новосибирск : Изд-во «Наука», Сибирское отделение, 1975.
3. Partial Differential Equations // Lectures in applied mathematics. 1957. V. 31.
4. Бицадзе А.В. Основы теории аналитических функций комплексного переменного. М. : Наука, 1969.
5. Гуц А.К., Терентьев С.А. Исследования особенностей спектральной плотности для электромагнитного поля в вертикально неоднородной проводящей среде // Сб.: Автоматизация анализа и синтеза структур ЭВМ и вычислительных алгоритмов. Омск : ОмПИ, 1982. С. 78–80.

6. Терентьев С.А., Гуц А.К. Теоретическое исследование электромагнитного поля в проводящих неоднородных средах // Отчёт по НИР. Омск : ОмГУ, 1980. Деп.во ВНТИЦ 25.02.81, № Б 919817. 48 с.
7. Терентьев С.А., Гуц А.К., Кайзер В.В. Теоретическое исследование электромагнитного поля в проводящих неоднородных средах // Депонированный отчёт по НИР. Инв. № 0283.0006913. Омск : ОмГУ, 1982. 58 с.
8. Терентьев С.А., Бронников И.Н. Разработка алгоритмов расчёта на ЭВМ электромагнитных полей источников различной конфигурации в горизонтально-слоистой среде // Депонированный отчёт по НИР. Инв. № 0285.0011141, № гос. рег. 0184.0015161. Омск : ОмГУ, 1984. 31 с.
9. Терентьев С.А. Алгоритм расчёта электромагнитного поля в вертикально-неоднородной проводящей среде // Сб.: Электрофизические проблемы защиты устройств связи от влияний на железнодорожном транспорте. Омск : ОмИИТ, 1985. С. 32–33.
10. Терентьев С.А., Балыкина О.Н, Романовская А.М., Ультан А.Е. Математические методы в прикладных исследованиях // Депонированный отчет по НИР. Инв. № 0986.0039352, № гос. рег. 0185.0051835. Омск : ОмГУ, 1985, 90 с.
11. Терентьев С.А., Балыкина О.Н, Романовская А.М., Ультан А.Е. Математические методы в прикладных исследованиях. // Депонированный отчёт по НИР. Инв. № 02.88.0022619, № гос. рег. 0185.0051835. Омск : ОмГУ, 1987. 54 с.

INVESTIGATIONS OF THE SPECTRAL DENSITY OF THE ELECTROMAGNETIC FIELD IN A VERTICALLY INHOMOGENEOUS CONDUCTIVE MEDIUM

S.A. Terentyev
PhD. (Phys.-Math.), Associate Professor, e-mail: sa.terentyev@gmail.com
A.K. Guts
Dr.Sc. (Phys.-Math.), Professor, e-mail: guts@omsu.ru

Dostoevsky Omsk State University, Omsk, Russia

Abstract. The electromagnetic field in electrical exploration problems is often represented as integrals with a fast-oscillating nucleus. When calculating these integrals on a computer, it is necessary to deform the contour of integration into the plane of the complex variable. The article studies the allowable deformation region of the integration contour in the case of a non-uniform medium. The source of the field is a vertical dipole. A similar problem was solved for a horizontally layered medium with a horizontal harmonious dipole as a source.

Keywords: Electrical exploration, electromagnetic field of vertical electric dipole, fast-oscillating integrals, deformation contour, complex plane, absence of singular points, deformation domain.

Дата поступления в редакцию: 17.10.2018

Математические структуры и моделирование 2018. № 4(48). С. 78–84

УДК 519.237.7
DOI: 10.25513/2222-8772.2018.4.78-84

ФАКТОРНЫЙ АНАЛИЗ НА БАЗЕ МЕТОДА K-СРЕДНИХ

В.А. Шовин
научный сотрудник, e-mail: v.shovin@mail.ru

Институт математики им. С.Л. Соболева Сибирского отделения РАН, Омск, Россия

Аннотация. Актуальной проблемой медицинских и математических исследований является анализ и поиск скрытых зависимостей в экспериментальных данных. Определение таких зависимостей позволяет построить модель явления или объекта, которая бы наиболее соответствовала экспериментальным данным и при этом обладала минимальной сложной структурой. Известным математическим и программным инструментом для автоматического построения таких моделей является факторный анализ. Представляются востребованными различные обобщения и модернизации методов факторного анализа. В статье предлагается новый подход к проведению факторного анализа на базе метода кластеризации данных k-средних и последующего факторного вращения. Факторный анализ выделяет из множества исходных показателей — k главных компонент или факторов — с наибольшей точностью аппроксимирующих разброс и распределение исходных данных. Такие главные компоненты формируют факторную структуру исходных данных. В качестве направлений и положений главных компонент могут быть использованы различные характеристические точки исходной структуры данных. В данной работе предлагается использовать центры кластеров исходных данных. Для разделения точек данных на классы существует большое число методов кластеризации. Наиболее популярным является метод k-средних. В результате метод k-средних позволяет найти факторную структуру в исходном многомерном пространстве данных из положений k центров выделенных кластеров. Последующее факторное вращение по оригинальному критерию интерпретируемости позволяет найти простую факторную структуру. Проведение численных экспериментов показало хорошее соответствие результатов данного метода факторного анализа с ранее известными методами. Предлагаемый метод факторного анализа обладает хорошей и превосходящей эффективностью по сравнению с другими методами факторного анализа. Он обладает меньшей сложностью и количеством необходимых действий для определения факторной структуры.

Ключевые слова: метод k-средних, факторный анализ, факторное вращение.

Введение

Факторное моделирование является одним из наиболее востребованных инструментов для изучения скрытых закономерностей в экспериментальных данных. Методы факторного анализа позволяют автоматически построить простую факторную модель данных и произвести диагностику новых объектов. Факторная структура данных отражает основные направления разброса данных в многомерном пространстве исходных показателей объектов. Суть факторной структуры данных — это каркас данных, с нужной точностью покрывающий исходные данные. Математическая постановка задач в области факторного анализа — это разработка подходов к вычислению и построению минимальных факторных структур в различных случаях распределений данных.

Факторный и кластерный анализ являются одними из самых популярных методов анализа данных и математической статистики. Кластерный анализ позволяет автоматически найти классы объектов, используя только информацию о количественных показателях объектов (обучение без учителя). Каждый такой класс может задаваться одним самым характерным для него объектом, например, средним по показателям. Существует большое число методов и подходов для классификации данных.

Факторный анализ позволяет найти факторную структуру показателей объектов такую, которая гипотетически объясняла бы значение экспериментальных данных и находилось в тесной математической связи с ними. Для получения первичного факторного решения на данный момент существует большое число методов. Например, итеративный метод и метод Якоби расчёта собственных векторов и собственных значений, центройдный метод, метод минимальных остатков, метод максимального правдоподобия [1]. Такие алгоритмы обладают большой вычислительной сложностью. Поэтому в работе предлагается подход, уменьшающий сложность алгоритма вычисления факторной структуры.

В данной статье предлагается использовать метод k-средних для определения центров кластеров и определения с помощью них факторной структуры, определяющей каркас данных.

Такая факторная структура может быть подвергнута факторному вращению для получения её простоты с точки зрения интерпретации. Существует большое количество методов факторного вращения. В работе предлагается использовать авторский критерий интерпретируемости, позволяющий упростить интерпретацию факторной структуры.

Современные исследования в области факторного анализа проводятся с целью нелинейных обобщений факторных структур [2,3], а также совершенствования эффективности методов в случае больших данных. Нелинейные факторные структуры позволяют с большей точностью аппроксимировать разброс данных, а также учитывать сложную топологию моделей данных. В таких методах часто задействуется инструмент нейронных сетей и их обучение. Огромным коммерческим спросом пользуется изучение больших данных, хранящихся в объёмных базах данных. Поэтому актуальными являются разработка и совершенствование методов, позволяющих быстро обрабатывать такие массивы

данных. Целью подобных исследований является снижение порядка сложности и количества действий алгоритмов.

В предлагаемом в работе подходе для расчёта факторной структуры замена метода классификации на другой метод позволит дополнительно улучшить скорость работы алгоритма и снизить его сложность.

1. Метод k-средних

Метод k-средних является одним из наиболее популярных методов кластеризации. Его целью является получение таких центров данных, которые соответствовали бы гипотезе компактности классов данных при их симметричном радиальном распределении. Одним из способов определить положение таких центров, при заданном их количестве k, является EM–подход.

В данном методе выполняются последовательно две процедуры.

1. Определение для каждого объекта данных X_i ближайшего центра C_j, и назначение данному объекту метки класса X_i^j. Далее для всех объектов определяется их принадлежность различным классам.

2. Вычисление нового положения центров всех классов.

Итеративно повторяя эти две процедуры из начального случайного положения центров k классов, можно добиться разделения объектов на классы, которые максимально соответствовали бы гипотезе радиальной компактности классов.

2. Факторный анализ

В линейной модели факторного анализа становится понятна матрица факторной структуры, определяемой из следующей формулы линейной связи значений факторов и исходных переменных [1,4]:

$$\begin{cases} a_{11}p_{1t} + a_{12}p_{2t} + \ldots + a_{1g}p_{gt} \to z_{1t}, \\ a_{21}p_{1t} + a_{22}p_{2t} + \ldots + a_{2g}p_{gt} \to z_{2t}, \\ \qquad\qquad\qquad \vdots \\ a_{m1}p_{1t} + a_{m2}p_{2t} + \ldots + a_{mg}p_{gt} \to z_{mt}, \end{cases} \tag{1}$$

где матрица $A \underset{m\times g}{\leftrightarrow} a_{ij}$ называется матрицей факторной структуры размерности $m \times g$ весовых коэффициентов и где m — число изучаемых параметров, g — число общих факторов.

Матрица $Z \underset{m\times n}{\leftrightarrow} z_{ij}$ — матрица значений измеряемых переменных у исследуемых объектов или состояний объекта размерности $m \times n$, где m — число измеряемых параметров, n — число объектов или состояний объекта (объём выборки).

Матрица $P \underset{g\times n}{\leftrightarrow} p_{ij}$ — матрица значений латентных переменных (факторов) объектов размерности $g \times n$, где g — число латентных параметров.

На вид факторной структуры A налагаются дополнительные ограничения: общности переменных факторной структуры должны быть не больше 1, а также не меньше определённого порога значимости:

$$h_i = \sqrt{\sum_{k=1}^{g} a_{ik}^2} \leqslant 1.$$

Результатом факторного анализа является определение двух матриц *A* и *P*.

3. Гипотеза соответствия

Можно предположить, что матрица координат центров g классов данных может быть использована как матрица факторной структуры A. В то время как матрица вероятностного отношения объектов к различным классам (матрица дистанций объектов до центров классов) — как матрица значений факторов *P*.

Для полного соответствия координаты центров g классов приводятся в диапазон [-1, 1] с приведением общностей $h_i = 1$ по формулам:

$$\min_i = \min_{j=1\ldots g} a_{ij};$$

$$\max_i = \max_{j=1\ldots g} a_{ij};$$

$$a_{ij} := \alpha a_{ij} + \beta.$$

$$\alpha = \frac{2}{\max - \min}, \beta = -\frac{\max + \min}{\max - \min}.$$

$$a_{ij} := \frac{2a_{ij} - \max_i - \min_i}{\max_i - \min_i};$$

$$a_{ij} := \frac{a_{ij}}{\sqrt{\sum_{k=1}^{g} a_{ik}^2}}.$$

Численный эксперимент

В качестве исходных данных были взяты 15 биофизических показателей для 131 лица с артериальной гипертензией начальной стадии [5]:

1) *вес,*
2) *индекс массы тела (ИМТ),*
3) *частота дыхания (ЧД),*
4) *сегментоядерные нейтрофилы (С),*
5) *лимфоциты (Л),*
6) *конечно-систолический размер левого желудочка (КСР),*

7) *конечно-систолический объём левого желудочка (КСО),*
8) *конечно-диастолический размер левого желудочка (КДР),*
9) *конечно-диастолический объём левого желудочка (КДО),*
10) *ударный объём (УО),*
11) *минутный объём сердца (МОС),*
12) *общее периферическое сосудистое сопротивление (ОПСС),*
13) *индекс Хильдебрандта (ИХ),*
14) *фракция выброса левого желудочка (ФВ),*
15) *фракция укорочения левого желудочка (ФУ).*

В таблице 1 приведена матрица факторной структуры, выделенная по методу главных компонент и подвергнутая факторному вращению по критерию интерпретируемости [6].

В таблице 2 приведена матрица факторной структуры, выделенная по гипотезе соответствия кластерного анализа и факторного анализа и подвергнутая факторному вращению по критерию интерпретируемости.

Незначительное несоответствие факторных структур двух методов могло быть результатом применения к матрице данных другого метода извлечения грубых ошибок. В целом по результату сравнения значимых факторных нагрузок можно утверждать, что гипотеза соответствия факторного и кластерного анализа оказалась правдивой на данных артериальной гипертензии.

Сложность алгоритма факторного анализа на базе метода k-средних $\sim m^2$ и $\sim n$. В то время как, например, подход на базе метода Якоби для вычисления собственных векторов и собственных значений имеет сложность $\sim m^4$ и $\sim n$.

4. Программная реализация

Метод кластеризации k-средних был реализован программно как web–приложение. Вычислительная часть приложения вынесена на сервер, написанный на языке PHP с использованием фреймворка Zend. Интерфейс приложения написан с использованием HTML, CSS, JavaScript, JQuery. Приложение доступно по адресу: http://svlaboratory.org/application/klaster — после регистрации нового пользователя. Приложение позволяет визуализировать принадлежность объектов различным кластерам в заданной плоскости координат.

Заключение

Предложен метод получения факторной структуры данных из матрицы координат центров k классов, выделяемых методом k-средних. Такой подход к извлечению факторной структуры является более эффективным в случае больших данных, т. к. требует меньшего количества действий, чем, например, метод Якоби для вычисления главных компонент.

Таблица 1. Факторная структура по критерию интерпретируемости, полученная из матрицы главных факторов

	*F*1	*F*2	*F*3	*F*4	*F*5
Вес	0,1949	0,0000	-0,021	-0,0062	**0,7783**
ИМТ	0,16	-0,0815	0,0000	0,0000	**0,7679**
ЧД	0,2165	-0,0211	0,01	**-0,8305**	0,0261
С	-0,0598	-0,0382	**-0,888**	0,0225	0,1504
Л	0,0000	0,0000	**0,9059**	0,0205	-0,19
КСР	**0,8631**	-0,4849	0,008	-0,0288	0,0149
КСО	**0,8502**	-0,4783	-0,0095	-0,0214	0,0097
КДР	**0,9721**	-0,0817	0,0251	0,0187	-0,0001
КДО	**0,9734**	-0,1221	-0,0066	0,0000	0,0178
УО	**0,9085**	0,2377	0,0152	0,0023	0,0202
МОС	**0,8934**	0,2084	-0,0105	0,0107	-0,0357
ОПСС	**-0,7374**	-0,2506	0,0000	0,0829	0,0839
ИХ	-0,1191	0,0000	0,0003	**0,8462**	0,0000
ФВ	-0,2005	**0,8173**	-0,0213	-0,0535	-0,0217
ФУ	-0,1863	**0,7167**	0,0441	0,0000	0,0094

Таблица 2. Факторная структура по критерию интерпретируемости, полученная из кластерного анализа

	*F*1	*F*2	*F*3	*F*4	*F*5
Вес	-0,3482	0,0147	-0,3964	0,1242	**0,774**
ИМТ	-0,2878	-0,0165	-0,3872	0,3025	**0,8224**
ЧД	-0,3177	-0,0757	-0,4894	**0,7997**	-0,0493
С	0,3827	-0,0557	**-0,8118**	-0,1787	0,2269
Л	-0,3032	-0,0097	**0,797**	0,0141	-0,4382
КСР	**-0,9656**	-0,046	-0,103	0,0591	0,0971
КСО	**-0,9596**	-0,0315	-0,152	0	-0,0075
КДР	**-0,9651**	-0,0355	0,0096	0,0537	0,1883
КДО	**-0,978**	-0,02	-0,065	-0,0082	0,0537
УО	**-0,983**	-0,0036	0,032	-0,0195	0,1101
МОС	**-0,9776**	-0,0148	0,0174	-0,0557	0,1226
ОПСС	**0,8594**	0,0181	-0,2225	-0,1846	-0,3785
ИХ	0,1003	-0,0087	0,6124	**-0,7831**	0
ФВ	**0,9086**	-0,0241	0,1947	0,1185	0,2376
ФУ	**0,9093**	0,3854	-0,1322	0,0765	0,0402

Литература

1. Харман Г. Современный факторный анализ / пер. с англ. В.Я. Лумельского. М. : Статистика, 1972.
2. Gorban A., Kegl B., Wunsch D., Zinovyev A. Principal Manifolds for Data Visualization and Dimension Reduction / Springer-Verlag Berlin Heidelberg. 2008. V. 58. P. 340.
3. Шовин В.А. Факторное моделирование артериальной гипертензии на базе метода Верле / Междисциплинарные исследования в области математического моделирования и информатики // Материалы 7-й научно-практической internet-конференции. 2016. С. 190–198.
4. Иберла К. Факторный анализ / пер. с нем. В.М. Ивановой. М. : Статистика, 1980.
5. Гольтяпин В.В., Шовин В.А. Косоугольная факторная модель артериальной гипертензии первой стадии // Вестник Омского университета. 2010. № 4. С. 120–128.
6. Шовин В.А., Гольтяпин В.В. Методы вращения факторных структур // Математические структуры и моделирование. 2015. № 2. С. 75–84.

FACTOR ANALYSIS BASED ON THE K-MEANS METHOD

V.A. Shovin
Scientist Researcher, e-mail: v.shovin@mail.ru

The Federal State Budgetary Institution of Science Sobolev Institute of Mathematics of the Siberian Branch of RAS (Omsk Branch)

Abstract. Actual problems and research methods are analysis and search for hidden dependencies in experimental data. The definition of such dependencies allows them to construct a model of phenomenon or object that would best fit the experimental data and at the same time have a minimally complex version. Known mathematical and software tools for the automatic construction of such models is a factor analysis. Various generalizations and modernizations of the methods of factor analysis are in demand. The article proposes a new approach to factor analysis based on the k-means data clustering method and subsequent factor rotation. Factor analysis identifies from a set of initial indicators k main components or factors — with the greatest accuracy approximating the scatter and distribution of the initial data. These main components form the factorial structure of the source data. Various characteristic points of the original data structure can be used as the directions and aspects of the main components. In this paper, we propose to use centers of raw data clusters. There is a large number of clustering methods for dividing data points into classes. The most popular is the k-means method. As a result, the k-means method allows finding the factor structure in the original multidimensional data space from the positions of the k centers of the selected clusters. The subsequent factor rotation according to the original criterion of interpretability allows us to find a simple factor structure. Conducting numerical experiments showed good agreement with the results of this method of factor analysis with previously known methods. The proposed method of factor analysis has good and superior efficiency compared with other methods of factor analysis. It has limited capabilities and resources needed to determine the factor structure.

Keywords: k-means method, factor analysis, factor rotation.

Дата поступления в редакцию: 07.11.2018

Математические структуры и моделирование 2018. № 4(48). С. 85–91

УДК 004.622:801.81
DOI: 10.25513/2222-8772.2018.4.85-91

ПРЕДСТАВЛЕНИЕ РАЗМЕТКИ КОРПУСА НАРОДНОЙ РЕЧИ СРЕДНЕГО ПРИИРТЫШЬЯ

Д.Н. Лавров
к.т.н., доцент, e-mail: lavrov@omsu.ru
М.А. Харламова
к.фил.н., доцент, e-mail: khr-spb@mail.ru
Е.А. Костюшина
д.ф.-м.н., доцент, e-mail: kea.omsu@gmail.com

Омский государственный университет им. Ф.М. Достоевского, Омск, Россия

Аннотация. В статье рассматриваются способы репрезентации диалектных записей в региональном корпусе. В центре внимания — модели представления тематической, структурной и отчасти фонетической разметок. Особое внимание уделяется и модели представления экстралингвистических данных. Предложенные решения основаны на представлении реляционных баз данных и формате XML.
Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 18-012-00519.

Ключевые слова: тематическая разметка, метатекстовая разметка, формат XML, региональный диалектный корпус.

Введение

Корпус народной речи Среднего Прииртышья формируется за счёт сбора и последующей расшифровки записанных в экспедициях диалектных текстов. Для хранения полученных данных разрабатывается специализированная информационная система — корпус народной речи.

Ранее в рамках проекта электронного словаря была разработана система для репрезентации фонетических особенностей говоров Среднего Прииртышья [1–3]. В рамках нового проекта — регионального корпуса народной речи — перед коллективом стоят следующие задачи: (1) описать манифестацию в корпусе экстралингвистической информации; (2) описать структурную и тематическую разметки текстов.

1. Экстралингвистическая разметка

Модель экстралингвистической информации после проведённого анализа распадается на две сущности: «Паспорт информанта» и «Паспорт текста». Анализ позволил выделить атрибуты каждой сущности.

Паспорт информанта:

- Фамилия — lname.
- Имя — fname.
- Отчество — sname.
- Пол — gender.
- Год рождения — birth_year.
- Место рождения — birth_location.
- Место рождения родителей — birth_parent_location.
- Кем себя считает — who_i_am.
- Образование — education.
- Род занятий — occupation.
- Говор — dialect.

Паспорт текста:
- Место записи — location.
- Год записи — year.
- Источник (материальный носитель записи) — source.
- Размеченный текст — record.

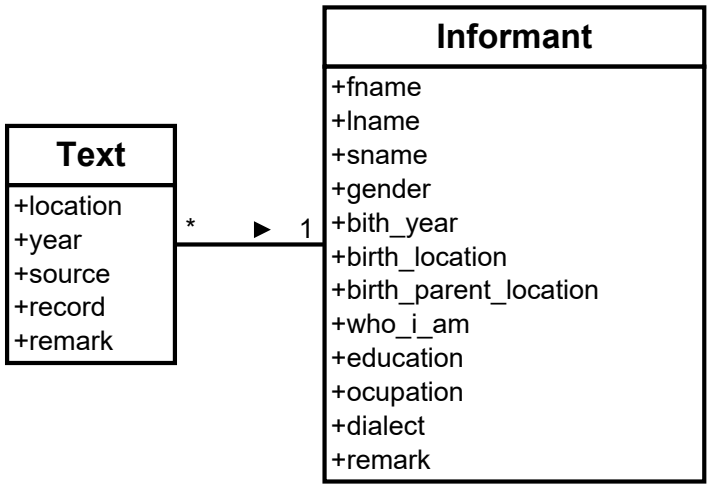


Рис. 1. Модель экстралингвистической информации

2. Фонетические знаки и их представление

В редакторе разметки необходимо ограничить количество вводимых символы для того, чтобы, с одной стороны, показывать фонетику, а с другой — не дозволять ввода служебных для XML символов. Для представления фонетических знаков принято решение использовать utf-8, а также стандартные теги и символы HTML (см. табл. 1).

Таблица 1. Представление фонетических знаков

Фонетический знак	**Внутреннее представление**	**Пример**
ў	&#1118 (или ў в utf-8)	ў-Репинки, хлеў
w	w	krawa, маркowка
h	h	hр'ибы, ahароч'чик
γ	&gamma	Мноγа
’	’	Жен'шина
”	”	ч”иста
^{a}	^a	чисто^{a}
l	l	Былo

3. Структурная разметка

Структурная разметка выполняется на основе формата XML. Для структурной разметки достаточно двух пар тегов: <вопрос>...</вопрос> и <ответ>...</ответ>.

4. Тематическая разметка

Для отображения тематической разметки предлагается использовать русскоязычные теги, название которых совпадают с названиями тем. Темы образуют иерархическую древовидную структуру. Вложения тем друг в друга описываются знаком «:». Так, если в тексте актуализируется тема «родина» и её подтема «деревня», то соответствующий тег будет <родина:деревня>.

Пример внешнего представления разметки Пример размеченного текста с фонетической, структурной и фонетической разметками:

```
<вопрос>А жили вы где? В какой деревне?</вопрос>
<ответ> Юрьифка//</ответ>
<вопрос>
   Сестра сказала, что вы последней оттуда съехали?
</вопрос>
<ответ>
   Да//
   Да/ Пач'ти последняя/
   <жизнь>
       Жалею вот ужэ пятый гот жывy кажэца-и
       жыз'нь и-жывёш
   </жизнь>
   / ни-магy привыкнуть г-гораду//
   <родина:деревня>
      Панимаити ни-магy я привыкнуть/
```

```
      а-там-эт жыла/ деревня свая//
      Природа и-кажэца вот выросла там/
      там радилась /там-и моладась мая
      прахадила/ там детей наражала/
      ну-вот фсё идиал'на// А-время-та
      нашэ како идиал'на-та была //
   </родина:деревня>
</ответ>
```

Обратите внимание на то, что данное представление используется только для отображения на экране в редакторе разметки (так что использование символа «:» на данном этапе некритично), внутреннее представление иное, и о нём пойдёт речь в следующем разделе.

5. Внутренне представление, используемое для обмена данными между приложениями

Предыдущий раздел описывал внешнее представление разметки.

Для визуализации разметки чем короче тег, тем лучше. Это вполне устраивает и разработчиков, и программистов. Казалось бы, почему не использовать это представление и для обмена данными между приложениями?

Есть несколько причин. На уровне спецификаций без внешних описаний только по названию тега невозможно определить, какой это тег — структурный или тематический. Кроме того, в указанном выше представлении вложение тем обозначается двоеточием, что для форматов HTML и XML неприемлемо. В тоже время исследователи-филологи активно его используют при выполнении ручной разметки. Решение состоит в создании внутреннего представления данных, которое будет скрыто от пользователя приложения-редактора.

Принципы, реализованные во внутреннем представлении.

- Все названия тегов — и структурных, и тематических — заменены на английские названия.
- Экстралингвистическая разметка соответствует полям таблиц базы данных (см. рис. 1).
- Структурные теги превращаются в `<question>` и `<answer>`.
- Тематический тег один `<theme class="тема--подтема">`.

Пример представления экстралингвистической, тематической и фонетической разметок во внутреннем формате для обмена данными между разрабатываемыми приложениями (данные вымышленные).

```
<doc>
  <informant>
    <fname>Ольга</fname>
    <sname>Карловна</sname>
    <lname>Карнелс</lname>
    <gender>женский</gender>.
    <birth_year>1930</birth_year>
```

```
  <birth_location>
     д. Новоникольск, жила в д. Баженово
     Тарского района 10 лет
  </birth_location>
  <birth_parent_location>
     родители переехали из Белоруссии, д. Николка
     в 1961 г. в Большие Уки
  </birth_parent_location>
  <who_i_am>
     считает себя «российской»
  </who_i_am>
  <education>4 класса</education>
  <ocupation>
      пенсионерка, сортировщик на почте
  </ocupation>
  <dialect>старожильческий</dialect>
  <remark>
     Год прожила в Казахстане, около 30 лет прожила
     в Таджикистане, 9 лет жила в Новосибирске.
  </remark>
</informant>

<text>
  <location>
     д. Большие Уки Большеуковский район
  </location>
  <year>2005</year>
  <source>
     тетрадь №122, кассета №82,
     записи произведены: Митюшовой Ириной,
     гр. ЯФ - 303, Полозковой Марией, гр. ЯФ – 302.
  </source>
  <remark></remark>
  <record>
    <![CDATA[
    <question>А жили вы где? В какой деревне?</question>
    <answer><b>Ю</b>рьифка//</answer>
    <question>
       Сестра сказала, что вы последней оттуда съехали?
    </question>
    <answer>
      Д<b>а</b>//
      Д<b>а</b>/ Пач’т<b>и</b> посл<b>е</b>дняя/
      <theme class="жизнь">
        Жал<b>е</b>ю вот уж<b>э</b> пятый г<b>о</b>т
        жыв<b>у</b> к<b>а</b>жэца-и ж<b>ы</b>з’нь
        и-жыв<b>ё</b>ш
      </theme>
```

```
        / ни-маг<b>у</b> прив<b>ы</b>кнуть
        г-г<b>о</b>раду//
        <theme class="родина--деревня">
          Паним<b>а</b>ити ни-маг<b>у</b> <b>я</b>
          прив<b>ы</b>кнуть/ а-т<b>а</b>м-эт
          жыл<b>а</b>/ дир<b>е</b>вня сва<b>я</b>//
          Прир<b>о</b>да и-к<b>а</b>жэца в<b>о</b>т
          в<b>ы</b>расла т<b>а</b>м/ т<b>а</b>м
          радил<b>а</b>сь /т<b>а</b>м-и м<b>о</b>ладась
          ма<b>я</b> прахад<b>и</b>ла/ там дит<b>е</b>й
          нараж<b>а</b>ла/ ну-в<b>о</b>т ӧсё
          иди<b>а</b>л'на// А-вр<b>е</b>мя-та н<b>а</b>шэ
          как<b>о</b> иди<b>а</b>л'на-та б<b>ы</b>ла //
        </theme>
      </answer>
      ]]>
    </record>
  </text>
</doc>
```

Использование CDATA позволяет не заботиться о точном соответствии спецификациям XML внутри поля record и без дополнительных преобразований использовать данный код для отображения на HTML-странице web-приложения.

Заключение

В настоящее время на основе разработанной модели представления созданы два прототипа приложений: десктоп-редактор для создания разметки в условиях экспедиций и отсутствия доступа к среде интернет и веб-приложение, позволяющее делать выборку из базы данных на основе MySQL по экстралингвистической информации и отображать её в виде HTML-страниц с возможностью интерактивной тематической разметки. Прототип веб-приложения создан на языке Python с использованием фреймворка Django и библиотеки jQuery.

В момент написания статьи проходило опытное тестирование и апробация указанных прототипов.

Результаты данной статьи были представлены в докладе на конференции «Математическое и компьютерное моделирование» [4].

Благодарности

Выражаем признательность Лапину Александру Петровичу и Черкащенко Илье Александровичу за ценные замечания и помощь в реализации прототипов. Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта №18-012-00519.

Литература

1. Лавров Д.Н., Харламова М.А. Словарь констант народной речи: выбор платформы представления // Вестник Омского университета. 2015. № 1(75). С. 213–216.
2. Харламова М.А. Константы народной речемысли и их лексикографическая интерпретация. Омск : Изд-во Ом. гос. ун-та, 2014. 290 с.
3. Балезин И.А., Лавров Д.Н., Харламова М.А. Архитектура мобильного клиента под iOS для доступа к веб-словарю народной речи Среднего Прииртышья // Математические структуры и моделирование. 2016. № 4(40). С. 133–142.
4. Лавров Д.Н., Харламова М.А., Костюшина Е.А. Модель представления экстралингвистической и тематической разметки в корпусе народной речи // VI-я Междунар. науч. конф. «Математическое и компьютерное моделирование», посвящ. памяти проф. Б.А. Рогозина. 23 ноября 2018. С. 115–118.

REPRESENTATION OF THE CORPUS OF MEDIUM IRTYSH FOLK DIALECT

D.N. Lavrov
Ph.D. (Eng.), Associate Professor, e-mail: lavrov@a.ru
M.A. Kharlamova
Ph.D. (Philological), Associate Professor, e-mail: khr-spb@mail.ru
E.A. Kostushina
Ph.D. (Eng.), Associate Professor, e-mail: kea.omsu@gmail.com

Dosotevsky Omsk State University, Omsk, Russia

Abstract. The article discusses ways of representing dialect entries in the regional corpus. The focus is on models for the presentation of thematic, structural and partly phonetic markings. Particular attention is paid to the presentation model of extralinguistic data. The proposed solutions are based on the representation of relational databases and XML format.
The reported study was funded by RFBR according to the research project № 18-012-00519.

Keywords: thematic markup, metatext markup, XML format, regional dialect body.

Дата поступления в редакцию: 20.11.2018

Математические структуры и моделирование
2018. № 4(48). С. 92–98

УДК 004.056
DOI: 10.25513/2222-8772.2018.4.92-98

О СЛОЖНОСТИ ПОДСИСТЕМ РАЗГРАНИЧЕНИЯ ДОСТУПА КРУПНОМАСШТАБНЫХ ИНФОРМАЦИОННЫХ СИСТЕМ

Н.Ф. Богаченко
к.ф.-м.н., доцент, e-mail: nfbogachenko@mail.ru

Омский государственный университет им. Ф.М. Достоевского, Омск, Россия

Аннотация. Описываются признаки сложных систем. Определяются возможные метрические характеристики этих признаков. Рассматриваются крупномасштабные информационные системы и их подсистемы разграничения доступа. Доказывается принадлежность таких подсистем разграничения доступа классу сложных систем. Обосновывается необходимость автоматизации процессов управления подсистемой разграничения доступа в крупномасштабных информационных системах.

Ключевые слова: сложные системы, разграничение доступа, управление, автоматизация.

Введение

В настоящее время наблюдается рост проблем, возникающих при работе подсистем разграничения доступа крупномасштабных информационных систем, в первую очередь связанных с ошибками проектирования и администрирования политики разграничения доступа. Объясняется это тем, что в силу своих размеров крупномасштабные системы обладают рядом характеристик, из которых следуют особенности разграничения доступа к информации в таких системах. Эти особенности существенно влияют на работу подсистемы разграничения доступа и обосновывают необходимость разработки новых методов и средств управления разграничением доступа к информационным ресурсам с учётом специфики крупномасштабных информационных систем. При этом управление включает в себя такие процессы, как формальный анализ, проектирование, построение, оптимизация и т. д.

В данной статье при помощи признаков и метрик сложных систем обосновывается актуальность задачи построения автоматизированных систем управления подсистемой разграничения доступа к ресурсам крупномасштабных информационных систем.

1. Признаки сложных систем

В системном анализе и в общей теории систем под сложной (большой) системой понимается множество, элементы которого находятся в некоторых

отношениях (связях) друг с другом и которые образуют определённую целостность (единство). В свою очередь, эта целостность обладает новым качеством, не присущим отдельным элементам множества [1]. В рамках развития теории сложных систем в работе [2] сложные (большие) системы определяются как сети, содержащие множество компонент, которые взаимодействуют друг с другом, как правило, нелинейным способом.

Понятие «сложная (большая) система» вводится, в первую очередь, не с целью классифицировать системы на «сложные (большие)» и «простые (небольшие)», а чтобы выделить общие принципы, присущие крупномасштабным системам с учётом их многообразия и сложности. Тем не менее, и в общей теории систем, и в теории сложных систем остаётся открытой проблема формализации критериев сложной системы в виде соответствующих метрик. Наличие метрик сложных систем позволило бы сравнивать и ранжировать системы между собой на основе формальных критериев.

Одним из признаков сложной системы является большое число её элементов — *масштабность*. Но этой характеристики не достаточно для того, чтобы отнести систему к классу сложных. Сложная система — это масштабный комплекс *сложно взаимодействующих* элементов, сбои или потери в одной части системы могут непредсказуемым образом сказаться на системе в целом. Формально сложная система может быть представлена набором

$$S = \langle A, N, R_1, \ldots, R_k \rangle,$$

где $A = \{a_1, \ldots, a_N\}$ — множество элементов, N — мощность множества A, $R_i \subseteq A \times \ldots \times A = A^{m_i}$ $(m_i > 1,\ i = 1, \ldots, k)$ — отношения на множестве A. В простейшем случае $k = 1$, $m_1 = m = 2$. Это означает, что на множестве A определено одно бинарное отношение R. Нередко такое отношение обладает свойствами антисимметричности и транзитивности, то есть является отношением порядка.

Согласно [3], «сложные системы часто являются иерархическими и состоят из взаимосвязанных подсистем, которые в свою очередь также могут быть разделены на подсистемы, и т. д., вплоть до самого низкого уровня». Этот признак назовём *возможностью декомпозиции*:

1) элемент сложной системы является системой более низкого порядка сложности:

$$a_i = \langle A_i, N_i, R_{i1}, \ldots, R_{ik_i} \rangle;$$

2) сложная система является элементом системы более высокого порядка сложности:

$$S \in \{S_1, \ldots, S_\nu\}.$$

Для сложной системы характерна *гетерогенность* (*разнородность*) элементов множества A. Данный признак хорошо укладывается в концепцию представления элементов сложной системы в виде подсистем более низкого порядка сложности: множества A_i $(i = 1, \ldots, N)$ могут иметь различную природу элементов, то есть принадлежать разным универсумам $U_1, \ldots, U_n$, $0 < n << N$.

Для информационных и технических систем значимым и даже определяющим является ещё один признак: *иерархическая структура управления* системой [4]. Это означает, что на множестве субъектов управления задаётся отношение порядка — первый шаг к увеличению сложности уже самой системы управления.

Указанные качественные признаки сложных систем и возможные количественные характеристики (метрики) представлены в табл. 1. Первые четыре метрики позволяют сравнивать сложные системы между собой. Для ранжирования сложных систем целесообразно использовать дополнительную метрику — порядок сложности системы. Данная характеристика является относительной и определяется допустимым уровнем абстракции.

Таблица 1. Характеристики сложных систем

	Признак	*Метрика*
1.	Масштабность	Число элементов N
2.	Сложность взаимодействия	Число связей, определяемое как сумма мощностей всех заданных на множестве A отношений: $\|R_1\| + \ldots + \|R_k\|$
3.	Разнородность (гетерогенность)	Число универсумов n
4.	Иерархия управления	Метрические характеристики теоретико-графового представления отношения порядка на множестве субъектов управления
5.	Возможность декомпозиции	Порядок сложности

Определение сложной системы по-прежнему возможно дать лишь на качественном уровне в виде следующих достаточных условий.

Предположение 1. *Система относится к классу сложных систем, если она обладает следующими признаками: масштабностью, сложностью взаимодействия, разнородностью, иерархией управления и возможностью декомпозиции.*

Следует отметить, что для сложных *информационных* систем возможность декомпозиции заложена в само понятие алгоритма и реализована в современных языках программирования. Поэтому далее, рассматривая сложные информационные системы, этот признак будем считать присутствующим по умолчанию.

В силу своих признаков сложная система характеризуется нехваткой ресурсов для эффективного управления [5]. Поэтому в таких системах должна быть подсистема принятия решения, что обосновывает следующий постулат.

Предположение 2. *Управление сложной системой осуществляется на основе совместного участия человека и автоматизированной системы управления.*

2. Управление политикой разграничения доступа

В рамках проблемы проектирования и администрирования политики разграничения доступа в современных информационных системах рассмотрим иерархию сложности систем и место подсистемы разграничения доступа в этой иерархии (см. рис. 1).

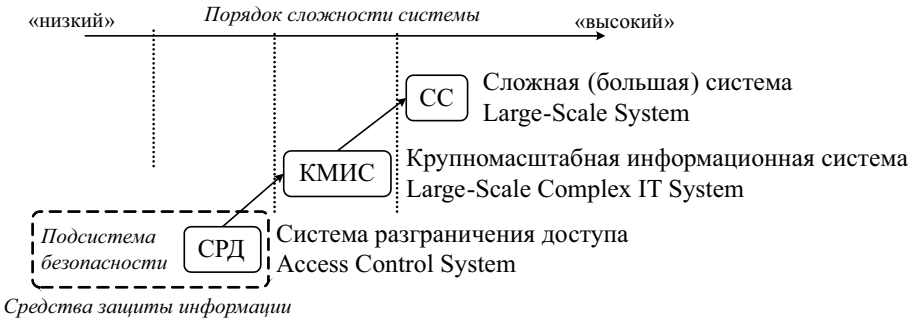


Рис. 1. Порядок сложности систем

Примерами сложных систем (Large-Scale Systems) наиболее высокого порядка являются транспортные, экономические, биологические, социальные и др. системы. Для их изучения и возможности управления разрабатываются автоматизированные системы управления или, в более широком смысле, информационные системы.

Информационные системы, также удовлетворяющие требованиям, предъявляемым к сложным системам, принято называть крупномасштабными информационными системами (КМИС, Large-Scale Complex IT Systems).

Далее наблюдается в некотором смысле самоподобный процесс (процесс метауправления[1]): с ростом сложности информационных систем возрастает сложность управления ими. В связи с этим модели сложных систем приходится использовать уже в управлении.

В рамках подсистемы безопасности КМИС выделяется система разграничения доступа (СРД, Access Control System), которая реализует политику разграничения доступа к ресурсам информационной системы. СРД относится к информационной системе более низкого порядка сложности, чем КМИС в целом. Но современные программно-технические комплексы таковы, что их СРД также обладают признаками сложной системы. В работе [6] выделены следующие основные особенности разграничения доступа в КМИС:

1) существенное увеличение числа пользователей и объектов;
2) распределённость и иерархичность объектов доступа;
3) изменяемость правил разграничения доступа;
4) необходимость совмещения политик разграничения доступа.

[1] Метауправление (metamanagement) — управление самой системой управления, призванное, во-первых, обеспечить согласованную и эффективную работу подсистем и элементов системы управления и, во-вторых, обеспечить в случае необходимости её изменение (развитие).

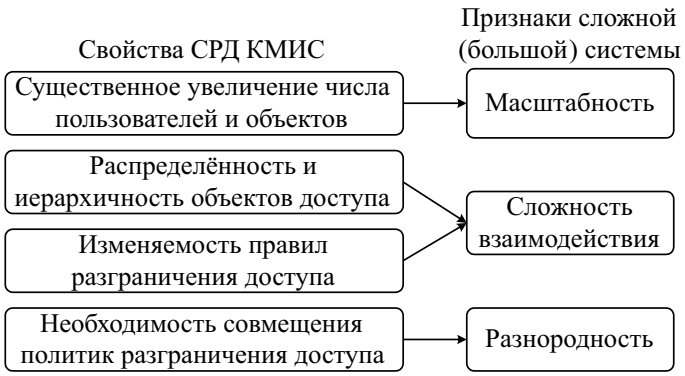


Рис. 2. Характеристика СРД КМИС как сложной (большой) системы

Эти особенности являются основными свойствами СРД КМИС и характеризуют её как сложную систему (см. рис. 2). В том случае, когда управление СРД, то есть управление самой политикой разграничения доступа, уже не может осуществляться одним или несколькими не взаимодействующими между собой администраторами безопасности, возникает необходимость в иерархии управления. Востребованность иерархической структуры администрирования СРД КМИС подтверждена как теоретическими, так и практическими решениями. В частности, иерархия управления разграничением доступа заложена в ролевую модель в виде иерархии административных ролей [7], а набор иерархически организованных административных ролей в современном программном обеспечении нередко задаётся «по умолчанию» [8, 9].

Согласно предположению 1, СРД с иерархией управления принадлежит классу сложных систем. Для таких СРД предположение 2 обосновывает востребованность автоматизированных систем управления политикой разграничения доступа — систем управления разграничением доступа (СУРД, Access Control Management Systems). Эти системы на первом этапе своего развития могут представлять собой системы поддержки принятия решений и позволят автоматизировать процессы проектирования и администрирования политики разграничения доступа КМИС.

Подводя итог, ещё раз отметим, что для управления сложной системой разрабатываются КМИС, для управления КМИС используются в частности СРД, а для управления СРД востребованы СУРД в том случае, когда возникает необходимость в иерархии управления политикой разграничения доступа (см. рис. 3).

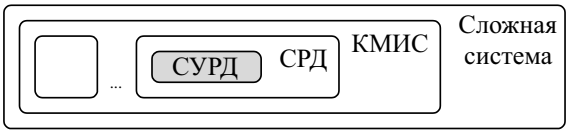


Рис. 3. Место системы управления разграничением доступа в иерархии сложности систем

Заключение

В настоящее время предлагаются программно-технические решения в области управления разграничением доступа к информационным ресурсам — системы управления идентификационными данными и доступом пользователей (Identity and Access Management System). Как показано в [6], основной недостаток таких систем заключается в том, что большинство операций по управлению политикой разграничения доступа перекладывается на «плечи» администратора безопасности или задаётся «по умолчанию» и через шаблонные решения. Это приводит к серьёзным проблемам в случае, когда подсистема разграничения доступа относится к классу сложных систем. По сути, полноценной автоматизированной управляющей надстройки над системой разграничения доступа не происходит. Вышесказанное обосновывает актуальность задачи разработки и развития СУРД.

Остаётся открытым вопрос об оценке эффективности внедрения СУРД. Возможно предложить следующие критерии: сложность администрирования и защищённость КМИС. Сложность информационной системы может пониматься как число элементарных операций, необходимых для получения результата при любом допустимом входном наборе данных. Если в качестве меры сложности управления СРД определить число элементарных операций, выполняемых администратором безопасности, то, очевидно, автоматизация процессов управления существенно снизит сложность администрирования. Возможно сопоставление числа настраиваемых параметров, которые администратором безопасности задаются «в ручном режиме», до и после внедрения СУРД в зависимости от масштабов информационной системы. Чтобы оценить, повысилась ли защищённость КМИС при переходе от «ручного» управления политикой разграничения доступа к СУРД, следует проанализировать, на сколько уменьшилось количество ошибок администрирования, связанных с управлением доступом к информационным ресурсам.

Литература

1. Цветков В.Я. Систематика сложных систем // Современные технологии управления. 2017. № 7(79). URL: `https://sovman.ru/article/7903/` (дата обращения: 11.11.2018).
2. Sayama H. Introduction to the Modeling and Analysis of Complex Systems. Open SUNY Textbooks, Milne Library. State University of New York at Geneseo, 2015. 498 p.
3. Буч Г. Объектно-ориентированный анализ и проектирование с примерами приложений на C++. М. : Бином. Лаборатория знаний, 2001. 560 с.
4. Володина А.А., Лёвкин И.М. Адаптивный подход к защите информации в больших информационных системах // Проблема комплексного обеспечения информационной безопасности и совершенствование образовательных технологий подготовки специалистов силовых структур : межвузовский сборник трудов VI Всероссийской научно-технической конференции КОНФИБ'15. СПб. : Университет ИТМО, 2016. С. 65–73.

5. Казиев В.М. Введение в системный анализ и моделирование. URL: `http://bigc.ru/theory/books/kvisam/` (дата обращения: 11.11.2018).
6. Богаченко Н.Ф. Анализ проблем управления разграничением доступа в крупномасштабных информационных системах // Математические структуры и моделирование. 2018. № 2(46). С. 135–152.
7. Sandhu R.S., Coyne E.J., Feinstein H.L., Youman C.E. Role-Based Access Control Models // IEEE Computer. 1996. No. 29(2). P. 38–47.
8. Основы ролевого администрирования для System Center Configuration Manager. URL: `https://msdn.microsoft.com/ru-ru/library/mt592917.aspx` (дата обращения: 11.11.2018).
9. Административные роли. URL: `https://helpx.adobe.com/ru/enterprise/using/admin-roles.html` (дата обращения: 11.11.2018).

ON THE COMPLEXITY OF ACCESS CONTROL SUBSYSTEMS OF LARGE-SCALE COMPLEX IT SYSTEMS

N.F. Bogachenko
Ph.D. (Phys.-Math.), Associate Professor, e-mail: nfbogachenko@mail.ru

Dostoevsky Omsk State University, Omsk, Russia

Abstract. Signs of large-scale systems are described. Possible metric characteristics of these signs are defined. Large-scale complex IT systems and their access control subsystems are considered. Belonging of such access control subsystems to a class of larged-scale systems is proved. The automation need of the management processes by the access control subsystem in the large-scale complex IT system is explained.

Keywords: larged-scale systems, access control, management, automation.

Дата поступления в редакцию: 12.11.2018

Математические структуры и моделирование
2018. № 4(48). С. 99–112

УДК 004.7:004.056
DOI: 10.25513/2222-8772.2018.4.99-112

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ДЛЯ МОДЕЛИРОВАНИЯ СЕТИ И ИМИТАЦИИ АТАК НА КОМПЬЮТЕРНУЮ СЕТЬ

А.В. Баженов
студент, e-mail: dr.bazhenoff2017@yandex.ru
А.К. Гуц
д.ф.-м.н., профессор, e-mail: guts@omsu.ru

Омский государственный университет им. Ф.М. Достоевского, Омск, Россия

Аннотация. В статье представлено программное приложение, позволяющее моделировать компьютерные сети и атаки на них.

Ключевые слова: программное приложение, моделирование, сетевые атаки, smurfing.

1. Введение

Целью данной работы является разработка программного обеспечения для моделирования различных сетевых атак. Такое приложение полезно в первую очередь для обучения студентов и администраторов, которые по долгу службы обязаны обеспечивать безопасность информационных ресурсов.

Для достижения цели требуется решение следующих задач:

1. Анализ существующих видов сетевых атак.
2. Разработка программных библиотек для имитации основных сетевых устройств.
3. Разработка пользовательского интерфейса для работы с программной библиотекой.
4. Тестирование разработанного программного комплекса с помощью демонстрации реализации сетевой атаки (в статье представлена имитация атаки Smurfing).

Данное исследование продолжает разработки по моделированию атак на компьютерные сети посредством создания специализированного программного обеспечения, начатые в [1]. Приложение не предусматривает демонстрацию защиты от сетевых атак.

2. Основные понятия

2.1. Сетевой протокол

Сетевой протокол — это набор правил и соглашений, который определяет единообразный способ передачи информации и обработки ошибок при взаимо-

действии и позволяет осуществлять соединение и обмен данными между двумя и более включёнными в сеть устройствами [2].

Каждый сетевой протокол работает на определённом уровне модели OSI. Протоколы высшего уровня строятся поверх протоколов более низкого уровня.

2.2. Уязвимость

В компьютерной безопасности термин «уязвимость» (англ. vulnerability) используется для обозначения недостатка в системе, используя который можно намеренно нарушить её целостность и вызвать неправильную работу. Уязвимость может быть результатом ошибок программирования, недостатков, допущенных при проектировании системы, ненадёжных паролей, вирусов и других вредоносных программ, скриптовых и SQL-инъекций. Некоторые уязвимости известны только теоретически, другие же активно используются и имеют известные эксплойты.

Метод информирования об уязвимостях является одним из пунктов спора в сообществе компьютерной безопасности. Некоторые специалисты отстаивают немедленное полное раскрытие информации об уязвимостях, как только они найдены. Другие советуют сообщать об уязвимостях только тем пользователям, которые подвергаются наибольшему риску, а полную информацию публиковать лишь после задержки или не публиковать совсем. Такие задержки могут позволить тем, кто был извещён, исправить ошибку при помощи разработки и применения патчей, но также могут и увеличивать риск для тех, кто не посвящён в детали.

Обычно уязвимость позволяет атакующему «обмануть» приложение — заставить его совершить действие, на которое у того не должно быть прав. Это делается путём внедрения каким-либо образом в программу данных или кода в такие места, что программа воспримет их как «свои». Некоторые уязвимости появляются из-за недостаточной проверки данных, вводимых пользователем, и позволяют вставить в интерпретируемый код произвольные команды (SQL-инъекция, XSS, SiXSS). Другие уязвимости появляются из-за более сложных проблем, таких как запись данных в буфер без проверки его границ (переполнение буфера) [3]. Атаки, рассмотренные в данной работе, как правило, используют недостатки сетевых протоколов.

3. Сетевые атаки

3.1. Метод оценки уязвимости

При рассмотрении атак будет использоваться система оценок CVSSv3.

Стандарт Common Vulnerability Scoring System был разработан группой экспертов по безопасности National Infrastructure Advisory Council. В эту группу вошли эксперты из различных организаций, таких как CERT/CC, Cisco, DHS/MITRE, eBay, IBM Internet Security Systems, Microsoft, Qualys, Symantec.

CVSS предлагает простой инструментарий для расчёта числового показателя по десятибалльной шкале, который позволяет специалистам по безопасности оперативно принимать решение о том, как реагировать на ту или иную уязвимость. Чем выше значение метрики, тем более оперативная реакция требуется.

Использование метрик CVSS для оценки уязвимостей закреплено в стандартах PCI DSS и СТО БР ИББС [4].

3.2. Фрагментация данных

При передаче пакета данных протокола IP по сети может осуществляться деление этого пакета на несколько фрагментов. Впоследствии, при достижении адресата, пакет восстанавливается из этих фрагментов. Злоумышленник может инициировать посылку большого числа фрагментов, что приводит к переполнению программных буферов на приёмной стороне и, в ряде случаев, к аварийному завершению работы системы [5].

Оценка CVSSv3: 7.5 [6].

3.3. Ping flooding

Данная атака требует от злоумышленника доступа к быстрым каналам в интернете.

Программа ping посылает ICMP-пакет типа ECHO REQUEST, выставляя в нём время и его идентификатор. Ядро машины-получателя отвечает на подобный запрос пакетом ICMP ECHO REPLY. Получив его, ping выдаёт скорость прохождения пакета.

При стандартном режиме работы пакеты высылаются через некоторые промежутки времени, практически не нагружая сеть. Но в «агрессивном» режиме поток ICMP echo request/reply-пакетов может вызвать перегрузку небольшой линии, лишив её способности передавать полезную информацию [5].

Оценка CVSSv3: 5.9 [6].

3.4. Smurfing

Атака заключается в передаче в сеть широковещательных ICMP запросов от имени компьютера-жертвы. В результате компьютеры, принявшие такие широковещательные пакеты, отвечают компьютеру-жертве, что приводит к существенному снижению пропускной способности канала связи и, в ряде случаев, к полной изоляции атакуемой сети. Эта атака исключительно эффективна и широко распространена [5].

Оценка CVSSv3: 7.5 [6].

3.5. DNS Cache Poisoning

Для осуществления атаки атакующий использует уязвимость в конфигурации DNS. Если сервер не проверяет ответы DNS на корректность, чтобы убедиться в их авторитетном источнике, он будет кэшировать некорректные

ответы локально и использовать их для ответов на запросы других пользователей, пославших такие же запросы.

Данная техника может использоваться для того, чтобы перенаправить клиентов на другой сайт по выбору атакующего. Например, при помощи спуфинга можно направить клиента на DNS-сервер, который выдаст заведомо неверный IP-адрес сайта и таким образом направит адресата на сервер, контролируемый злоумышленником [5].

Оценка CVSSv3: 9.0 [6].

3.6. Навязывание пакетов

Злоумышленник отправляет в сеть пакеты с ложным обратным адресом. С помощью этой атаки злоумышленник может переключать на свой компьютер соединения, установленные между другими компьютерами. При этом права доступа злоумышленника становятся равными правам того пользователя, чьё соединение с сервером было переключено на компьютер злоумышленника [5].

Оценка CVSSv3: 9.0 [6].

3.7. SYN Flood

Обычно, когда клиент пытается установить TCP-соединение с сервером, они обмениваются сообщениями по следующей схеме:

1. Клиент запрашивает соединение, посылая SYN-пакет.
2. Сервер подтверждает полученный запрос, отвечая SYN-ACK-пакетом.
3. Клиент, подтверждая, что соединение установлено, отправляет ACK-пакет.

Данная атака основывается на отправке множества SYN-пакетов с разных IP-адресов, которые могут быть даже не закреплены за кем-то или закреплены за устройствами, не принимающими участие в атаке.

Для каждого полученного SYN-пакета сервер выделяет место в буфере до тех пор, пока не получит ACK-ответ. Таким образом, атака может заполнить буфер сервера так, что другие попытки установить с ним соединение не увенчаются успехом из-за отсутствия свободного места в буфере.

Оценка CVSSv3: 5.9 [6].

3.8. UDP Flood

Данная атака основывается на отправке большого количества UDP-пакетов на случайные порты удалённого хоста. Так как UDP-протокол не требует установления соединения, хост-жертва сразу будет проверять наличие приложений, на порты которых отправлены UDP-пакеты. Так как выбор портов случаен, наверняка, большая часть из них будет не привязана к какому-то приложению, и хост будет отправлять ICMP Destination Unreachable.

Таким образом, жертва будет занята отправкой ICMP-пакетов, а не обработкой реальных запросов.

Оценка CVSSv3: 7.0 [6].

3.9. Land Attack

Атака заключается в отправке хосту пакетов с идентичными адресами отправителя и получателя. Может возникнуть бесконечная петля обращений хоста жертвы к самой себе.

Оценка CVSSv3: 5.9 [6].

3.10. ARP Spoofing

Данная атака основывается на проблеме аутентификации в ARP-протоколе. Любой хост в сети может отправить ARP-пакет, который ассоциирует любой IP-адрес с MAC-адресом злоумышленника, таким образом, хост-жертва будет думать, что отправляет пакеты одному хосту, но на самом деле эти пакеты будет получать хост с заданным MAC-адресом.

Оценка CVSSv3: 8.8. [6].

3.11. MAC Flooding

Данная атака основывается на отправке коммутатору большого количества Ethernet-кадров, которые содержат различные MAC-адреса отправителя. Свитч заносит все эти MAC-адреса в свою таблицу, тем самым забивая её. Если таблица забита, коммутатор работает в режиме хаба: отправляет пакет на все интерфейсы, кроме того, откуда пришёл пакет. Таким образом можно реализовывать другие атаки в рамках целых сетей, к которым подключён свитч.

Оценка CVSSv3: 8.6.[6].

3.12. IP Spoofing

Данная атака заключается в изменении адреса отправителя в заголовке IP-пакета. Хост-жертва будет думать, что получает пакеты от одного хоста, а на самом деле — совершенно от другого. Данная атака позволяет обойти защиту по белому списку, анонимизировать злоумышленника или перенаправить трафик в сети.

Оценка CVSSv3: 7.5. [6].

3.13. MAC Spoofing

Эта атака аналогична атаке IP Spoofing, кроме того, что используется на более низком уровне модели OSI.

Оценка CVSSv3: 7.5. [6].

3.14. Sniffing

Данная атака основывается на специальном программном обеспечении для перехвата и анализа сетевого трафика. Сниффер может анализировать весь трафик, который проходит через сетевую карту. Внутри одного сегмента сети Ethernet-трафик рассылается всем машинам, таким образом, в такой сети весь трафик можно прослушивать.

Данную атаку можно легко сочетать с другими атаками, например, IP Spoofing и MAC Spoofing.

Оценка CVSSv3: 8.6 [6].

4. Разработка программной библиотеки для имитации основных сетевых устройств

Для разработки программной библиотеки был выбран язык $C\sharp$, так как он позволяет сконцентрироваться на абстракциях, а не на деталях реализации.

Приложение[1] работает под Windows с Microsoft.Net Framework 4.5.3+ и Linux (MacOS) с последней версией Mono Framework. Операционная система 64-битная, но запускается и на x86. Файл запуска программы UserInterface.exe.

4.1. Разработка механизма передачи сообщений

Для передачи сообщений было решено использовать протокол UDP, который позволяет отправлять данные без предварительного открытия соединения. Был реализован класс UdpClient, который содержит в себе экземпляр класса System.Net.Sockets.UdpClient и хранилище пар IP:Port для всех, кто отправлял сообщения по адресу нашего UdpClient:

```
private readonly System.Net.Sockets.UdpClient client;
protected readonly ISet<IPEndPoint> _connections = new
HashSet<IPEndPoint>();
```

У данного класса были реализованы методы для отправки сообщения по IP-адресу, Broadcast отправки, а также получения сообщения:

```
public void Send(string ipAddress, string message) {
var port = ipAddress.HashCode();
var endpoint = new IPEndPoint(IPAddress.Parse("127.0.0.1"), port);
var data = Encoding.UTF8.GetBytes(message);
client.Send(data, data.Length, endpoint);
}
public void SendBroadcast(string message) {
foreach (var endpoint in _connections)
{
var data = Encoding.UTF8.GetBytes(message);
```

[1]Приложение доступно по адресу: http://fkn.univer.omsk.su/teaching/Simulator/Bazhenov.zip

```
client.Send(data, data.Length, endpoint);
}
}
public async Task<Received> Receive() {
var result = await client.ReceiveAsync();
_connections.Add(result.RemoteEndPoint);
return new Received
{
Message = Encoding.UTF8.GetString(result.Buffer, 0,
result.Buffer.Length),
Sender = result.RemoteEndPoint
};
} [7].
```

4.2. Разработка модели физического интерфейса сетевого устройства

Был разработан класс NetworkInterface, который наследует функционал UdpClient и содержит поля, хранящие его IP-адрес и тип устройства, чей это интерфейс.

4.3. Разработка моделей компьютера и компьютера злоумышленника

Указанные классы отличаются лишь тем, что компьютер злоумышленника может отправлять сообщения, редактируя адрес отправителя, поэтому описания компьютера будет достаточно.

Класс компьютера определяется тремя параметрами: IP-адрес, IP-адрес сетевого устройства, к которому он подключён, и UdpClient для отправки сообщений.

При проектировании был принят унифицированный формат сообщений: “IP_источника/IP_назначения/сообщение/Тип_Устройства_Отправителя”.

При создании экземпляра класса Computer или HackerComputer необходимо указать адрес шлюза по умолчанию, это может быть адрес интерфейса маршрутизатора или коммутатора, на этот адрес будет отправлен пакет SYN и ожидаться ACK-подтверждение.

Компьютер умеет принимать и обрабатывать сообщения, а также ведёт журнал, в котором отображается, какие пакеты и от кого он получает. Для обработки сообщений было решено использовать асинхронную модель: создаётся поток, который обрабатывает пришедшие пакеты.

4.4. Разработка модели маршрутизатора

Был разработан класс Router, который имеет набор экземпляров класса NetworkInterface, таблицу маршрутизации и очередь для приходящих сообщений.

Для каждого интерфейса запускается отдельный поток, который ожидает сообщения.

Когда сообщение получено, этот поток его обрабатывает:

1. Если сообщение SYN, то отправляется ACK-ответ, а пара адрес отправителя – интерфейс заносится в таблицу маршрутизации.
2. Если сообщение DIST (означает, что оно содержит информацию о сетях, отправленную другим сетевым устройством), то роутер добавляет пары сеть – интерфейс в свою таблицу маршрутизации.
3. Если это широковещательное сообщение, то оно отправляется на все интерфейсы маршрутизатора.
4. Если сообщение не одно из выше перечисленных, то оно заносится в очередь.

Отдельным потоком работает функция маршрутизации: ждёт, пока в очереди появится сообщение, далее ищет адрес назначения в таблице маршрутизации и, если находит, — отправляет на нужный интерфейс, причём если есть несколько подходящих интерфейсов, то сообщение отправляется на любой, не являющийся роутером, чтобы избежать петель маршрутизации.

```
    private void StartRouting() =>
    Task.Factory.StartNew(async () => {
    while (true) {
    var msg = await queue.ReceiveAsync();
    var target = msg.Split('/')[1];
    if (routingTable.ContainsKey(target)) {
    var I = routingTable[target];
    var i = interfaces.OrderBy(x => x.Type ==
    NetworkDeviceType.ROUTER).First();
    interface.Send(msg);
    }
}
    }); [3]
```

Был реализован функционал рассылки таблицы маршрутизации на все интерфейсы, к которым подключён другой роутер. Сообщение с рассылаемой таблицей маршрутизации в адресе назначения имеет широковещательный адрес. Для рассылки используется отдельный поток, который повторяет её каждые 0,1 секунды, чтобы сходимость в сети была быстрая.

```
    private void StartTableDistribution() =>
    Task.Factory.StartNew(() => {
    while (true) {
    Interfaces.Where(i => i.Type ==
    NetworkDeviceType.ROUTER).ToList()
    .ForEach(i => {
    var serializedTable = Serialize(routingTable);
    var interfaces = string.Join("#
```

```
Interfaces.Select(x => x.IpAddress));
var routingInfo = Merge(serializedTable,
interfaces);
i.Send($"{i.IpAddress}/255.255.255.255/DIST
/{NetworkDeviceType.ROUTER}/
{routingInfo}");
});
Thread.Sleep(100);
}
});
```

4.5. Разработка модели коммутатора

Был разработан класс Switch, работающий по принципу, схожему с Router. Отличие состоит в том, что коммутатор не получает рассылку таблицы маршрутизации от роутеров и, если пришедший пакет имеет адрес назначения, которого нету в его таблице маршрутизации, то свитч отправит этот пакет на все порты, кроме того, с которого он пришёл.

Коммутатор и маршрутизатор имеют метод Connect, который имитирует подключение между своим портом и указанным портом другого сетевого устройства. Этот метод использует двойное рукопожатие для установления соединения.

```
public void Connect(string sourceIp, string destinationIp) {
var srcInterface = Interfaces.First(i => i.IpAddress == sourceIp);
srcInterface.Send(destinationIp, $"{sourceIp/destinationIp}/
{PacketType.HANDSHAKE.SYN}/{NetworkDeviceType.Switch
}"
);
}
```

5. Разработка модели пользовательского интерфейса

Так как для разработки программной библиотеки был использован язык $C\sharp$, то для реализации пользовательского интерфейса лучше всего подойдёт Windows Forms [3] фреймворк.

5.1. Разработка основного окна

Основное окно должно включать функционал для создания окон остальных сетевых элементов. Для создания компьютера и компьютера-хакера: поля для ввода собственного IP-адреса и IP-адреса маршрута по умолчанию, а также кнопка. Для маршрутизатора и коммутатора — окно для ввода IP-адресов интерфейсов (см. рис. 1).

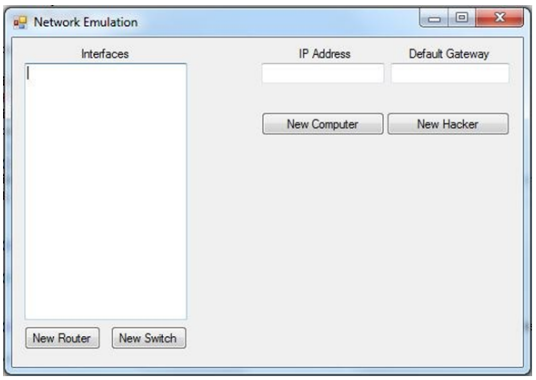


Рис. 1. Основное окно

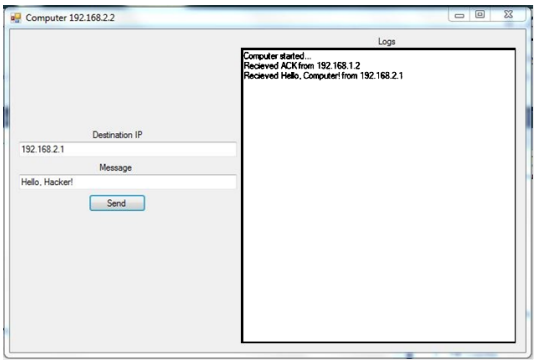


Рис. 2. Окно компьютера

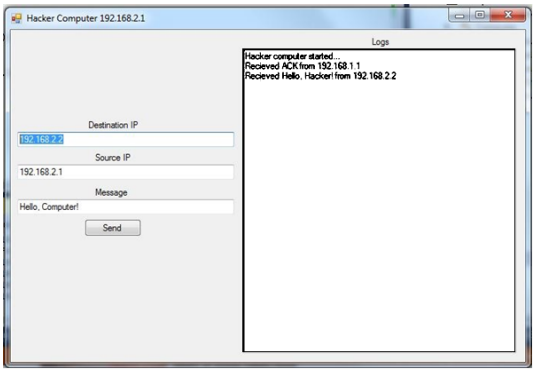


Рис. 3. Окно компьютера-злоумышленника

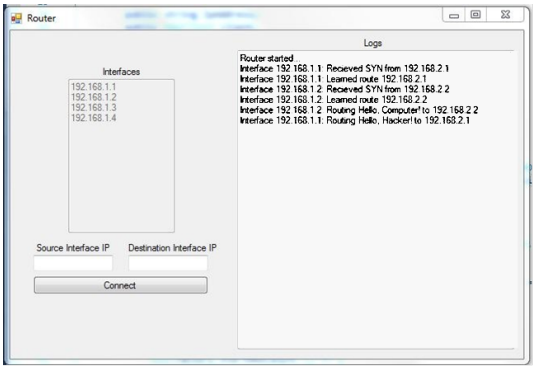


Рис. 4. Окно маршрутизатора

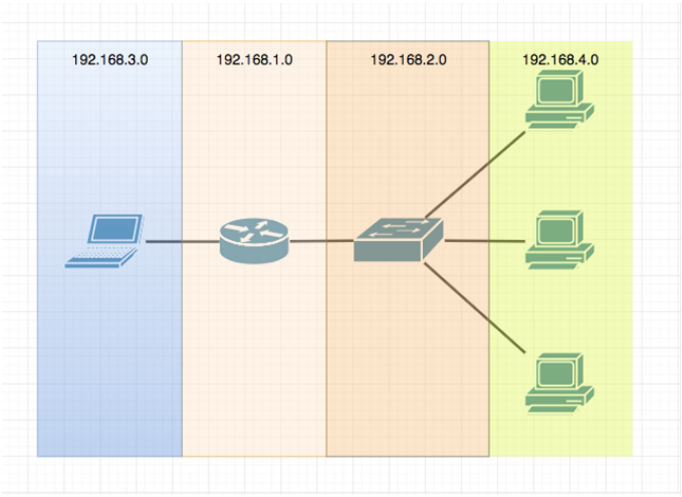


Рис. 5. Топология сети

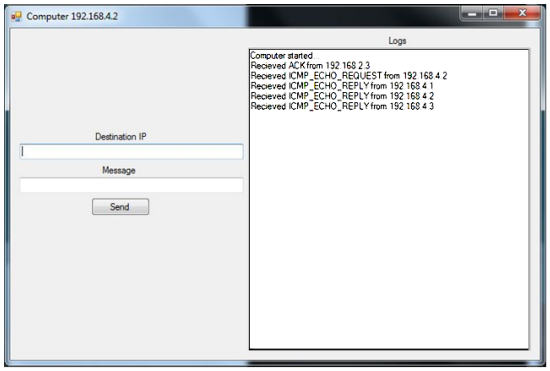


Рис. 6. Окно логгирование у компьютера с IP адресом 192.168.4.2

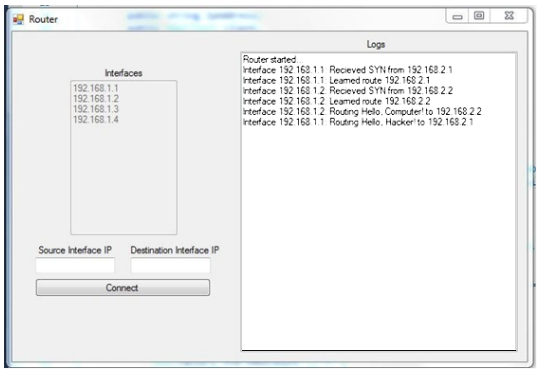


Рис. 7. Окно маршрутизатора после атаки

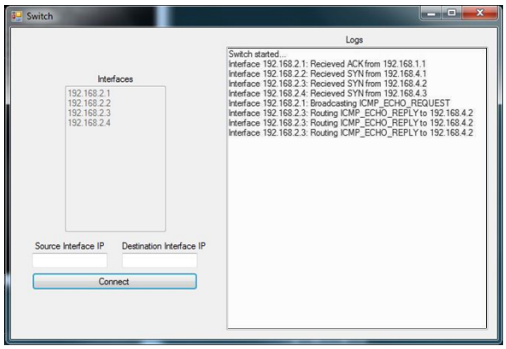


Рис. 8. Окно коммутатора после атаки

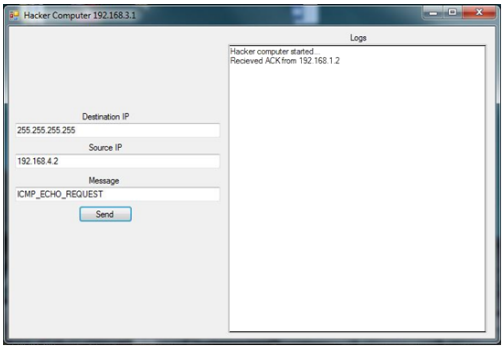


Рис. 9. Окно компьютера-злоумышленника после атаки

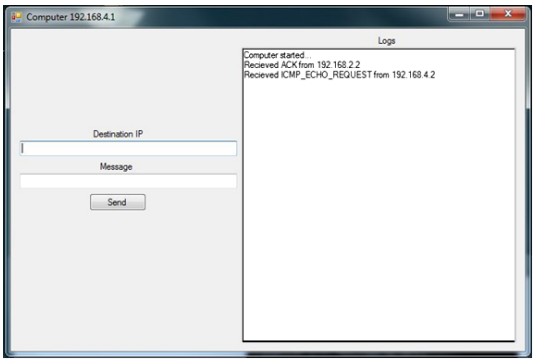


Рис. 10. Окно компьютера 192.168.4.1 после атаки

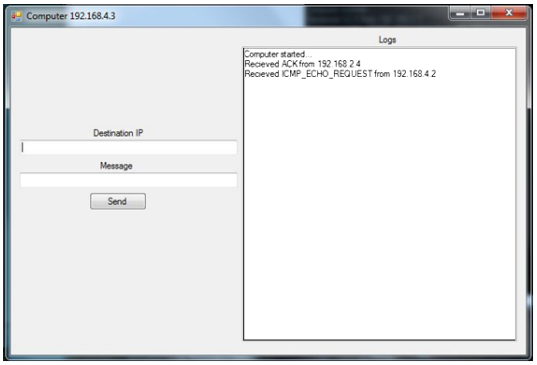


Рис. 11. Окно компьютера 192.168.4.3 после атаки

5.2. Разработка окна компьютера

Окно компьютера должно иметь функционал для отправки сообщения по IP-адресу: поля для ввода IPадреса назначения и сообщения, а также кнопка “Send”. Для удобства и понимания того, что происходит с компьютером, было создано окно с логгированием событий (см. рис. 2).

5.3. Разработка модели окна компьютера-злоумышленника

Компьютер злоумышленника должен иметь тот же функционал, что и обычный компьютер, но должен иметь возможность отправлять сообщения с произвольного IP-адреса (см. рис. 3).

5.4. Разработка моделей окон маршрутизатора и коммутатора

Для функционирования сети нужна возможность соединять сетевые устройства друг с другом. Для этого предусмотрены два окна ввода для IP-адресов: собственного интерфейса и интерфейса другого сетевого устройства, а также кнопка “Connect”.

Также предусмотрены окна для демонстрации списка IP-адресов собственных интерфейсов и отображения системных сообщений маршрутизатора (см. рис. 4).

Окно коммутатора внешне не отличается от окна маршрутизатора.

6. Демонстрация работоспособности программного обеспечения на примере простейшей сетевой атаки

В качестве атаки используем Smurfing, так как она довольно проста в реализации, но, тем не менее, отлично покажет работоспособность программного обеспечения на примере реальной сети. Будем имитировать сеть со следующей топологией (см. рис. 5).

Шаг 1. Создать маршрутизатор с двумя интерфейсами, которые имеют IP-адреса 192.168.1.1 и 192.168.1.2.

Шаг 2. Создать коммутатор с четырьмя интерфейсами, которые имеют IP-адреса 192.168.2.1, 192.168.2.2, 192.168.2.3, 192.168.2.4.

Шаг 3. Подключить коммутатор интерфейсом с IP адресом 192.168.2.1 к интерфейсу маршрутизатора с IP-адресом 192.168.1.1.

Шаг 4. Создать компьютер злоумышленника, используя IP-адрес 192.168.3.1 и шлюз по умолчанию 192.168.1.2.

Шаг 5. Создать три компьютера с IP-адресами 192.168.4.1, 192.168.4.2, 192.168.4.3 и шлюзами 192.168.2.2, 192.168.2.3, 192.168.2.4 соответственно.

Шаг 6. Отправить сообщение ICMP_ECHO_REQUEST с адресом отправителя 192.168.4.2 и широковещательным адресом назначения 255.255.255.255 с компьютера злоумышленника.

Шаг 7. Теперь в окне логгирования у компьютера с IP-адресом 192.168.4.2 три сообщения ICMP ECHO REPLY (см. рис. 6), это значит, что все компьютеры в сети получили ICMP ECHO REQUEST сообщение и отправили ICMP ECHO REPLY по IP-адресу отправителя, которым был указан хост 192.168.4.2.

Окно маршрутизатора после атаки представлено на рис. 7, а окно коммутатора после атаки — на рис. 8.

Окно компьютера-злоумышленника после атаки представлено на рис. 9, а окно компьютера 192.168.4.1 после атаки — на рис. 10.

Окно компьютера 192.168.4.3 после атаки — см. рис. 11.

Заключение

В этой работе были описаны некоторые атаки на уязвимости сетевых протоколов. Знание принципов этих атак поможет администратору избежать значительных моральных и финансовых потерь.

Представлена программная библиотека на языке $C^\sharp$, которая позволяет моделировать реальные масштабные сети. Она содержит классы для имитации работы настоящих компьютеров, маршрутизаторов и коммутаторов как поодиночке, так и в связке друг с другом.

Был разработан пользовательский интерфейс, который позволяет любому человеку, не вникая в детали реализации, построить сеть любых размеров. Был смоделирован прототип реальной сети, на который успешно проведена атака Smurfing.

Разработанное программное обеспечение не накладывает на программиста никаких ограничений и может быть использовано для дальнейшей реализации сетевых протоколов и имитации различных атак на компьютерные сети.

Другой подход по моделированию атак на компьютерные сети разработан в [8]. Был разработан прототип, т. е. черновая реализация мультиагентного компьютерного симулятора атак. Каждый из компонентов симулятора атаки, а точнее атакующий, атакуемая компьютерная сеть и трафик (сеансы между хостами, общий трафик — суммарный поток между компонентами симулятора) был создан как агент многоагентной системы. Каждый агент взаимодействует с другими агентами, средой, которая воспринимается и, возможно, модифицируется агентами, и пользователь взаимодействует с агентами через свой интерфейс.

Ознакомление администраторов с атаками на сети возможно и без создания специального программного обеспечения. Например, только с использованием технологии виртуальных машин для имитации сетевых соединений [9, Приложение 1].

Очевидно, что крайне желательно иметь программное приложение, которое демонстрирует не только атаки, но и способы защиты от них. Последнее исследуется, например, в работах [10, 11].

Литература

1. Гуц А.К., Эннс Е.П. Программа, моделирующая компьютерную сеть и сетевые атаки // Математические структуры и моделирование. 2017. № 3(43). С. 139–149.
2. Википедия. Протокол передачи данных. URL: `https://ru.wikipedia.org/wiki/Протокол_передачи_данных` (дата обращения: 27.11.2017).
3. Cleary S. Concurrency in $C^\sharp$ Cookbok. O'Reilly Media, 2014. 207 p.
4. Positive Technologies. Оценка уязвимостей CVSS 3.0. URL: `https://habr.com/company/pt/blog/266485/` (дата обращения: 16.03.2018).
5. Hauzer. Сетевые атаки и кое-что ещё. URL: `http://forum.is.ua/showthread.php?t=16215` (дата обращения: 19.12.2017).
6. Common Vulnerability Scoring System Version 3.0 Calculator. URL: `https://www.first.org/cvss/calculator/3.0` (дата обращения: 16.03.2018).
7. MSDN. Task Class (System.Threading.Tasks). URL: `https://msdn.microsoft.com/en-us/library/system.threading.tasks.task(v=vs.110).aspx` (дата обращения: 26.02.2018).
8. Gorodetski V., Kotenko I. Attacks against Computer Network: Formal Grammarbased Framework and Simulation Tool // Lecture Notes in Computer Science. 2002. V. 2516. P. 219–238.
9. Синадский Н.И., Хорьков Д.А. Защита информации в компьютерных сетях: учеб. пособие. Екатеринбург : УрГУ, 2008. 225 с.
10. Котенко И.В., Шоров А.В. Имитационное моделирование механизмов защиты компьютерных сетей от инфраструктурных атак на основе подхода «нервная система сети» // Труды СПИИРАН. 2012. Вып. 3(22). С. 45–70.
11. Бекенева Я.А., Шипилов Н.Н., Борисенко К.А., Шоров А.В. Моделирование DDOS-атак и механизмов защиты от них // Известия СПбГЭТУ «ЛЭТИ». 2015. № 3. С. 32–39.

SOFTWARE FOR MODELING OF NETWORK AND SIMULATION OF COMPUTER NETWORK ATTACKS

A.V. Bazhenov
Student, e-mail: dr.bazhenoff2017@yandex.ru
A.K. Guts
Dr.Sc. (Phys.-Math.), Professor, e-mail: guts@omsu.ru

Dostoevsky Omsk State University, Omsk, Russia

Abstract. The article presents a software application that simulates different computer networks and attacks on these networks.

Keywords: software application, modeling, network attack, smurfing.

Дата поступления в редакцию: 17.11.2018

Математические структуры и моделирование 2018. № 4(48). С. 113–119

УДК 004.932.2
DOI: 10.25513/2222-8772.2018.4.113-119

СТЕГАНОАНАЛИЗ АЛГОРИТМА КОХА-ЖАО

С.В. Белим
д.ф.-м.н., профессор, e-mail: sbelim@mail.ru
Д.Э. Вильховский
аспирант, e-mail: vilkhovskiy@gmail.com

Омский государственный университет им. Ф.М. Достоевского

Аннотация. Проведён анализ стеганографического алгоритма Коха–Жао. Рассмотрена возможность атаки на обнаружение сообщения. Предложен алгоритм вычисления границ встроенного сообщения, основанный на анализе коэффициентов дискретного косинусного преобразования. Проведён компьютерный эксперимент. Определены параметры встраивания, позволяющие осуществить атаку.

Ключевые слова: стеганография, стегоанализ, алгоритм Коха–Жао, дискретное косинусное преобразование.

Введение

Основной целью стеганографического анализа (стегоанализа) является исследование стойкости схемы стеганографического встраивания к атакам различного типа. Традиционно формулируются три основные задачи стегоанализа. Первая состоит в обнаружении факта наличия встроенного сообщения. При обнаружении встроенного сообщения ставится задача вычисления его размера и расположения в контейнере. Третьей и самой сложной задачей является извлечение встроенного сообщения и его интерпретация без каких-либо данных о параметрах встраивания.

На сегодняшний день основные успехи стегоанализа связаны с решением первой задачи. Причём применяются преимущественно статистические методы исследования контейнера с встроенным сообщением. Все эти методы основаны на предположении о том, что встраивание сообщения вносит изменения в статистические характеристики контейнера, которые могут быть обнаружены на основе исследования различных распределений. Так в статьях [1, 2] делают предположение о случайном характере распределения младших битов синей компоненты и на его основе применяют критерий χ-квадрат для задачи обнаружения стегановставки. Предложенный метод даёт хорошие результаты при равномерном заполнении контейнера. Для решения второй и третьей задачи статистических методов недостаточно и необходимо применять интеллектуальные алгоритмы. Например, в статьях [3–8] для анализа младшего слоя используется метод анализа иерархий, что позволяет с высокой степенью точности определить размеры и положение встроенной информации.

Целью данной работы является стегоанализ алгоритма Коха–Жао [9].

1. Алгоритм встраивания и постановка задачи

В качестве исходного объекта будем рассматривать изображение, в которое, предположительно, осуществлено встраивание сообщения. Достоверная информация о том, было осуществлено встраивание или нет, отсутствует. Однако известно, что встраивание могло быть осуществлено только методом Коха–Жао [9]. Причём встраивание сообщения могло быть осуществлено только непрерывным блоком. Поставим задачу обнаружения факта встраивания и извлечения встроенного сообщения, если оно присутствует.

Метод стеганографического встраивания Коха–Жао [9] использует двумерное дискретное косинусное преобразование и может быть записан в виде следующего алгоритма:

Шаг 1. Разбить исходное изображение на блоки размером 8×8 пикселей.

Шаг 2. Применить дискретное косинусное преобразование к каждому блоку. Получить набор матриц коэффициентов D_i ($i = 1, ..., N$; N — количество блоков) размером 8×8.

Шаг 3. Выбрать блоки для встраивания. Записать в каждый выбранный блок 1 бит встраиваемой информации.

Шаг 4. В каждом блоке выбрать два коэффициента дискретного косинусного преобразования (ДКП) симметричные относительно главной диагонали. Рекомендуется выбирать коэффициенты в среднечастотной области ($D_i[3,4]$ и $D_i[4,3]$, $D_i[3,5]$ и $D_i[5,3]$, $D_i[4,5]$ и $D_i[5,4]$).

Шаг 5. Если встраиваемый бит равен 0, то разность модулей пары коэффициентов дискретного косинусного преобразования должна превышать пороговое значение M_0, для встраивания единичного бита разность должна быть меньше M_0. Поэтому для встраивания нулевого бита увеличивается модуль первого коэффициента и на ту же величину уменьшается модуль второго. Для встраивания единичного бита, наоборот, уменьшается модуль первого коэффициента и на ту же величину увеличивается модуль второго коэффициента.

Шаг 6. Выполняются пункты 4 и 5 для каждого блока.

Шаг 7. Выполнить обратное дискретное косинусное преобразование для каждого блока.

Изменение среднечастотных компонент дискретного косинусного преобразования позволяет минимизировать визуальные эффекты встраивания. Встраивание в низкочастнотные компоненты приводит к заметному изменению фона изображения. Встраивание в высокочастное компоненты приводит к потере мелких деталей изображения.

При извлечении встроенного сообщения считается, что известны пары изменяемых коэффициентов дискретного косинусного преобразования. Первые четыре пункта алгоритма извлечения совпадают с пунктами алгоритма встраивания. Остальные шаги алгоритма имеют вид:

Шаг 5. Найти модуль разности модулей пар коэффициентов дискретного косинусного преобразования, в которое осуществлялось встраивание.

Шаг 6. Если разность превышает M_0, то был встроен нулевой бит, в противном случае — единичный.

Шаг 7. Последовательно определяем встроенные биты для каждого блока.

Для атаки на алгоритм Коха–Жао необходимо определить используемые пары коэффициентов дискретного косинусного преобразования и пороговое значение M_0. При осуществлении стегоанализа будем исходить из трёх предположений:

1. Встраивание происходит в непрерывную область, то есть используются подряд идущие блоки.

2. Во всех блоках для встраивания используются одни и те же пары коэффициентов.

3. Во всех блоках используется одно и то же пороговое значение.

2. Стеганографический анализ

Для решения задач стегоанализа будем исходить из того, что пороговое значение M_0 должно иметь большое значение. В противном случае особенности изображения, используемого в качестве контейнера, могут приводить к ошибкам при извлечении данных.

На первом этапе необходимо выявить пары коэффициентов дискретного косинусного преобразования, используемые для встраивания информации. По аналогии с алгоритмом извлечения сообщения разобъём изображение-контейнер на блоки B_i $(i = 1, ..., N)$ размером 8×8 пикселей. Выполним дискретное косинусное преобразование для каждого блока B_i $(i = 1, ..., N)$ и найдём матрицы коэффициентов D_i $(i = 1, ..., N)$, которые также имеют размер 8×8. Выполним анализ элементов матриц D_i $(i = 1, ..., N)$. Для этого построим три последовательности:

$$C_i(1) = ||D_i[3,4]| - |D_i[4,3]||, \quad i = 1, ..., N,$$
$$C_i(2) = ||D_i[3,5]| - |D_i[5,3]||, \quad i = 1, ..., N,$$
$$C_i(3) = ||D_i[4,5]| - |D_i[5,4]||, \quad i = 1, ..., N.$$

Если встраивание было осуществлено в среднечастотную компоненту, то одна из этих последовательностей должна значительно измениться. Для каждой из последовательностей $C_i(j)$ $(j = 1, 2, 3;\ i = 1, ..., N)$ проанализируем гистограмму зависимости от номера блока i Встроенное сообщение приводит к появлению изменённого блока в виде ступенчатой зависимости. Причём высота ступени зависит от порогового значения M_0. Пример гистограммы для последовательности без встроенного сообщения приведён на рис. 1, а аналогичная последовательность с встроенным сообщением — на рис. 2.

Таким образом, стегоанализ алгоритма Коха–Жао сводится к анализу зависимости последовательностей $C_i(j)$ $(j = 1, 2, 3;\ i = 1, ..., N)$ и выявлению участка ступенчатых изменений. После обнаружения ступенчатых участков необходимо определить их границы. Для этого выполним численное дифференцирование зависимости $C_i(j)$ $(j = 1, 2, 3;\ i = 1, ..., N)$ по i с использованием

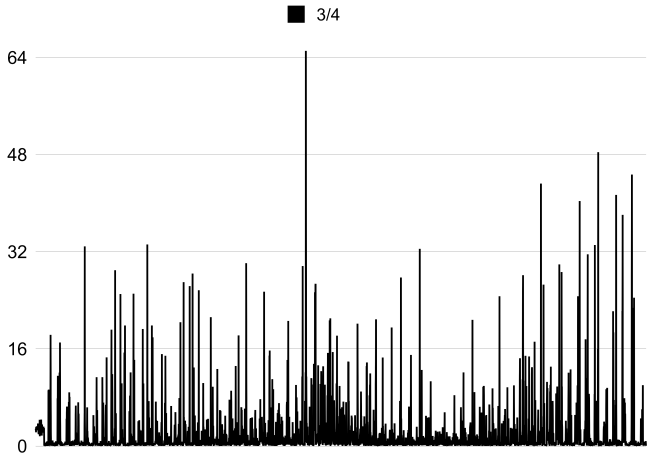


Рис. 1. Гистограмма зависимости $C_i(1)$ без встроенного сообщения

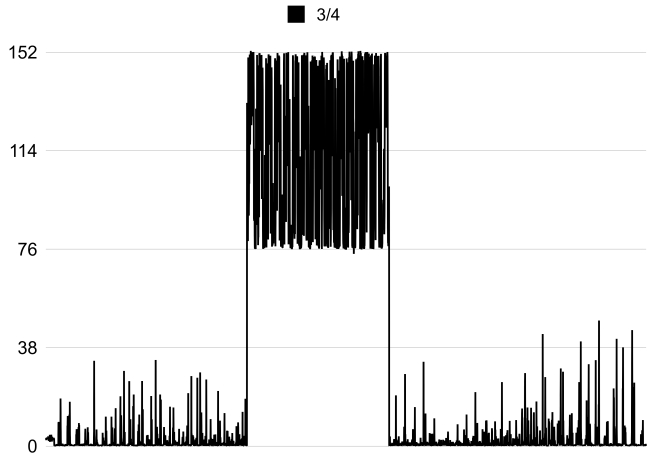


Рис. 2. Гистограмма зависимости $C_i(1)$ с встроенным сообщением

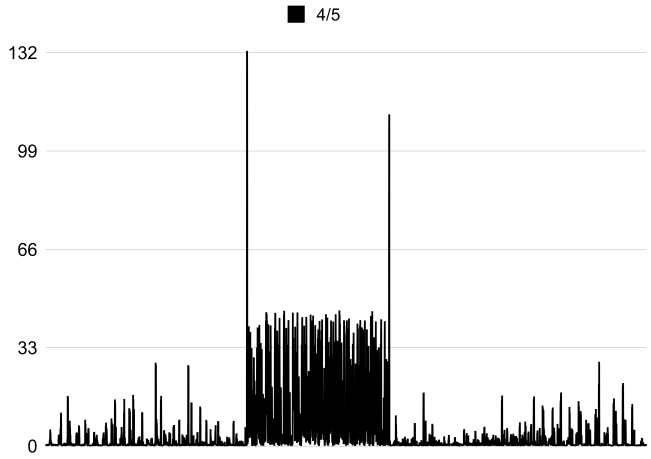


Рис. 3. Гистограмма последовательности $d_i(1)$ с встроенным сообщением

конечных разностей

$$dC_i(j) = C_i(j) - C_{i-1}(j).$$

В точках ступенчатого изменения после дифференцирования будут наблюдаться высокие пики, соответствующие границе встраивания сообщения. Гистограмма $dC_i(j)$ для случая встроенного сообщения представлена на рис. 3.

Поставим задачу автоматического определения границ области встраивания. Для этого для каждой последовательности $d(j)$ определим несколько характеристик: M_j — максимальное значение элементов последовательности $d(j)$, N_j — среднее значение элементов последовательности $d(j)$, O_j — среднеквадратичное отклонение для элементов последовательности $d(j)$. Далее вычислим $R_j = N_j + O_j$. Введём переменную Y_j, принадлежащую интервалу $[R_j,\ M_j]$. Подберём такое значение Y_j, чтобы существовало ровно два элемента последовательности $d(j)$, превышающих его: $C_{i1}(j) > Y_j$ и $C_{i2}(j) > Y_j$. Полученные индексы элементов i_1 и i_2 являются границами встраивания сообщения. Для определения порогового значения M_0 найдём значение минимального элемента последовательности $C_i(j)$ на интервале $[i_1,i_2]$.

Алгоритм выявления встроенного изображения принимает вид:

Шаг 1. Разбить изображение на блоки B_i размером 8×8 пикселей.

Шаг 2. К каждому блоку B_i применить дискретное косинусное преобразование и получить матрицы коэффициентов дискретного косинусного преобразования D_i.

Шаг 3. Вычислить элементы трёх последовательностей величин:

$$C_i(1) = ||D_i[3,4]| - |D_i[4,3]||, \quad i = 1, ..., N,$$
$$C_i(2) = ||D_i[3,5]| - |D_i[5,3]||, \quad i = 1, ..., N,$$
$$C_i(3) = ||D_i[4,5]| - |D_i[5,4]||, \quad i = 1, ..., N.$$

Шаг 4. Выполнить численное дифференцирование каждой из последователь-

ностей $C_i(j)$ по i:

$$dC_i(j) = C_i(j) - C_{i-1}(j) \quad j = 1, 2, 3 \quad i = 1, ..., N.$$

Шаг 5. Вычислить: M_j — максимальное значение элементов массива $d(j)$, N_j — среднее значение элементов массива $d(j)$, O_j — среднеквадратичное отклонение для элементов массива $d(j)$, $R_j = N_j + O_j$.

Шаг 6. Перебрать различные значения величины Y_j в интервале от R_j до M_j с шагом dY. Определить Y_j такое, что существует ровно два значения $C_{i1}(j) > Y_j$ и $C_{i2}(j) > Y_j$. Если такое значение определить невозможно, то уменьшить шаг dY. Определить i_1 и i_2.

Шаг 7. Найти минимальное значение $C_i(j)$ на интервале от i_1 до i_2. Присвоить M_0 найденное значение.

Шаг 8. Извлечь сообщение, используя найденные параметры.

Компьютерный эксперимент показал, что данный алгоритм позволяет безошибочно находить и извлекать встроенное сообщение при значениях $M_0 > 54$.

Заключение

Стегоанализ алгоритма стеганографического встраивания Коха–Жао, проведённый в данной статье, выявил неустойчивость к атаке анализа коэффициентов дискретного косинусного преобразования. Предложенный в статье алгоритм позволяет с высокой точностью определить положение встроенного сообщения и извлечь его. Алгоритм применим при встраивании в непрерывную область.

Литература

1. Provos N., Honeyman P. Detecting steganographic content on the internet // Technical Report CITI 01-1a. University of Michigan, 2001.
2. Westfeld A., Pfitzmann A. Attacks on Steganographic Systems // Lecture Notes in Computer Science. 2000. V. 1768. P. 61–75.
3. Vilkhovskiy D.E., Belim S.V. Detection the Stego-Insertions Like LSB-Substitution in Bitmap Images // Proceedings of the Workshop on Data, Modeling and Security (DMS 2017). CEUR Workshop Proceedings. 2017. V. 1965. URL: `http://ceur-ws.org/Vol-1965/paper11.pdf` (дата обращения: 26.10.2018).
4. Belim S.V., Vilkhovskiy D.E. Usage of analytic hierarchy process for steganographic inserts detection in images // 2016 Dynamics of Systems, Mechanisms and Machines (Dynamics). 2016. URL: `http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7818977&isnumber=7818960` (дата обращения: 26.10.2018).
5. Belim S.V., Vilkhovskiy D.E. Steganalysis Algorithm Based on Heirarchy Analysis Method // Proceedings of the Workshop on Data Analysis and Modelling (DAM 2016). CEUR Workshop Proceedings. 2016. V. 1732. URL: `http://ceur-ws.org/Vol-1732/paper7.pdf.` (дата обращения: 26.10.2018).
6. Belim S.V., Vilkhovskiy D.E. Algorithm for detection of steganographic inserts type LSB-substitution on the basis of an analysis of the zero layer // Journal of Physics: Conf. Series. 2017. V. 944. P. 012012(1–6).

7. Белим С.В., Вильховский Д.Э. Алгоритм выявления стеганографических вставок типа LSB-замещения на основе анализа слоя младших битов // Информатика и системы управления. 2017. № 4(54). С. 3–11.
8. Белим С.В., Вильховский Д.Э. Алгоритм выявления стеганографических вставок типа LSB-замещения на основе метода анализа иерархий // Вестник компьютерных и информационных технологий. 2018. № 4. С. 25–33.
9. Koch E., Zhao J. Towards robust and hidden image copyright labeling // IEEE Workshop on Nonlinear Signal and Image Processing. 1995. P. 452–455.

KOCH-ZHAO ALGORITHM STEGANALYSIS

S.V. Belim
Dr.Sc. (Phys.-Math.), Professor, e-mail: sbelim@mail.ru
D.E. Vilkhovskiy
Postgraduate Student, e-mail: vilkhovskiy@gmail.com

Dostoevsky Omsk State University

Abstract. The analysis of the Koch–Zhao steganographic algorithm was carried out. The possibility of an attack on the detection of messages is considered. An algorithm for calculating the boundaries of the embedded message based on the analysis of the discrete cosine transform coefficients is proposed. A computer experiment is conducted. Embedding parameters that allow the attack are defined.

Keywords: steganography, steganalysis, Koch–Zhao algorithm, discrete cosine transform.

Дата поступления в редакцию: 17.11.2018

Математические структуры и моделирование 2018. № 4(48). С. 120–127

УДК 004.932
DOI: 10.25513/2222-8772.2018.4.120-127

ВЫБОР БЛОКОВ В ВИДЕОПОТОКЕ ДЛЯ ВСТРАИВАНИЯ ЦИФРОВЫХ ВОДЯНЫХ ЗНАКОВ

С.В. Белим
д.ф.-м.н., профессор, e-mail: sbelim@mail.ru
П.Г. Черепанов
аспирант, e-mail: pcherepanov@gehtsoft.com

Омский государственный университет им. Ф.М. Достоевского, Омск, Россия

Аннотация. В статье предложен метод выбора кадров для встраивания цифровых водяных знаков в видеофайлы. Видеопоток представляется как трёхмерный объект. Для встраивания цифровых водяных знаков используется метод Коха–Жао, обобщённый на трёхмерный случай. Основная цель предлагаемого метода состоит в снижении видимых искажений от встраивания данных. Для отбора кадров выполняется их статистический анализ. В качестве основного параметра выбрана дисперсия интенсивности цветов отдельных пикселей. Проведён компьютерный эксперимент. Для оценки уровня заметности вычисляется отношение шума к основному сигналу. Опытным путём определено пороговое значение дисперсии, выше которого встраивание является наименее заметным.

Ключевые слова: цифровые водяные знаки, стеганография видеофайлов, дискретное косинусное преобразование.

Введение

При встраивании цифровых водяных знаков в изображение или видеофайл ставятся две основные задачи — устойчивость и незаметность. Целью данной статьи является повышение незаметности цифровых водяных знаков в видеофайлах. Заметность встроенной информации обусловлена визуальными эффектами от встраивания.

Наибольшее распространение для видеофайлов получил алгоритм встраивания, являющийся логическим продолжением подхода, развитого для изображений. Видеопоток разбивается на отдельные кадры, после чего производится встраивание частей цифрового водяного знака в последовательность изображений [1,2]. Несмотря на высокую скорость работы, такой подход обладает существенными недостатками. Основной из них связан с проблемой синхронизации кадров и требует встраивания дополнительной информации для однозначного извлечения цифрового водяного знака. В качестве таких дополнительных данных могут выступать статистические характеристики кадра [3,4] и специальные метки синхронизации [2], также может применяться избыточное кодирование [5]. В связи с тем, что части цифрового водяного знака локализованы

в достаточно малой области, покадровое встраивание приводит к заметным визуальным эффектам при переходе от одного кадра к другому.

Для снижения видимости эффектов от встраивания цифрового водяного знака нужно использовать стеганографические алгоритмы, обеспечивающие равномерное распределение встраиваемой информации по всему видеопотоку. В этом случае локальное изменение данных приводит к изменениям во всех кадрах и становится визуально менее заметным.

Одним из возможных решений, распределяющим цифровой водяной знак по всем кадрам равномерно, является представление видеопотока как трёхмерного объекта и построение стеганографических алгоритмов на основе трёхмерных преобразований. Трёхмерное дискретное косинусное преобразование впервые было использовано в статье [6]. В дальнейшем оно применялось в основном для оценивания качества кодирования видеопотока на основе исследования распределения коэффициентов преобразования [7–10]. Алгоритмы встраивания стеганографических вставок в видеопоток как трёхмерный объект были разработаны на основе различных технологий: статических и динамических временных компонентов небольшой волны вдоль оси времени [11], на основе трёхмерного преобразования Фурье [12,13], на базе трёхмерного вейвлет-преобразования [14–16], на базе трёхмерного дискретного косинусного преобразования [17–19].

В данной статье предложен алгоритм выбора блоков для встраивания цифрового водяного знака в видеофайл, снижающий визуальные эффекты.

1. Метод встраивания цифровых водяных знаков

Для встраивания цифрового водяного знака будем использовать модель YUV видеопотока. Каждый пиксель определяется тремя цветовыми координатами: яркостью (Y) и двумя цветоразностными компонентами (U,V). Структура видеоизображения задаётся в чёрно-белом представлении распределением яркостной компоненты. Две оставшиеся координаты U и V позволяют восстановить цвет. Для встраивания цифрового водяного знака будем использовать только яркостную компоненту, так как изменение двух других компонент существенно сказывается на визуальном восприятии видеоизображения.

Представим видеопоток как трёхмерный объект. Первые две координаты будут определять положение точки на кадре, а третья — номер кадра. Встраивание информации будем осуществлять в частотную область. Для этого используем трёхмерное дискретное косинусное преобразование. Разобьём видеопоток на одинаковые трёхмерные блоки размером $N \times N \times N$. В каждый блок попадут пиксели из N идущих подряд кадров. Причём из каждого кадра будет задействовано N^2 пикселей. Значение интенсивности в точке (x, y, z) для i-го блока будем обозначать через $f^i_{x,y,z}$, а спектральные коэффициенты дискретного косинусного преобразования в точке (u, v, k) — $f^i_{u,v,k}$. Трёхмерное дискретное косинусное преобразование может быть записано в следующем виде:

$$f^i_{u,v,k} = \frac{\xi(u)\xi(v)\xi(k)}{\sqrt{8/N^3}} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \sum_{z=0}^{N-1} f^i_{x,y,z} \cos\left(\frac{\pi u(2x+1)}{2N}\right) \times$$

$$\times \cos\left(\frac{\pi v(2y+1)}{2N}\right)\cos\left(\frac{\pi k(2z+1)}{2N}\right).$$

Здесь

$$\xi(u)=\begin{cases}\frac{1}{\sqrt{2}},\ if\ u>0\\ 1,\ if\ u=0.\end{cases}$$

Результатом данного преобразования будет набор трёхмерных матриц спектральных коэффициентов размером $N\times N\times N$.

Для встраивания скрытого сообщения будем использовать принцип, предложенный в методе Коха–Жао [20]. При этом в каждую матрицу спектральных коэффициентов встраивается ровно один бит цифрового водяного знака. Для снижения визуальных искажений, обусловленных встраиванием сообщения, встраивание надо производить в среднечастотную область. В низкочастотных компонентах, расположенных в окрестности элемента $(0,0,0)$, содержатся основные данные видеопотока. Высокочастотные компоненты, соседние с элементом (N,N,N), отвечают мелким деталям изображений кадров и наиболее чувствительны к встраиванию.

Для встраивания одного бита в блок выберем два коэффициента из среднечастотной области (u_1,v_1,k_1) и (u_2,v_2,k_2). Кроме этого, необходимо задать пороговое значение P. Цифровой водяной знак m запишем в виде последовательности бит. Пусть значение i-го бита равно m_i. Вычислим вспомогательные функции:

$$\omega_1^i(u_1,v_1,k_1)=|f^i_{u_1,v_1,k_1}|,$$
$$\omega_2^i(u_2,v_2,k_2)=|f^i_{u_2,v_2,k_2}|,$$
$$Z_1^i(u_1,v_1,k_1)=\begin{cases}-1,\ if\ f^i_{u_1,v_1,k_1}<0\\ 1,\ if\ f^i_{u_1,v_1,k_1}\geqslant 0.\end{cases}$$
$$Z_2^i(u_2,v_2,k_2)=\begin{cases}-1,\ if\ f^i_{u_2,v_2,k_2}<0\\ 1,\ if\ f^i_{u_2,v_2,k_2}\geqslant 0.\end{cases}$$

Для того чтобы встроить бит m_i в блок $f^i_{u,v,k}$, изменим коэффициенты дискретного косинусного преобразования:

$$\Omega^i(u_1,v_1,k_1)=\begin{cases}P+\omega_2^i(u_2,v_2,k_2)+1,\ if\ (\omega_1-\omega_2)\leqslant P\ and\ m_i=0,\\ \omega_1^i(u_1,v_1,k_1),\ if\ (\omega_1-\omega_2)>P\ and\ m_i=1.\end{cases}$$
$$\Omega^i(u_2,v_2,k_2)=\begin{cases}P+\omega_1^i(u_1,v_1,k_1)+1,\ if\ (\omega_1-\omega_2)\geqslant P\ and\ m_i=1,\\ \omega_2^i(u_2,v_2,k_2),\ if\ (\omega_1-\omega_2)<P\ and\ m_i=0.\end{cases}$$
$$F^i(u_1,v_1,k_1)=Z_1^i(u_1,v_1,k_1)\cdot\Omega^i(u_1,v_1,k_1),$$
$$F^i(u_2,v_2,k_2)=Z_2^i(u_2,v_2,k_2)\cdot\Omega^i(u_2,v_2,k_2).$$

После чего произведём обратное дискретное косинусное преобразование

$$F^i_{x,y,z}=\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}\sum_{k=0}^{N-1}\frac{\xi(u)\xi(v)\xi(k)}{\sqrt{8/N^3}}F^i_{u,v,k}\cos\left(\frac{\pi u(2x+1)}{2N}\right)\times$$

$$\times \cos\left(\frac{\pi v(2y+1)}{2N}\right) \cos\left(\frac{\pi k(2z+1)}{2N}\right),$$

Полученные блоки объединим в видеопоток в обратном порядке.

Извлечение цифрового водяного знака осуществляется аналогично встраиванию. Видеопоток разбивается на трёхмерные блоки, к которым применяется дискретное косинусное преобразование. Для определения значения i-го бита цифрового водяного знака m_i используется соотношение

$$\omega_1^i(u_1, v_1, k_1) = |F^i_{u_1,v_1,k_1}|,$$
$$\omega_2^i(u_2, v_2, k_2) = |F^i_{u_2,v_2,k_2}|,$$
$$m_i = \begin{cases} 0, \ if \ (\omega_1^i(u_1, v_1, k_1) \geqslant \omega_2^i(u_2, v_2, k_2), \\ 1, \ if \ (\omega_1^i(u_1, v_1, k_1) < \omega_2^i(u_2, v_2, k_2). \end{cases}$$

2. Снижение видимости цифровых водяных знаков

В простейшем случае блоки формируются из последовательных кадров, а разбиение кадров осуществляется слева-направо и сверху-вниз. Однако компьютерный эксперимент показал, что при наличии на кадрах видеопотока больших областей равномерной заливки или заливки, близкой к равномерной, появляются визуальные эффекты, позволяющие легко определять наличие цифрового водяного знака. Избавиться от этих визуальных эффектов можно, выбирая блоки, которые по своей структуре делают встраивание наименее заметным. Данная операция легко выполняется в ручном режиме, но требует поиска характеристик блоков для автоматизации процесса.

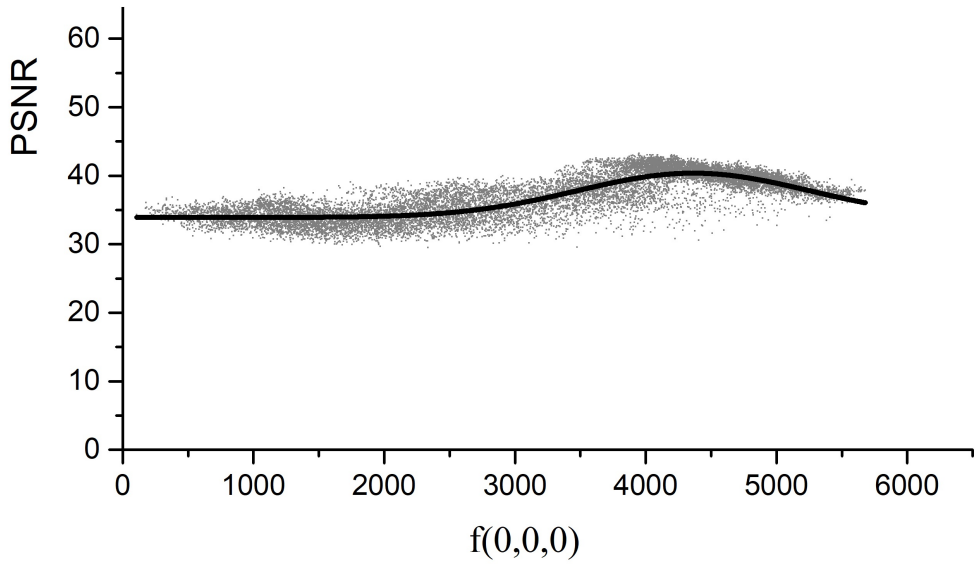


Рис. 1. Зависимость $PSNR$ блока со встроенным сообщением от $f(0, 0, 0)$

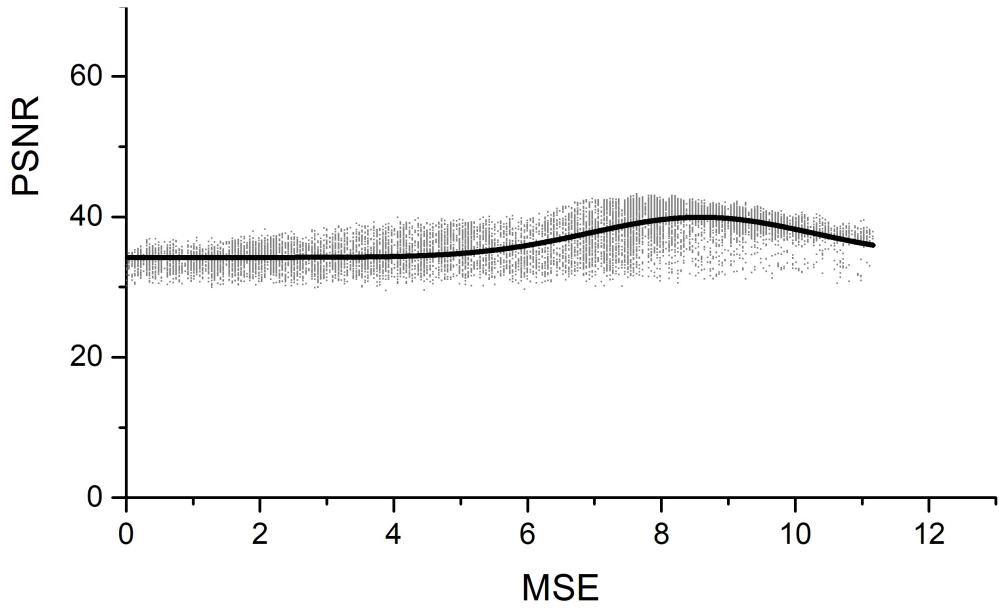


Рис. 2. Зависимость $PSNR$ блока со встроенным сообщением от MSE_0

Для оценки влияния встроенного цифрового водяного знака на видеопоток может быть использована [21] величина отношения сигнала к шуму (PSNR)

$$PSNR = 20 \log_{10} \frac{255}{\sqrt{MSE}},$$

где

$$MSE = \frac{1}{MK} \sum_i \sum_j [P(i,j) - \tilde{P}(i,j)]^2,$$

$P(i,j)$ и $\tilde{P}(i,j)$ — значения яркости пикселя в исходном видеопотоке и в видеопотоке со встроенным водяным знаком соответственно, а M и K — ширина и высота кадров в видеопотоке. Единицей измерения $PSNR$ служит децибел (db).

Введём две характеристики блоков видеопотока. Первым параметром будет нулевой коэффициент дискретного косинусного преобразования $f(0,0,0)$. Данная характеристика показывает яркость фона участка кадров, входящих в блок. Для участков кадров с ярким фоном можно ожидать более заметного эффекта от встраивания. Вторым параметром будет служить среднеквадратичное отклонение яркости пикселей, входящих в блок от среднего значения

$$MSE_0 = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} [P(i,j) - P_0]^2,$$

где $P(i,j)$ — значения яркости пикселей, а P_0 — среднее значение яркости

пикселей, входящих в блок,

$$P_0 = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} P(i,j).$$

Проведём компьютерный эксперимент для определения зависимости $PSNR$ блока с встроенным цифровым водяным знаком от $f(0,0,0)$ и MSE_0. На рис. 1 приведена зависимость $PSNR$ блока от $f(0,0,0)$. На рис. 2 приведена зависимость $PSNR$ блока от MSE_0.

На обоих рисунках точками показаны значения для отдельных блоков, а сплошной линией — аппроксимированное значение. Как видно, на обоих графиках присутствует явно выраженный максимум, который позволяет проводить отбор блоков. Необходимо выбирать блоки, параметры которых лежат близко к максимуму. Таким образом, визуальные эффекты от встраивания цифровых водяных знаков могут быть снижены с помощью отбора блоков для встраивания информации.

Литература

1. Lin E.T., Delp E.J. Temporal synchronization in video watermarking // IEEE Transactions on Signal Processing. 2004. V. 52(10). P.3007–3022.
2. Delannay D., Macq B. Classification of watermarking schemes robust against loss of synchronization // Proceedings of SPIE 5306, Security, Steganography and Watermarking of Multimedia Contents VI. 2004. P. 581–591.
3. Chen C., Ni J., Huang J. Temporal statistic based video watermarking scheme robust against geometric attacks and frame dropping // Digital Watermarking. Springer, 2009. P. 81–95.
4. Sun S.W., Chang P.C. Video watermarking synchronization based on profile statistics // Proceedings of IEEE 37th Annual 2003 International Carnahan Conference on Security Technology. 2003. P. 410–413.
5. Митекин В.А., Федосеев В.А. Метод встраивания информации в видео, стойкий к ошибкам потери синхронизации // Компьютерная оптика. 2014. Т. 38, № 3. С. 564–573.
6. Roese J., Pratt W., Robinson G. Interframe cosine transform image coding // IEEE Transaction on Communication. 1977. V. 25, No. 11. P. 1329–1339.
7. Bauer M., Sayood K. Video coding using 3 dimensional DCT and dynamic code selection // Proceedings of Data Compression Conference. 1995. P. 451.
8. Servais M., de Jager G. Video compression using the three dimensional discrete cosine transform (3D-DCT) // Proceedings of the South African Symposium on Commun. and Signal Process. 1997. P. 27–32.
9. Chan R.K.W., Lee M.C. 3D-DCT quantization as a compression technique for video sequences // Proc. of International Conf. on Virtual Sys. and MultiMedia. 1997. P. 188–196.
10. Bozinovic N., Konrad J. Motion analysis in 3D DCT domain and its application to video coding // Signal Processing: Image Communication. 2005. No. 20. P. 510–528.

11. Swanson M., Zhu B., Tewfik A.T. Multiresolution scene-based video watermarking using perceptual models // IEEE Journal on Sel. Areas in Comm. 1998. V. 16, No. 4. P. 540–550.
12. Deguillaume F., Csurka G., O'Ruanaidh J., Pun T. Robust 3D DFT video watermarking // Proc. SPIE, Security and Watermarking of Multimedia Content II. 2000. V. 3971. P. 346–357.
13. Liu H., Chen N., Huang J., Haung X., Shi Y.Q. A robust DWT-based video watermarking algorithm // IEEE International Symposium on Circuits and Systems. 2002. P. 631–634.
14. Kucukgoz M., Harmanci O., Mihcak M.K., Venkatesan R. Robust Video Watermarking via Optimization Algorithm for Quantization of Pseudo-Random Semi-Global Statistics // Proc. SPIE, Security, Steganography, and Watermarking of Mult. Cont. VII. 2005. P. 5681.
15. Campisi P., Neri A. Video watermarking in the 3D-DWT domain using perceptual masking // IEEE Int. Conference on Image Processing. Genoa, Italy. 2005.
16. Campisi P. Video watermarking in the 3D-DWT domain using quantization-based methods // IEEE International Workshop on Multimedia Signal Processing. 2005.
17. Lim J.H., Kim D.J., Kim H.T., Won C.S. Digital video watermarking using 3D-DCT and Intra-Cubic Correlation // Proc. SPIE, Security and Watermarking Contents III. 2001. V. 4314. P. 54–72.
18. Campisi P., Neri A. 3D-DCT video watermarking using quantization-based methods // 15th European Signal Processing Conference (EUSIPCO 2007). 2007. P. 2544–2548.
19. Cherepanov P.G., Belim S.V. Robust Algorithm of Embedding of Digital Water Marks in a Video Stream // Proceedings of the Workshop on Data, Modeling and Security (DMS 2017). CEUR Workshop Proceedings. 2017. V. 1965. URL: http://ceur-ws.org/Vol-1965/paper12.pdf (дата обращения: 26.10.2018).
20. Koch E., Zhao J. Towards robust and hidden image copyright labeling // IEEE Workshop on Nonlinear Signal and Image Processing. 1995. P. 452–455.
21. Huynh-Thu Q., Ghanbari M. Scope of validity of PSNR in image/video quality assessment // Electronics Letters. 2008. V. 44, No. 13. P. 800–801.

SELECTION OF BLOCKS IN A VIDEO STREAM FOR EMBEDDING DIGITAL WATERMARKS

S.V. Belim
Dr.Sc. (Phys.-Math.), Professor, e-mail: sbelim@mail.ru
P.G. Cherepanov
Postgraduate Student, e-mail: pcherepanov@gehtsoft.com

Dostoevsky Omsk State University

Abstract. The article proposes a method for selecting blocks for embedding digital watermarks in video files. The video stream is represented as a three-dimensional object. For embedding digital watermarks, the Koch-Zhao method is used, which is generalized to the three-dimensional case. The main goal of the proposed method is to reduce visible distortions from embedding data. For the selection of blocks, their statistical analysis is performed. The variance of the color intensity of individual pixels is chosen as the main parameter. A computer experiment is conducted. To assess the level of visibility, the ratio of noise to the main signal is calculated. The threshold value of the variance above which the embedding is the least noticeable is experimentally determined.

Keywords: digital watermarks, video steganography, discrete cosine transform.

Дата поступления в редакцию: 17.11.2018

Математические структуры и моделирование 2018. № 4(48). С. 128–138

УДК 004.056
DOI: 10.25513/2222-8772.2018.4.128-138

О ПРЕДСТАВЛЕНИИ НЕКОТОРЫХ РОЛЕВЫХ МОДЕЛЕЙ РАЗГРАНИЧЕНИЯ ДОСТУПА ОБЪЕКТНО-ОРИЕНТИРОВАННОЙ МОДЕЛЬЮ HRU

С.В. Усов
к.т.н., доцент, e-mail: raintower@mail.ru

Омский государственный университет им. Ф.М. Достоевского, Омск, Россия

Аннотация. Рассматривается возможность представления ролевых политик безопасности с помощью объектно-ориентированных дискреционных моделей разделения доступа. Рассмотрены ролевая модель без иерархии, а также ролевая модель с иерархией с наследованием «сверху». Полномочия ролевой модели представлены в виде наборов пар «объект — право доступа». Построена иерархия классов модели OOHRU, отражающая ролевую политику разграничения доступа. Описаны команды модели OOHRU, соответствующие переназначению ролей в исходной модели.

Ключевые слова: ролевые модели разграничения доступа, OOHRU, иерархия ролей.

Введение

Дискреционные политики безопасности известны ещё с 70-х годов прошлого столетия и традиционно базируются на субъектно-объектной парадигме компьютерной системы. Однако в связи с возрастающей актуальностью объектно-ориентированного подхода к построению компьютерных систем возникает необходимость в пересмотре классических политик безопасности. Так, например, в [1] была предложена объектно-ориентированная модель разграничения доступа, базирующаяся на модели HRU (Харриссона–Руззо–Ульмана) [2], однако обладающая более широкими возможностями, в частности, в рамках охвата компьютерных систем, которые можно описать с помощью этой модели.

Ролевые политики разграничения доступа стали широко известны только в 90-х годах после выхода работ Феррайоло и Куна [3]. Они отличаются от списков контроля доступа (ACL), используемых в системах управления доступом, основанных на дискреционных моделях безопасности, тем, что позволяют назначать на сложные операции с составными данными, а не только на атомарные операции с низкоуровневыми объектами данных.

Концепции иерархии ролей и ограничений позволяют создать или смоделировать контроль доступа на основе решетки доступов. RBAC широко используется для управления пользовательскими привилегиями в пределах единой системы или приложения. Список таких систем включает в себя Microsoft

Active Directory, FreeBSD, Solaris, СУБД Oracle, PostgreSQL 8.1 и множество других. В работах Рави Санду (Ravi Sandhu) [4] было показано, что технология управления доступом на основе ролей обладает достаточной гибкостью для моделирования как дискреционных, так и мандатных политик безопасности. Однако в данных работах моделируются лишь частные дискреционные политики, близкие по структуре к Take-Grant, но заметно отличающиеся от HRU.

В данной работе будет построено ролевое описание некоторого частного случая объектно-ориентированной модели HRU, но в первую очередь мы рассмотрим обратное отображение, позволяющее смоделировать ролевую политику разграничения доступа на основе дискреционной модели Харриссона–Руззо–Ульмана.

В работе [5] была предложена иерархическая модель OOHRU, устройство которой подразумевает, что объект o, находящийся на более низком уровне иерархии, чем объект o', обладает меньшим набором прав (как в отношении доступа к другим объектам, так и в отношении ограничения доступа других объектов по отношению к себе) по сравнению с объектом o'. Такая структура напоминает решётку иерархии ролей, что позволяет предположить структурную близость данных моделей.

1. Объектно-ориентированная модель безопасности с дискреционным разграничением доступа (OOHRU)

Компьютерная система в OOHRU рассматривается в виде множества объектов $\mathbf{O}$, разбитых по множеству классов $\mathbf{K}$ (все объекты одного класса имеют одинаковый набор полей и методов), обладающих открытыми полями $f \in \mathbf{F}$ и скрытыми полями $p \in \mathbf{P}$, а также методами обработки полей $s \in \mathbf{S}$. Здесь $F = \bigcup_{k \in \mathbf{K}} k.\mathbf{F}$ — множество всевозможных открытых полей всех объектов и классов, $k.\mathbf{F}$ — множество открытых полей класса k (каждый объект класса k обладает тем же набором $k.\mathbf{F}$ открытых полей), аналогично определяются $\mathbf{P}$ и $\mathbf{S}$. Причём если поле $k.f$ наследуется классом k у класса k', то соответствующее поле класса k' мы будем для удобства обозначать именно $k'.f$, подчёркивая тем самым их взаимосвязь (таким образом, $f \in k.\mathbf{F}$ и $f \in k'.\mathbf{F}$). Пусть $\mathbf{O}^k \in \mathbf{O}$ — множество объектов класса $k \in \mathbf{K}$. В случае, если требуется уточнить класс объекта, поле f объекта $o^k \in \mathbf{O}^k$ будем обозначать $o^k.f$, поле f класса k — $k.f$. Для скрытых полей класса будем использовать аналогичные обозначения.

Для построения модели дискреционного разделения доступов для каждого объекта и для каждого класса вводится дополнительное скрытое поле M, содержащее локальную матрицу доступов, и методы работы с матрицей доступов. Строки матрицы доступов объекта o соответствуют объектам и классам системы, столбцы — полям и методам объекта o, а в ячейке, находящейся на пересечении строки, соответствующей объекту o', и столбца, соответствующего полю либо методу $x \in \mathbf{X} = \mathbf{F} \cup \mathbf{S}$, находится подмножество множества R прав доступа, определённых в системе, которое обозначается $o.M[o', x]$. Модификация матриц доступа производится посредством выполнения команд системы безопасности, о которых будет сказано ниже.

Модель безопасности OOHRU называется иерархической (или моделью с иерархией), если на множестве объектов $\mathbf{O}$ задан частичный порядок-иерархия, и в любой момент работы системы для любых двух объектов $o, o' \in \mathbf{O}$ таких, что $o' \leqslant o$, для любого поля или метода $x \in \mathbf{X}$, общего для объектов o и o', и для любого поля или метода $x' \in \mathbf{X}$ объекта $o'' \in \mathbf{O}$, то верно следующее: $o''.M[o, x'] \subset o''.M[o', x']$ и $o'.M[o'', x] \subset o.M[o'', x]$. Здесь и далее «$\leqslant$» — отношение частичного порядка, задающее иерархию.

Состояние системы в модели HRU изменяется под действием команд, которые состоят из условной части и последовательности элементарных операторов [2], которая выполняется, только если истинна условная часть. Список элементарных операторов в OOHRU включает [5]:

1. $Create(o^k, k)$ — создаёт объект o^k класса $k \in \mathbf{K}$, если $o^k \in \mathbf{O}$.
2. $Destroy(o^k)$ — уничтожает объект $o^k \in \mathbf{O}$.
3. $Enter(r, o^k, o'^{k'}.f)$ — вносит право доступа r в $o'^{k'}.M[o^k, o'^{k'}.f]$, где o^k – объект класса k, $o'^{k'}$ — объект класса k'.
4. $Delete(r, o^k, o'^{k'}.f)$ — удаляет право доступа r из $o'^{k'}.M[o^k, o'^{k'}.f]$.
5. $Grant(r, o^k, o'^{k'}.s)$ — разрешает вызов объектом o^k метода $o'^{k'}.s$.
6. $Deprive(r, o^k, o'^{k'}.s)$ — запрещает вызов объектом o^k метода $o'^{k'}.s$.

Изменения, производимые операторами, отражаются в матрицах доступа объектов системы. Подробное описание модели OOHRU можно найти в [5].

2. Ролевая политика безопасности

Ролевая политика разграничения доступа вводит в субъектно-объектную модель компьютерной системы новый класс активных сущностей — класс ролей.

Компьютерная система представляется совокупностью следующих множеств: множества пользователей U, множества ролей $\mathbf{R}$, множества полномочий $\mathbf{P}$ и множества сеансов работы пользователей с системой [3]. Множество объектов системы не задаётся в явном виде, а представляется через множество полномочий $\mathbf{P}$: каждое полномочие заключает в себе отношение на декартовом произведении множества прав доступа и множества объектов системы.

Ролевые отношения устанавливаются отображениями, связывающими множество ролей с множеством полномочий и множеством пользователей.

Управление доступом осуществляется на основе изменения множества активных сеансов системы. Каждому сеансу $c \in C$ сопоставляется пользователь $u_c \in U$, подмножество доступных пользователю в рамках данного сеанса ролей $R_c \subset \mathbf{R}$ и подмножество допустимых полномочий $P_c \subset \mathbf{P}$. Считается, что система функционирует безопасно, если пользователь u_c может осуществлять действия только в рамках полномочий из множества P_c во время сеанса $c \in C$.

На множестве ролей в реальных системах обычно возникает иерархия, индуцированная организационно-управленческими схемами организаций, в которых эксплуатируется система. Как правило, подчинённость ролей в иерархии включает наследование прав и полномочий, которое может быть направлено как «снизу», так и «сверху».

При наследовании «сверху» подчинённый субъект наследует права родительских субъектов. Такой подход, тесно связанный с объектно-ориентированной парадигмой, редко упоминается при рассмотрении ролевых моделей разграничения доступа, но на деле бывает очень полезен, когда иерархия ролей строится путём «от абстрактного к конкретному». Так, например, можно рассмотреть фрагмент иерархии «Сотрудник — Сотрудник финансового отдела — Бухгалтер — Главный бухгалтер». Здесь каждая последующая роль цепочки является уточнением предыдущей. Сохраняя полномочия роли-родителя, она получает новые полномочия.

Однако базовые ролевые модели эксплуатируют наследование «снизу». Здесь можно выделить три ключевых подхода к построению иерархии:

1. Строгий таксономический листовой подход. Всё множество полномочий разбивается на непересекающиеся подмножества, каждое из подмножеств приписывается листу дерева иерархии ролей. Роль в нелистовой вершине наделяется множеством полномочий, являющимся объединением множеств полномочий непосредственно подчинённых ролей (соответствие между ролями и полномочиями, таким образом, индуцируется полномочиями листовых вершин и устанавливается при движении по дереву ролей снизу вверх).

2. Нестрогий таксономический листовой подход. Единственное отличие от предыдущего подхода — отсутствие требования на запрет пересечения подмножеств полномочий, сопоставленных листовым вершинам.

3. Иерархически охватный подход. Граф такой иерархии ролей не обязан быть деревом (но, конечно, не может содержать сильных циклов). При таком подходе считается, что если пользователю сопоставлена некоторая роль $r \in \mathbf{R}$, то ему также должны быть сопоставлены и все роли, подчинённые r. Что позволяет исключить из роли r полномочия, уже содержащиеся в подчинённых ей ролях.

3. Связь между субъектно-объектной ролевой моделью без иерархии на множестве ролей и OOHRU

Основной результат данной работы заключается в том, что субъектно-объектная ролевая модель может быть реализована объектно-ориентированной моделью OOHRU.

В первую очередь договоримся представлять полномочия ролевой модели в виде совокупности так называемых «элементарных» полномочий, каждое из которых задаётся парой (x, r) и заключает в себе право доступа $r \in R$ к объекту или группе одинаковых объектов x субъектно-объектной ролевой модели. В качестве r может также выступать право вызова, если x представляет собой активную сущность системы.

Дабы избежать путаницы в обозначениях, переобозначим множества ролевой модели следующим образом:

$Z = \{z_1, z_2, \dots z_n\}$ — множество ролей системы,

$Y = \{y_1, y_2, \dots y_m\}$ — множество полномочий системы,

X — множество всех объектов системы (в рамках субъектно-объектной парадигмы),

$z.y$ — элементарное полномочие y, относящееся к роли z.

$z.Y$ — полный набор полномочий роли z, $z.Y \in Y$.

Базовая ролевая политика разграничения доступа подразумевает статичность подсистемы безопасности в рамках одного сеанса, то есть полномочия неизменны, к каждой роли относится фиксированный набор полномочий, и каждому пользователю назначен неизменный в течение сеанса набор ролей. Таким образом, перераспределения прав доступа в смысле дискреционной политики безопасности не происходит. Построим иерархическую объектно-ориентированную модель HRU $\Sigma' = (O(t), M(t), K, R)$, соответствующую данной субъектно-объектной ролевой модели $\Sigma = (U, Z, Y, C)$.

Во-первых, сформируем множество R прав доступа модели Σ', вычленив права доступа r из элементарных полномочий. Во-вторых, представим каждое полномочие $y \in Y$ матрицей, строки которой подписаны элементами из множества R, столбцы — элементами из множества X, а на пересечении строки r и столбца $a \in X$ стоит 1, если полномочие y подразумевает право доступа r к объекту a, в противном случае стоит 0.

Рассмотрим всевозможные подмножества ролей из Z, всего их 2^n. Каждому такому подмножеству $\alpha \in 2^Z$ сопоставим класс $k_\alpha \in K$. Иерархия на классах вводится естественным образом: класс k_α является непосредственным наследником класса k_β, если и только если существует такая роль z, что $\alpha = \beta \cup z$.

(На самом деле достаточно ввести только классы, соответствующие наборам ролей, которыми могут наделяться пользователи системы, чтобы избежать избыточности классов. В этом случае иерархия строится аналогичным образом: класс k_α является непосредственным наследником класса k_β если и только если существует такое подмножество ролей $\gamma \subset Z$, что $\alpha = \beta \cup \gamma$, но ни для какого подмножества $\gamma' \subset \gamma$ класса $k_{\beta \cup \gamma'}$ не существует.)

В дальнейшем мы будем ссылаться на эту иерархию классов как на естественную иерархию подмножеств ролей.

Теперь если пользователь $u \in U$ при авторизации на сеанс $c \in C$ получает набор ролей α, то в модели Σ' в классе k_α создаётся новый объект $o_u \in O$. Каждый такой объект содержит в качестве private-поля собственную матрицу доступов (заполняется при создании объекта на основе совокупности всех элементарных полномочий, которыми обладает пользователь u во время сеанса c), поле идентификатора и методы, необходимые для работы с объектами системы. Так, например, каждому полномочию может соответствовать свой метод.

Также нам необходимо ввести классы для объектов доступа ролевой модели. Так как в ролевой модели отсутствует формальное описание объектов системы, извлекать объекты будем из полномочий. Пусть X — множество всех объектов системы (в рамках субъектно-объектной парадигмы). Произведём разбиение на этом множестве по следующему правилу: два объекта a и b из X отнесём к разным блокам разбиения, если выполнено хотя бы одно из следующих условий:

1. Объекты a и b обладают принципиально различной природой (как,

например, текстовый документ и исполняемое приложение), то есть требуют различного набора методов взаимодействий.

2. Существует полномочие $y \in Y$, в матрице которого столбцы, соответствующие объектам a и b, не совпадают.

Таким образом, объекты попадают в один блок разбиения, если над ними можно осуществлять одни и те же операции одними и теми же методами, и в матрице любого полномочия столбцы этих объектов совпадают.

Поскольку система конечна, конечным окажется и число блоков разбиения. Каждому такому блоку $X' \subset X$ сопоставим класс $k_{X'}$, а каждому объекту этого блока — объект класса $k_{X'}$. Структура объектов класса будет повторять структуру объектов исходной системы.

Переход от субъектно-объектной модели множества X к объектно-ориентированной HRU-модели произведём согласно алгоритму, изложенному в работе [5]. В частности, пассивной сущности $x \in X$ субъектно-объектной модели будет соответствовать объект o, содержащий поле f, содержащее информацию объекта x, и приватное поле M — матрицу доступов. Активной сущности x' субъектно-объектной модели будет соответствовать объект o', содержащий метод s, выполняющий активные функции субъекта x', и приватное поле M — матрицу доступов.

Заполнение матриц доступов объектов происходит следующим образом: если множество ролей $\alpha \in 2^Z$ содержит роль с элементарным полномочием $y = (x, r) = (o.f, r)$, где x — объект субъектно-объектной модели, объектно-ориентированной интерпретацией которого является объект o, то в матрицу доступов объекта o в ячейку на пересечении строки поля f и столбца класса k_α заносится право доступа r: $o.M[k_\alpha, f] := o.M[k_\alpha, f] \cup r$.

Если множество ролей α содержит роль с элементарным полномочием $y = (x', r) = (o'.s, r)$, где x' — субъект субъектно-объектной модели, объектно-ориентированной интерпретацией которого является объект o', а r — право вызова процедуры, то в матрицу доступов объекта o в ячейку на пересечении строки метода f и столбца класса k_α заносится 1: $o.M[k_\alpha, f] := 1$.

Отсутствие перераспределения доступов в рамках одного сеанса влечёт отсутствие таких элементарных операторов, как $Enter$, $Delete$, $Grant$, $Deprive$ в командах модели Σ'. Старт пользователем u сеанса c, в котором ему назначается множество ролей α, будет выражен в Σ' следующей командой:

$CommandStartSession\_\alpha(o_u : k_\alpha)$

$\quad Create(o_u, k_\alpha)$.

Команда завершения сеанса этим же пользователем:

$CommandEndSession\_\alpha(o_u : k_\alpha)$

$\quad Destroy(o_u)$.

Тот факт, что перед стартом сеанса выполняется проверка возможности пользователю u назначить множество ролей α, может быть также отражён исключительно средствами модели OOHRU. Для этого вне иерархии введём два дополнительных класса: класс сеансов k_C и класс пользователей k_U. Для каждого сеанса c в класс k_C помещается объект o_c, содержащий поле-идентификатор сеанса, метод $user$ инициализации пользователя и матрицу

доступов. Для каждого пользователя u в класс k_U помещается объект $o*_u$, содержащий метод $user$ и матрицу доступов. Аналогичный метод $user$ будут содержать все классы и объекты естественной иерархии подмножеств ролей. Роль этого метода формальна, нам потребуется лишь соответствующая ему ячейка в матрице доступов содержащего его объекта (класса).

Положим, что сеанс c подразумевает назначение пользователю u ролей из множества α. В этом случае при создании объекта o_c происходит модификация матриц доступа объекта $o*_u$ и класса k_α следующей командой:

$CommandCreateSession\_\alpha(o_c : k_C; o*_u : k_U; k_\alpha)$

$Create(o_c, k_C)$,

$Grant(o_c, o *_u .user)$,

$Grant(o_c, k_\alpha.user)$.

При выполнении этой команды матрицы доступов модифицируются следующим образом: $o *_u .M[o_c, user] = 1$, $k_\alpha.M[o_c, user] = 1$ (объект o_c получает право запускать метод $user$ объекта $o*_u$, относящегося к пользователю u, и аналогичный метод класса k_α). Команда может быть выполнена, только если пользователь u зарегистрирован в системе, при регистрации был создан соответствующий ему объект $o*_u$.

Теперь в команду сеанса мы можем включить проверку возможности пользователю u назначить множество ролей α:

$CommandStartSession\_\alpha(o_c : k_C; o*_u : k_U; o_u : k_\alpha)$

If

$o *_u .M[o_c, user] = 1$ and $k_\alpha.M[o_c, user] = 1$

then

$Create(o_u, k_\alpha)$.

Для смены набора назначенных ролей α на набор β пользователю необходимо завершить текущий сеанс и начать новый, в котором ему будет назначен требуемый набор ролей (конечно, если такой сеанс существует). В OOHRU это осуществимо последовательностью команд:

$CommandEndSession\_\alpha$,

$CommandStartSession\_\beta$.

Также средства OOHRU позволяют модифицировать множество привилегий, назначенных роли z, однако это потребует введения в модель дополнительных сущностей.

Заметим, что построенная нами модель OOHRU будет однородной в зоне классов естественной иерархии подмножеств ролей. Сформулируем полученный результат в виде теоремы.

Теорема 1. *Для любой субъектно-объектной базовой ролевой модели, свободной от иерархии, существует реализующая её иерархическая модель OOHRU.*

4. Связь между субъектно-объектной ролевой моделью с иерархией на множестве ролей и OOHRU. Случай наследования «сверху»

Перейдём к рассмотрению случая ролевой модели с иерархией, основанной на наследовании «сверху». Классы объектов, а также вспомогательные класс пользователей и класс сеансов в реализующей её модели OOHRU останутся неизменными, необходимо построить лишь иерархию классов ролей.

Иерархия классов ролей в OOHRU будет повторять иерархию ролей в исходной ролевой модели. Так, каждой роли z будет сопоставлен класс k_z, и если роль z' является непосредственным потомком роли z в исходной модели (при этом набор полномочий $z.Y$ роли z будет содержаться в наборе полномочий $z'.Y$ роли z'), то класс $k_{z'}$ будет непосредственным потомком класса k_z в OOHRU. В дальнейшем мы будем называть такую иерархию в OOHRU индуцированной иерархией ролей.

Как и ранее, если пользователь $u \in U$ при авторизации на сеанс $c \in C$ получает роль z, то в модели Σ' в классе k_z создаётся новый объект o_u. Каждый такой объект содержит в качестве private-поля собственную матрицу доступов (заполняется при создании объекта на основе совокупности всех элементарных полномочий, которыми обладает пользователь u во время сеанса c), поле идентификатора и методы, необходимые для работы с объектами системы. Так, например, каждому полномочию может соответствовать свой метод.

Переход от субъектно-объектной модели множества X к объектно-ориентированной HRU-модели произведём согласно алгоритму, изложенному в работе [5].

Заполнение матриц доступов объектов происходит следующим образом: если роль z обладает элементарным полномочием $y = (x, r) = (o.f, r)$, где x — объект субъектно-объектной модели, объектно-ориентированной интерпретацией которого является объект o, то в матрицу доступов объекта o в ячейку на пересечении строки поля f и столбца класса k_z заносится право доступа r: $o.M[k_z, f] := o.M[k_z, f] \cup r$.

Если роль z обладает элементарным полномочием $y = (x', r) = (o'.p, r)$, где x' — субъект субъектно-объектной модели, объектно-ориентированной интерпретацией которого является объект o', а r — право вызова процедуры, то в матрицу доступов объекта o в ячейку на пересечении строки метода f и столбца класса k_z заносится 1: $o.M[k_z, f] = 1$.

Дополним OOHRU командой старта сеанса c пользователем u с авторизацией его на роль z и командой завершения сеанса.

$CommandStartSession\_z(o_u : k_z)$
$\quad Create(o_u, k_z)$.

Команда завершения сеанса этим же пользователем:

$CommandEndSession\_z(o_u : k_z)$
$\quad Destroy(o_u)$.

Первая команда может быть дополнена проверкой возможности сеанса c:

$CommandStartSession\_z(o_c : k_C; o*_u : k_U; o_u : k_z)$

If
$o_u.M[o_c, user] = 1$ and $k_z.M[o_c, user] = 1$
then
$Create(o_u, k_z)$.

Существенное отличие от случая ролевой модели без иерархии заключается в том, что каждый пользовательский объект в индуцированной иерархии ролей соответствует одной роли из числа назначенных субъекту. В информационных системах, эксплуатирующих подход иерархии «вниз» на множестве ролей, назначение одной роли субъекту кажется достаточным, поскольку роль определяется непосредственно сферой обязанностей пользователя, и при необходимости совмещения нескольких обязанностей достаточно создать роль, являющуюся в иерархии наследницей ролей, упомянутых выше обязанностей. Так, если в системе существуют роли z и z', которые необходимо назначить одному пользователю, совмещающему соответствующие обязанности, то достаточно создать новую роль z'', набор полномочий которой будет являться объединением наборов полномочий ролей z и z', а сама роль z'' — наследником как z, так и z'.

Однако если необходимо назначить пользователю u роли z и z' в рамках уже построенной системы безопасности, достаточно создать при назначении пользователя на эти роли два отвечающих пользователю u объекта как в классе k_z, так и в классе $k_{z'}$:

$CommandStartSession\_\{z, z'\}(o_u : k_z, o'_u : k_{z'})$
$Create(o_u, k_z)$,
$Create(o'_u, k_{z'})$.
$CommandEndSession\_\{z, z'\}(o_u : k_z, o'_u : k_{z'})$
$Destroy(o_u)$,
$Destroy(o'_u)$.

Аналогичные команды можно использовать для произвольного набора ролей α.

Как и в случае ролевой модели, свободной от иерархии, для смены набора назначенных ролей α на набор β пользователю необходимо завершить текущий сеанс и начать новый, в котором ему будет назначен требуемый набор ролей.

Модификация множества привилегий, назначенных роли z, осуществима следующим образом. Допустим, роли $z_1 \dots z_t$ являются непосредственными потомками роли z, которой необходимо добавить новое элементарное полномочие $y = (o.f, r)$, то есть право доступа r к полю f объекта o некоторого класса k_o. Для этого используем команду

$CommandAddPermission\_z\_r\_f(k_z, o : k_o)$
$Enter(r, k_z, o.f)$.

Данная команда не нарушает наследование прав доступа в иерархии, поскольку в OOHRU оператор $Enter$ может быть выполнен, только если соблюдены так называемые условия целостности [1]:

$$(r \in o.M[k_{z_1}, f]) \& \dots \& (r \in o.M[k_{z_s}, f]).$$

При необходимости добавить более широкий набор полномочий, расширяем список элементарных операторов $Enter$ в команде, добавляющих необходимые права доступа. Если элементарное полномочие содержит право вызова метода s объекта o некоторого класса k_o, используем команду несколько иного вида:

$CommandAddPermission\_z\_s(k_z, o : k_o)$

$Grant(k_z, o.s)$.

Удаление элементарного полномочия осуществляется аналогичным способом, только условия целостности будут наложены уже на родительские классы:

$CommandDeletePermission\_z\_r\_f(k_z, o : k_o)$

$Delete(r, k_z, o.f)$.

Модификация строк матриц доступа объекта o, соответствующих объектам класса k_z, не требуется за отсутствием таковых, поскольку изменение набора полномочий, соответствующего роли, не может осуществляться в ролевой модели безопасности во время сеанса, завязанного на эту роль.

Тем самым доказана

Теорема 2. *Для любой субъектно-объектной иерархической ролевой модели с наследованием «сверху» существует реализующая её иерархическая модель OOHRU.*

Литература

1. Усов С.В. Объектно-ориентированный подход в построении политики безопасности. Системы с естественной иерархией. // Математические структуры и моделирование. 2010. N. 21. С. 152-162.
2. Harrison M.A., Ruzzo W.L., Ulman J.D. Protection in Operating Systems // Communications of the ACM. 1975. P. 14–25.
3. Ferraiolo D.F., Kuhn D.R. Role-Based Access Control // 15th National Computer Security Conference. 1992. P. 554–563.
4. Sandhu R., Munawer Q. How to do discretionary access control using roles // 3rd ACM Workshop on Role-Based Access Control. 1998. P. 47–54.
5. Усов С.В. Об отношении между дискреционными моделями объектно-ориентированных и субъектно-объектных компьютерных систем // Проблемы информационной безопасности. Компьютерные системы. 2013. Т. 3. С. 18–26.

ON THE REPRESENTATION OF ROLE-BASED ACCESS CONTROL MODELS BY OBJECT-ORIENTED HRU MODEL

S.V. Usov
Ph.D. (Eng.), Associste Professor, e-mail: raintower@mail.ru

Dostoevsky Omsk State University, Omsk, Russia

Abstract. In this paper the possibility of representing of some types of role-based access control models by object-oriented discretionary access control model is considered. The role-based security model without hierarchy and the role-based security model with hierarchy with inheritance "from above" are considered. The permissions of the role-based access control model are represented as a set of pairs of object and access right. A hierarchy of classes of the object-oriented HRU model based on the role-based access control policy is constructed. Commands of the object-oriented HRU model corresponding to the reassignment of roles in the original role-based model are described.

Keywords: role-based access control model, object-oriented discretionary access control model, role hierarchy.

Дата поступления в редакцию: 15.11.2018

Математические структуры и моделирование 2018. № 4(48). С. 139–146

УДК 004.052.32:004.722
DOI: 10.25513/2222-8772.2018.4.139-146

ПРИНЦИПЫ ПОСТРОЕНИЯ ПРОТОКОЛА ГАРАНТИРОВАННОЙ ДОСТАВКИ СООБЩЕНИЙ

Д.Н. Лавров
к.т.н., доцент, e-mail: lavrov@omsu.ru

Омский государственный университет им. Ф.М. Достоевского, Омск, Россия

Аннотация. Как правило, в компьютерных сетях информационная безопасность рассматривается с точки зрения трёх свойств: конфиденциальность, целостность и доступность. В настоящее время актуально рассматривать и такое свойство информации, как анонимность. Для обеспечения последнего используется такое средство, как TOR. Сеть TOR даёт нам иллюзию анонимности в сети Интернет и усыпляет бдительность конечного пользователя. Но на выходном узле TOR-сети трафик расшифровывается, и его содержимое может стать известно владельцу этого выходного узла. Если владельцем узла или серии узлов являются авторитарное правительство или иностранный агент, то он может не только узнать содержимое передаваемого сообщения, но и попытаться модифицировать его или вовсе прервать канал связи. Конфиденциальность сообщения можно защитить с помощью шифрования, но остаётся лишь надеяться, что оно надёжно. Доступность и целостность остаются под угрозой. Здесь под доступностью понимается гарантия доставки целостного сообщения до места назначения. В статье рассматриваются принципы, которые должны быть положены в основу протокола передачи данных с гарантированной доставкой. Идея протокола основана на существовании в сети нескольких независимых маршрутов доставки и использовании криптографических (k, n)–пороговых схем разделения секрета в n сетевых потоках разных маршрутов. Это позволит не только дополнительно анонимизировать трафик, но и в случае контроля авторитарным правительством выходных узлов (меньших порога k) предоставит дополнительную защиту конфиденциальностит трафика.
Исследование выполнено в рамках научного проекта НИОКТР № 01-06/683.

Ключевые слова: разделение секрета, маршрутизируемая сеть, анонимность, доступность.

Введение

В компьютерных сетях информационная безопасность рассматривается с точки зрения трёх свойств: конфиденциальность, целостность и доступность. В настоящее время рассматривается и такое свойство информации, как аноним-

ность. Для обеспечения последнего используются различные инструменты анонимайзеры. Одно из таких средств — это технология TOR (The Onion Routing). Сеть TOR даёт пользователю иллюзию безопасности в сети Интернет. Так ли это на самом деле?

Кратко опишем суть технологии [1, 2]. Сеть TOR состоит из компьютеров добровольцев. По умолчанию маршрут до адресата формируется через три узла TOR-сети: guard (сторожевой) или entry (входной, в более ранней версии), middle (промежуточный) и exit (выходной). Все три узла могут находиться в разных странах и под разной юрисдикцией. Исходный трафик вместе с заголовками многократно шифруется. Так, первый узел шифрует трафик и направляет его на входной узел, тот в свою очередь шифрует вместе с заголовком, содержащим адрес отправителя, и передаёт его промежуточному узлу. Далее полученный трафик ещё раз шифруется и направляется на выходной узел. Получается вложенное туннелирование, так что промежуточный узел не знает, кому и от кого отправлено сообщение. Выходной узел извлекает исходные данные и отправляет их адресату.

Итак, на выходном узле TOR-сети трафик расшифровывается и его содержимое может стать известно владельцу этого выходного узла. Если владельцем узла или серии узлов является злоумышленник, авторитарное правительство или иностранный агент, то они могут не только узнать содержимое передаваемого сообщения, но и попытаться модифицировать его или вовсе прервать канал связи. Конфиденциальность сообщения можно защитить с помощью шифрования (использование https), но остаётся лишь надеяться, что оно надёжно. Имеется информация, что «выходные узлы TOR могут прослушивать коммуникации и осуществлять атаки посредника (MiTM), даже при использовании HTTPS» [3].

Кто разработчики? Технология TOR создана в «Центре высокопроизводительных вычислительных систем» Исследовательской лаборатории Военно-морских сил США совместно с DARPA по федеральному заказу (1999 год). В дальнейшем исходный код был опубликован под свободной лицензией (2003 год).

Кто контролирует выходные узлы? Теоретически проект заявляет, что это делается добровольцами. Имеется информация о контроле выходных узлов спецслужбами иностранных государств [4].

Анализ всех представленных данных показывает, что доступность и целостность находятся под угрозой в TOR-сетях. Под доступностью понимается гарантия доставки целостного сообщения до места назначения.

Таким образом, необходимо разработать принципы, которые могут быть положены в основу протокола передачи данных с гарантированной доставкой. Такой протокол может работать в любой маршрутизируемой сети с несколькими маршрутами доставки до адресата.

1. Модель злоумышленника

Злоумышленник может быть пассивным и только просматривать трафик и активным — может вмешиваться: модифицировать или даже уничтожать сооб-

щения.

Вне зависимости от типа злоумышленник может контролировать менее k-каналов.

2. Существующие подходы

В работе В.И. Ефимова и Р.Т. Файзуллина [6] предложена простая двухканальная схема маскирования трафика.

Пусть M — исходное сообщение. С помощью демультиплексора поток разделяется на два байтовых потока чётных M_0 и нечётных байт M_1. По первому каналу передаётся $M_0 \oplus M_1$, по второму — только M_1 передаётся как есть. У адресата на мультиплексор поступает $M_0 := (M_0 \oplus M_1) \oplus M_1$ и поток M_1, приходящий по второму каналу. Мультиплексор объединяет эти два потока в поток M.

С точки зрения экономии схема не порождает лишнего трафика, но с точки зрения безопасности протокол подвержен атаке восстановления сообщения по словарю. Для восстановления сообщения необходимо наблюдать все каналы.

Ещё одна схема разделения секрета, которую можно использовать для маскирования трафика в системе из нескольких каналов, — это n-канальная схема, описанная у Б. Шнайера [5, п. 3.6].

Пусть M — исходное сообщение. Отправитель генерирует $n-1$ случайных битовых последовательностей по длине равных длине сообщения K_i, $i = 1, \ldots, n-1$. Далее по первому каналу отправляется $M \oplus \bigoplus\limits_{i=1}^{n-1} K_i$, по второму — K_1, по третьему — K_2, ..., по n-ому — K_{n-1}. На принимающей стороне все фрагменты складываются, и исходное сообщение восстанавливается $M := (M \oplus \bigoplus\limits_{i=1}^{n-1} K_i) \oplus K_1 \oplus \ldots \oplus K_{n-1}$.

Достоинством протокола является высокая безопасность (при условии правильного выбора случайных одноразовых ключей K_i, $i = 1, \ldots, n-1$). Первый недостаток в том, что каждый фрагмент по объёму равен всему сообщению. Второй недостаток в том, что при повреждении хотя бы одного фрагмента становится невозможным восстановить всё сообщение.

В работах [7, 8] предлагается оригинальный алгоритм на основе побитового мултиплексирования с битовыми сдвигами или перестановками, осуществляемыми над фрагментами. Сами фрагменты после сдвигов и перестановок становятся автоключом для соседнего канала.

Избыточность алгоритма — низкая, надёжность с точки зрения безопасности — приемлемая [8]. Недостатком подхода является то, что повреждение фрагмента секрета приведёт к невозможности восстановления всего сообщения.

В основу протокола может быть положен и любой из известных алгоритмов разделения секрета. Например, схема Шамира. К сожалению, избыточность алгоритма будет высокой потому, что размер фрагмента секрета (тени) равен в большинстве случаев размеру самого сообщения.

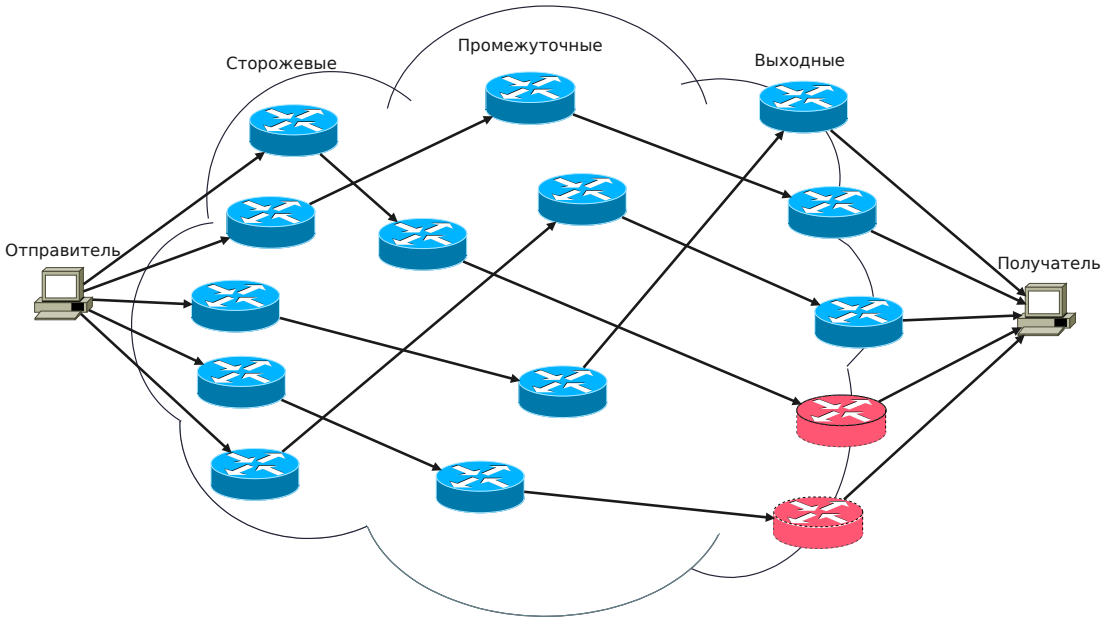


Рис. 1. Схема маршрутизируемой сети с несколькими маршрутами до адресата. Облако обозначает часть маршрутизируемой сети (например, TOR-сеть). Красным выделены выходные маршрутизаторы, находящиеся под управлением злоумышленника

3. Принцип гарантированной доставки

В работе [9] описаны несколько подходов к реализации протокола гарантированной доставки, в частности одним из подходов является использование схемы разделения секрета. Первые три подхода, описанные в предыдущем разделе, также являются видом (n, n)-пороговой схемы.

Идея протокола основана на существовании в сети нескольких независимых маршрутов доставки и использовании криптографических (k, n)-пороговых схем разделения секрета в n сетевых потоках разных маршрутов. Это позволит не только дополнительно анонимизировать трафик, но и в случае контроля злоумышленником выходных узлов (меньших порога k) позволит предоставить дополнительную защиту конфиденциальности трафика.

Рассмотрим схему сети, представленную на рис. 1. Два последних маршрутизатора находятся под контролем злоумышленника. В таблице. 1 представлены сводные результаты возможности восстановления секрета пассивным наблюдателем (конфиденциальность) и возможность повредить сообщение без возможности восстановления (доступность). Из таблицы видно, что в указанной ситуации добиться одновременной доступности и конфиденциальности (без непосредственного шифрования) можно при схеме $(3, 5)$. Гарантию доставки (доступность) можно добиться при пороге $k \leqslant 3$.

Таблица 1. Защищённость многоканального соединения

Схема	**Конфиденциальность**	**Доступность**
(1,5)	–	+
(2,5)	–	+
(3,5)	**+**	**+**
(4,5)	+	–
(5,5)	+	–

4. Восстановление сообщения

Возможны два подхода. *Первый* — комбинаторный. Необходимым условием корректности восстановления является совпадение восстановленного сообщения на нескольких пороговых комбинациях.

Рассмотрим его на примере. Пусть имеется сообщение M, разделённое на пять фрагментов M_1, M_2, M_3, M_4, M_5. Для определённости пусть повреждена пятая тень. Используется пороговая схема $(3, 5)$. Из 5 теней можно скомбинировать $C_5^3 = 10$ комбинаций:

$M_1,\ M_2,\ M_3$; <– $M_2,\ M_3,\ M_4$; <–
$M_1,\ M_2,\ M_4$; <– $M_2,\ M_3,\ \mathbf{M}_5$;
$M_1,\ M_2,\ \mathbf{M}_5$; $M_3,\ M_4,\ \mathbf{M}_5$;
$M_1,\ M_4,\ M_3$; <– $M_1,\ M_4,\ \mathbf{M}_5$;
$M_1,\ \mathbf{M}_5,\ M_3$; $M_1,\ \mathbf{M}_5,\ M_4$.

Из четырёх комбинаций, помеченных "<–", однозначно восстанавливается сообщение. И по ним же мы можем проверить необходимое условие совпадения результата восстановления.

При повреждении двух фрагментов, пусть это будут для определённости M_4 и M_5, только одна комбинация M_1, M_2, M_3 позволяет восстановить сообщение, но что именно по этой комбинации возможно восстановление определить нельзя.

Недостатки подхода: 1) имеется лишь необходимое условие корректности восстановления (достаточное условие неизвестно); 2) при повреждённых фрагментах на единицу меньших порога нет возможности установить, какое из восстановленных сообщений верно; 3) при росте каналов растёт трудоёмкость, как C_n^k.

Второй подход заключается в использовании алгоритма HMAC [10]. Для этого на этапе согласования параметров необходимо сгенерировать общий ключ K, например, с помощью алгоритма Диффи–Хеллмана [11], который в дальнейшем и будет использоваться для алгоритма HMAC.

Каждый фрагмент «подписывается» алгоритмом HMAC на ключе K и передаётся вместе с фрагментом секрета.

5. Описание протокола гарантированной доставки

Первый этап протокола гарантированной доставки — это согласование параметров передачи. На этом этапе осуществляется обмен сообщениями одновременно по всем каналам («лавинная» рассылка). На данном этапе необходимо согласовать следующие параметры:

- тип схемы разделения секрета и её параметры;
- число каналов n;
- порог схемы k;
- модуль схемы разделения секрета p;
- алгоритм HMAC и его параметры;
- ключ K для алгоритма HMAC;
- алгоритм генерации общего ключа и его параметры.

На втором этапе происходит передача данных. Сообщение M разбивается на блоки $B_i < p$, $i = 1, \ldots, N$, p — согласованный ранее параметр, простое число, модуль конечного поля. Каждое B_i разбивается на n фрагментов секрета (теней):

$$M_{1,i}, M_{2,i}, \ldots, M_{n,i}.$$

К каждой тени применяется HMAC:

$$h_{1,i} = HMAC(M_{i,1}|i), \ldots, h_{n,i} = HMAC(M_{n,i}|i).$$

По каналам отправляются пары:

$$(M_{1,i}, h_{1,i}, i);\ (M_{2,i}, h_{2,i}, i), \ldots, (M_{n,i}, h_{n,i}, i).$$

По j-ому каналу передаётся тройка $(M_{j,i}, h_{j,i}, i)$. Третий параметр, номер блока, необходим для отслеживания правильного порядка в сетях с коммутацией пакетов при использовании транспорта без установления соединения.

На принимающей стороне проверяется целостность теней, из неповреждённых фрагментов секрета восстанавливаются блоки B_i исходного сообщения M и затем само сообщение.

Заключение

В работе рассмотрены основные принципы построения протокола гарантированной доставки. В настоящее время идёт разработка расширяемых программных модулей, осуществляющих реализацию подходов, описанных в данной статье.

Благодарности

Выражаю огромную признательность Гуссу Святославу Владимировичу, Бречке Денису Михайловичу, Черкашину Антону Васильевичу за обсуждение принципов архитектуры протокола и конструктивную критику. Часть результатов данной статьи озвучена в докладе на конференции «Современное программирование» (2018 г. [12]).

Исследование выполнено в рамках научного проекта НИОКТР №01-06/683.

Литература

1. Dingledine R., Mathewson N., Syverson P. Tor: The Second-Generation Onion Router. [Электронный ресурс]. 2004. URL: `https://svn.torproject.org/svn/projects/design-paper/tor-design.html` (дата обращения: 15.11.2018).
2. Dingledine R., Mathewson N. Tor Protocol Specification. [Электронный ресурс]. URL: `https://gitweb.torproject.org/torspec.git/tree/tor-spec.txt` (дата обращения: 15.11.2018).
3. Анонимность в TOR: что нельзя делать / псевдоним автора : m1rko // Хабр [Электронный ресурс]. 2017. URL: `https://habr.com/post/329756/` (дата обращения: 15.11.2018).
4. Ализар А. ФБР контролирует выходные узлы TOR? URL: `https://xakep.ru/2014/11/11/fbi-tor/` (дата обращения: 15.11.2018).
5. Шнайер Б. Прикладная криптография: протоколы, алгоритмы, исходные тексты на языке Си. Переводчик: Дубнова Н. 2-е издание. М. : Диалектика, 2003. 610 с.
6. Ефимов В.И., Файзуллин Р.Т. Система мультиплексирования разнесённого TCP/IP трафика // Математические структуры и моделирование. 2002. Вып. 10. С. 170-172
7. Д.Н. Лавров. Схема разделения секрета для потоков данных маршрутизируемой сети // Математические структуры и моделирование. 2002. Вып. 10. С. 192–197.
8. Дулькейт В.И., Лавров Д.Н., Михайлов П.И., Свенч А.А. Анализ надёжности алгоритма разделения секрета в сетевых потоках // Математические структуры и моделирование. 2003. Вып. 12. С. 146–154.
9. Гусс С.В., Лавров Д.Н. Подходы к реализации сетевого протокола обеспечения гарантированной доставки при мультимаршрутной передаче данных // Математические структуры и моделирование. 2018. № 2(46). С. 95–101.
10. Krawczyk H., Bellare M., Canetti R. HMAC: Keyed-hashing for message authentication. // IETF. Febrary, 1997. URL: `https://tools.ietf.org/html/rfc2104` (дата обращения: 15.11.2018).
11. Diffie W., Hellman M. E. New Directions in Cryptography // IEEE Trans. Inf. Theory / F. Kschischang — IEEE. 1976. V. 22, No. 6. P. 644–654.
12. Лавров Д.Н., Черкашин А.В. Гарантированная доставка на основе разделения секрета в сети с несколькими маршрутами // I Международная научно-практическая конференция «Современное программирование» // Нижневартовск : Изд-во Нижневарт. гос. ун-та, 2018.

PRINCIPLES OF BUILDING A PROTOCOL FOR GUARANTEED MESSAGE DELIVERY

D.N. Lavrov
Ph.D.(Eng.), Associate Professor, e-mail: lavrov@omsu.ru

Dostoevsky Omsk State University, Omsk, Russia

Abstract. As a rule, in computer networks information security is considered from the point of view of three properties: confidentiality, integrity and availability. Currently, it is important to consider such property of information as anonymity. To ensure the latter, a tool such as TOR is used. The TOR network gives us the illusion of anonymity on the Internet and lulls the vigilance of the end user. But at the output node of the TOR network, the traffic is decrypted and its contents may become known to the owner of this output node. If the owner of a node or a series of nodes is an authoritarian government, or a foreign agent, then he can not only know the contents of the message being transmitted, but also try to modify it or completely interrupt the communication channel. Confidentiality of the message can be protected by encryption, but one can only hope that it is secure. Accessibility and integrity remain at risk. Here, availability means a guarantee of delivering a complete message to a destination. The article discusses the principles that should be the basis of the data transfer protocol with guaranteed delivery. The idea of the protocol is based on the existence in the network of several independent delivery routes and the use of cryptographic (k, n)—threshold secret separation schemes in the n network flows of different routes. This will allow not only to further anonymize traffic, but also in the case of control by the authoritarian government of the output nodes (lower than k threshold) will provide additional protection for the confidentiality of traffic.
This research was done as part of a research project NIOKTR number 01-06/683.

Keywords: secret sharing, routed network, anonymity, availability.

Дата поступления в редакцию: 21.11.2018

Математические структуры и моделирование 2018. № 4(48). С. 147–153

УДК 004.056
DOI: 10.25513/2222-8772.2018.4.147-153

ИМПОРТ И ЭКСПОРТ РОЛЕВОЙ ПОЛИТИКИ БЕЗОПАСНОСТИ В СУБД ORACLE

Ю.С. Ракицкий
к.т.н., доцент, e-mail: yrakitsky@gmail.com

Омский государственный университет им. Ф.М. Достоевского, Омск, Россия

Аннотация. В статье рассматривается ролевая политика безопасности, реализованная в СУБД Oracle. Модель ролей в СУБД может быть достаточно сложной, содержать большое количество ролей и полномочий. Эта информация хранится в большом количестве связанных между собой таблиц. Реализован механизм извлечения данных, сопоставленных ролевой политике безопасности в СУБД Oracle, и представление этой информации в формате GraphML. Также реализована возможность импорта ролевой политики безопасности, представленной в формате GraphML, в СУБД Oracle.

Ключевые слова: ролевая модель, СУБД, язык представления графов.

Введение

Современные базы данных используют ролевую модель для выдачи полномочий пользователям, выполняя таким образом ролевое разграничение доступа. При использовании ролевой политики безопасности задаётся множество разрешённых системных операций путём введения дополнительных объектов – ролей, наделённых набором разрешённых доступов.

Часто возникает необходимость проанализировать модель, построенную средствами СУБД с теоретической точки зрения, так как для больших систем управление огромным количеством ролей, пользователей и разрешений является сложной задачей, которую трудно выполнять малой группой администраторов безопасности.

Сложность администрирования безопасности СУБД связана с тем, что в СУБД достаточно трудно получить общую картину о политике безопасности, так как информация находится в большом количестве взаимосвязанных друг с другом таблиц [1]. Для решения этой проблемы отношения между субъектами удобно представить в каком-либо формате, пригодном для последующего анализа.

Оценка реальной модели с теоретической точки зрения позволяет выявить ошибки администрирования, проанализировать отношения субъектов и объектов, а также проверить адекватность построенной политики безопасности [2].

Помимо непосредственного анализа модели возможно проведение ряда работ по улучшению текущей ролевой модели [3], уменьшению количества ролей,

выявлению и устранению уязвимостей. Также после анализа и оптимизации политики безопасности нужно внедрить новые правила, поэтому удобно автоматизировать применение правил из-за их большого количества и возможных ошибок администратора базы данных при непосредственной настройке.

В связи с этим появляется необходимость в экспорте ролевой политики безопасности из базы данных в каком-либо формате для последующего анализа. Цели, которые преследуются в ходе анализа, могут быть очень разнообразны. Например, в работе [4] описан процесс автоматизации сортировки полномочий по значению риска утечки. А в работе [5] описаны способы выполнения эквивалентных преобразований иерархии ролей для оптимизации RP-орграфа или приведения его к древовидной структуре. Именно для осуществления подобного рода задач и необходимо данное программное решение. Так как в базе данных насчитываются сотни ролей и полномочий, наличие инструмента, позволяющего автоматически извлечь нужную информацию, существенно упростит дальнейшие действия для администратора.

1. Основные источники информации и формат представления данных

Таблица DBA_ROLES в СУБД Oracle содержит информацию обо всех ролях, которые имеются в базе данных, включая системные предопределённые роли.

В рамках тестирования производилась выборка одного столбца из данной таблицы – $ROLE$.

Таблица $ROLE\_TAB\_PRIVS$ включает в себя информацию обо всех привилегиях, выданных ролям, имеющимся в системе. Выборка производится для каждой роли с условием $WHEREROLE = ROLE\_NAME$. Необходимые для работы столбцы таблицы: $TABLE\_NAME, PRIVILEGE, OWNER$.

В рамках исследований были выделены 16 объектных привилегий в СУБД Oracle, для хранения информации о наличии или отсутствии соответствующей привилегии достаточно одного бита, поэтому можно представлять роль как битовую строку длины 16. Список привилегий можно при необходимости расширить путём внесения нового соответствия. Перечень объектных привилегий и соответствующие позиции битов в битовой строке представлены в таблице 1.

В качестве примера приведём роль $TEST\_ROLE$. Указанная роль имеет права на удаление записей, добавление записей, обновление записей и выборку из объекта $TEST\_TABLE$. Поэтому битовая строка прав $TEST\_ROLE$ на $TEST\_TABLE$ будет выглядеть следующим образом: 0100001000010100.

2. Алгоритмы импорта и экспорта данных

Выделим основные этапы, выполнение которых необходимо для осуществления экспорта данных из СУБД Oracle:

1. Поиск всех ролей в базе данных.

Таблица 1. Заданная нумерация битов для объектных привилегий

Номер бита	Объектная привилегия
0	ALTER
1	DELETE
2	EXECUTE
3	DEBUG
4	FLASHBACK
5	INDEX
6	INSERT
7	ON COMMIT REFRESH
8	QUERY REWRITE
9	READ
10	REFERENCES
11	SELECT
12	UNDER
13	UPDATE
14	WRITE
15	DEQUEUE

2. Для каждой роли запуск процедуры поиска выданных привилегий на объекты системы.
3. Сохранение роли и списка её привилегий на объекты.
4. Заполнение графа с помощью Graph API от TinkerPop.
5. Запись графа в выбранном формате.

Аналогичным образом, выделим основные этапы, выполнение которых необходимо для осуществления импорта данных из СУБД Oracle:

1. Считывание графа из файла заданного формата.
2. Запуск процедуры для считывания данных из графа с помощью Graph API.
3. Для каждой роли запуск процедуры для выдачи и отзыва привилегий на объекты.

3. Компьютерный эксперимент

Программный продукт представляет собой консольное приложение для взаимодействия с оператором, реализованное на языке программирования Java.

Основное меню программы содержит следующие пункты:

1. Установить соединение.
2. Считать данные из БД.
3. Записать данные в БД.
4. Выйти.

Пункт меню «Установить соединение» предлагает пользователю ввести ряд параметров для подключения к базе данных:

- Имя хоста / IP-адрес сервера.
- Порт.
- SID.
- Пароль пользователя SYS.

Осуществление действий, производимых программой, возможно только с привилегиями пользователя SYS. Подключение к базе данных производится с административной привилегией SYSDBA: «SYS AS SYSDBA». В случае успешного подключения появится надпись «Соединение установлено», иначе выведется предупреждение «Введены некорректные данные или хост недоступен».

Пункт «Считать данные из БД» предлагает выбрать формат файла (GraphML или GraphSon), затем ввести путь к файлу для сохранения ролевой политики.

Пункт «Записать данные в БД» предлагает выбрать формат файла (GraphML или GraphSon), затем ввести путь к файлу для считывания из него ролевой политики.

Для выполнения действий по считыванию и записи данных в базу данных необходимо предварительно установить соединение с базой данных.

Пункт «Выйти» предназначен для завершения работы программы.

После выполнения каждого из пунктов, кроме выхода, вызывается функция по очистке консольного окна, и меню печатается снова.

4. Тестирование

Тестирование осуществлялось как с точки зрения производительности, так и корректности работы. Аппаратное и программное окружение, на котором осуществлялось тестирование:

1. Процессор — Intel Core i5-3230M 2.60 GHz.
2. ОЗУ – 8.00 Гб.
3. Windows 10 x64.
4. ORACLE 11g.
5. Java version 1.8.0_51.

Тестирование производительности проводилось с помощью функции System.nanoTime() путём измерения начального времени (перед запуском) и конечного времени (после запуска) и нахождения разницы между этими значениями.

Из таблицы 2 можно заметить, что среднее значение считывания данных из базы данных для формата GraphML составляет около 1100 миллисекунд, в то время как среднее значение для формата GraphSon составляет примерно 1300 миллисекунд. Данную разницу можно объяснить накладными издержками при сохранении файлов в разных форматах. В целом можно отметить, что данная операция выполняется довольно быстро.

Из таблицы 3 можно увидеть, что среднее значение записи данных в базу данных для формата GraphML составляет около 169 секунд, а среднее значение для формата GraphSon составляет примерно 170 секунд. Разница между

Таблица 2. Производительность чтения из базы данных

Номер теста	GraphML, мс	GraphSon, мс
1	999	1293
2	1059	1261
3	1068	1274
4	1064	1278
5	1052	1262
6	1049	1270

Таблица 3. Производительность записи в базу данных

Номер теста	GraphML, сек.	GraphSon, сек.
1	167	169
2	170	171
3	172	175
4	171	172
5	166	166
6	169	170

форматами невелика, её можно объяснить разными накладными расходами на обработку.

Стоит отметить, что запись происходит значительно дольше считывания из-за следующих факторов:

1. Необходимо выполнять формирование запросов на выдачу прав.
2. Необходимо явно отзывать права, для которых в соответствие поставлено отсутствие данного права. Из-за чего в разы увеличивается количество операций.
3. Чтобы выдать или отозвать права, создаётся курсор. Из-за их большого количества также происходят временные задержки.

Результаты зависят от того, насколько нагружена система, так как запись данных в базу данных является очень трудоёмкой операцией, и наличие других ресурсоёмких процессов может серьёзно замедлить выполнение программы.

В целом данные результаты довольно условны, так как всё зависит не только от производственных мощностей, но и от количества объектов в исследуемой базе данных и связей между ними. Но вместе с тем можно сделать вывод, что программа выполняет свои задачи в достаточно приемлемое время, которое не требует от пользователя предварительного планирования и длительного ожидания.

Тестирование корректности чтения данных производилось с помощью создания объектов в базе данных и установления различных связей между ними, затем осуществлялось считывание ролевой политики безопасности в файл формата GraphML. Тестирование корректности записи данных происходило путём изменения файла формата GraphML, внесения изменений с помощью этого

файла и проверки записей в базе данных на соответствие внесённым.

Тестовая конфигурация:

1. Роли R1, R2, R3.
2. Таблицы O1, O2, O3.
3. R1 имеет права ALTER, DELETE, INSERT на таблицу O1.
4. R1 имеет права SELECT и UPDATE на таблицу O2.
5. R2 имеет права DELETE и INSERT на таблицу O3.
6. R3 имеет право SELECT на таблицу O2.
7. R3 имеет права SELECT, INSERT, DELETE на таблицу O3.

Данная конфигурация была настроена в тестовой базе данных. В результате операции считывания получен файл, содержимое файла полностью соответствует настройкам, осуществлённым в базе данных.

В полученном файле было произведено следующее изменение данных: ребро с идентификатором 10 поменяло значение на 0000001000010000, ребро с идентификатором 4 — на 0100000000000100, ребро с идентификатором 5 — на 1100001000000100, ребро с идентификатором 7 -– на 0000001000000000, ребро с идентификатором 9 — на 0100000000010000. Проверка осуществлялась для каждой из ролей. Для роли R1 использовалась команда «SELECT TABLE_NAME, PRIVILEGE FROM ROLE_TAB_PRIVS WHERE ROLE = 'R1'». Результаты обработки отображены в таблице 4.

Таблица 4. Привилегии роли R1

TABLE NAME	PRIVILEGE
O2	UPDATE
O1	UPDATE
O1	INSERT
O2	DELETE
O1	DELETE
O1	ALTER

Для роли R2 получена одна привилегия — INSERT на таблицу O3.

Привилегии, полученные для роли R3, отображены в таблице 5.

Таблица 5. Привилегии роли R3

TABLE NAME	PRIVILEGE
O2	DELETE
O3	INSERT
O3	SELECT
O2	SELECT

Полученные в ходе тестирования результаты являются корректными, что позволяет сделать вывод о том, что запись данных в базу данных производится правильно и соответствует ожиданиям пользователя.

Заключение

Предложенный подход к экспорту и импорту ролевой политики безопасности может применяться для широкого класса задач, являясь при этом вспомогательным инструментом для их осуществления. Использование предложенного инструмента возможно для осуществления анализа и оптимизации ролевой политики безопасности, а также для поиска потенциальных уязвимостей.

В результате тестирования было выявлено, что приложение работает достаточно быстро. Также была проведена проверка корректности работы программы, которая дала положительные результаты.

Литература

1. Database Documentation - Oracle Database // Oracle Help Center. URL: `https://docs.oracle.com/en/database` (дата обращения: 25.01.2017).
2. Гайдамакин Н.А. Теоретические основы компьютерной безопасности: учебное пособие. Екатеринбург : изд-во Урал. ун-та, 2008. 212 с.
3. Девянин П.Н. Модели безопасности компьютерных систем. М. : Издательский центр «Академия», 2005. 144 с.
4. Белим С.В., Богаченко Н.Ф. Применение метода анализа иерархий для оценки рисков утечки полномочий в системах с ролевым разграничением доступа // Информационно-управляющие системы. 2013. № 4. С. 67–72.
5. Белим С.В., Белим С.Ю., Богаченко Н.Ф. Теоретико-графовый анализ ролевой политики безопасности // Математические структуры и моделирование. 2009. № 19. С. 85–96.

IMPORT AND EXPORT OF ROLE-BASED SECURITY POLICIES IN ORACLE DBMS

Y.S. Rakitskiy
Ph.D. (Eng.), Associate Professor, e-mail: yrakitsky@gmail.com

Dostoevsky Omsk State University, Omsk, Russia

Abstract. The article discusses the role-based security policy implemented in the Oracle DBMS. The role model in a DBMS can be quite complex, contain a large number of roles and authorities. This information is stored in a large number of related tables. A mechanism has been implemented to extract data mapped to role-based security policies in an Oracle DBMS and to present this information in GraphML format. It also implemented the ability to import role-based security policy, presented in GraphML format, in Oracle DBMS.

Keywords: role-based model, DBMS, graph representation language.

Дата поступления в редакцию: 20.11.2018

Математические структуры и моделирование 2018. № 4(48). С. 154–161

УДК 004.93
DOI: 10.25513/2222-8772.2018.4.154-161

СХЕМА ХАОТИЧЕСКОЙ МАСКИРОВКИ СООБЩЕНИЙ НА ОСНОВЕ ОРТОГОНАЛЬНЫХ ФУНКЦИЙ

С.В. Белим
д.ф.-м.н., профессор, e-mail: sbelim@mail.ru
Ю.С. Ракицкий
к.т.н., доцент, e-mail: yrakitsky@gmail.com

Омский государственный университет им. Ф.М. Достоевского, Омск, Россия

Аннотация. В статье предложен метод хаотической маскировки дискретного сигнала. Предложенный подход не требует синхронизации хаотических генераторов и устойчив к шумам в канале связи. Эти свойства снимают основную проблему хаотической маскировки, связанную с рассинхронизацией управляющих параметров. Основная идея состоит в использовании кодирования сообщения с помощью ортогональных функций. Далее применяется простое суммирование скрытого сообщения с хаотическим сигналом. Извлечение сообщения может быть выполнено на основе свойства ортогональности без вычитания хаотической составляющей.

Ключевые слова: ортогональные функции, сигнал, маскировка.

Введение

Для скрытой передачи сообщений могут использоваться различные подходы на основе стеганографических алгоритмов, электронной цифровой подписи [1, 2], динамического хаоса [3] и т.д. Эти методы отличаются друг от друга прежде всего контейнером, используемым для встраивания сообщения. Контейнер служит для маскировки факта передачи сообщения. В данной статье предложен алгоритм, основанный на маскировке сообщения с помощью динамического хаоса, получивший название хаотической маскировки.

Использование динамического хаоса для сокрытия передаваемого сообщения предполагает наличие двух связанных идентичных хаотичных генераторов. Разработаны несколько способов использования динамического хаоса в таких задачах: хаотическая маскировка [4], переключение хаотических режимов [5] нелинейное подмешивание передаваемого сообщения к хаотическому сигналу [6], модулирование управляющих параметров хаотического генератора [7]. На основе этих методов разработан ряд алгоритмов передачи данных.

Одним из первых методов использования динамического хаоса для сокрытия сообщения является хаотическая маскировка [4]. Схема использования хаотической маскировки для скрытой передачи сообщения приведена на рис. 1. Абонент, передающий сообщение $m(t)$, добавляет его в сумматоре к хаотичному сигналу $x(t)$. Далее сумма двух сигналов $m'(t) = m(t) + x(t)$ передаётся

по каналу связи. Принимающий абонент синхронизирует свой хаотический генератор $u(t)$ с помощью принимаемого сигнала. В результате синхронизации хаотические генераторы у передающего и принимающего абонента становятся идентичными $u(t) = x(t)$. Переданное сообщение восстанавливается с помощью вычитания из полученного сигнала $m'(t)$ синхронизированного хаотического сигнала $u(t)$.

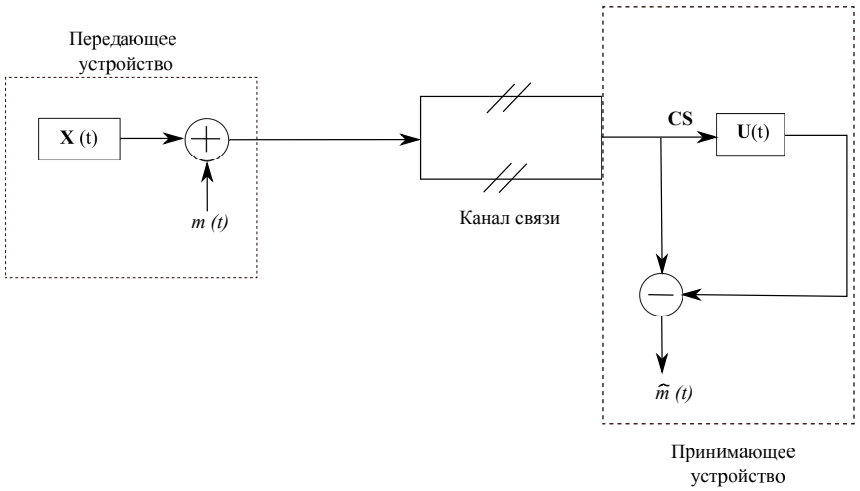


Рис. 1. Схема использования хаотической маскировки для скрытой передачи сообщения

Хаотическая маскировка сообщения эффективна при низком уровне шума в канале связи. При этом уровень шума превышает скрытый сигнал на 35-65 дБ [8]. Наличие шума в канале связи резко снижает качество передаваемой информации. Также к ухудшению качества приёма сообщения приводит рассинхронизация управляющих параметров генераторов шума. Кроме того, хаотическая маскировка характеризуется низким уровнем конфиденциальности [9–11]. Эти недостатки делают метод хаотической маскировки малопригодным для практического применения. Следует отметить, что влияние шумов в канале связи является одной из основных проблем реализации устойчивых систем передачи информации. Достаточно большое количество работ посвящено реализации источников хаотического сигнала. Классификация динамических систем, которые могут быть использованы при генерации хаотического сигнала, несущего встроенное сообщение, приведена в работе [12]. В данной статье приведено условие на скорости выработки полезного сигнала и генерации хаотической несущей, позволяющей снизить вероятность искажения скрытого сообщения. Для них используется понятие оптимальных кодирующих систем. Получен критерий определения таких систем. Кроме хаотической маскировки разработаны системы передачи сообщений на основе модуляции хаотического сигнала [13]. Информационный сигнал используется для изменения параметров хаотического генератора. Такой подход позволяет существенно повысить скорость передачи информации и является устойчивым к шумам в канале связи. В данной статье предложена схема формирования передаваемого скрытого

сообщения, которая позволяет не проводить синхронизацию хаотических генераторов передающей и принимающей стороны.

1. Кодирование на основе ортогональных функций

Построим схему передачи скрытого цифрового сообщения, замаскированного в аналоговом шуме. Для этого на основе цифрового сигнала построим непрерывную функцию, по которой однозначно восстанавливается исходное сообщение. Используем семейство ортогональных функций — множество функций $\{f_i(x)|i \in N\}$ таких, что существует весовая функция $w(x)$ и интервал $[a,b]$, для которых

$$\int_a^b f_i(x) f_j(x) w(x) dx = \delta_{ij},$$

где δ_{ij} — символ Кронекера.

Пусть информация, которую необходимо закодировать, представляет собой последовательность вещественных чисел $k_i (i = 1, ..., n)$. Построим функцию

$$F(x) = \sum_{i=1}^{n} k_i f_i(x).$$

Исходные числа k_i могут быть восстановлены из $F(x)$ на основе свойства ортогональности с помощью простого соотношения:

$$k_i = \int_a^b f_i(x) F(x) w(x) dx.$$

Описанный подход позволяет на основе взвешенной суммы функций и самих функций легко найти коэффициенты при слагаемых этой суммы. Кроме того, даже при искажении взвешенной суммы весовые коэффициенты могут быть восстановлены с удовлетворительной точностью.

В качестве примера рассмотрим семейство ортогональных тригонометрических функций

$$f_n(x) = \sqrt{2} \sin(\pi n x).$$

Ограничим множество возможных весовых коэффициентов двумя значениями: $\{0, 1\}$. Определим функцию $F(x) = f_1(x) + f_3(x) + f_7(x) + f_8(x)$. Возьмём 100 значений функции $f(x)$ с равным шагом в интервале $[0, 1]$. Исказим каждое значение функции равномерным шумом в диапазоне $[-2; 2]$. На рис. 2 представлен график оригинальной и искажённой функций.

Результаты восстановления коэффициентов показаны на рис. 3, где восстановленные значения коэффициентов отмечены меньшими повёрнутыми квадратами. Из рисунка видно, что несмотря на значительный по сравнению с диапазоном значений уровень шума восстановленные коэффициенты имеют значения, близкие к исходным, а при округлении до ближайшего порогового значения (0 или 1) полностью с ними совпадают.

На основе рассмотренного выше подхода кодирование n-битовой строки $(b_1, b_2, \ldots, b_n)$ сообщения состоит из двух шагов:

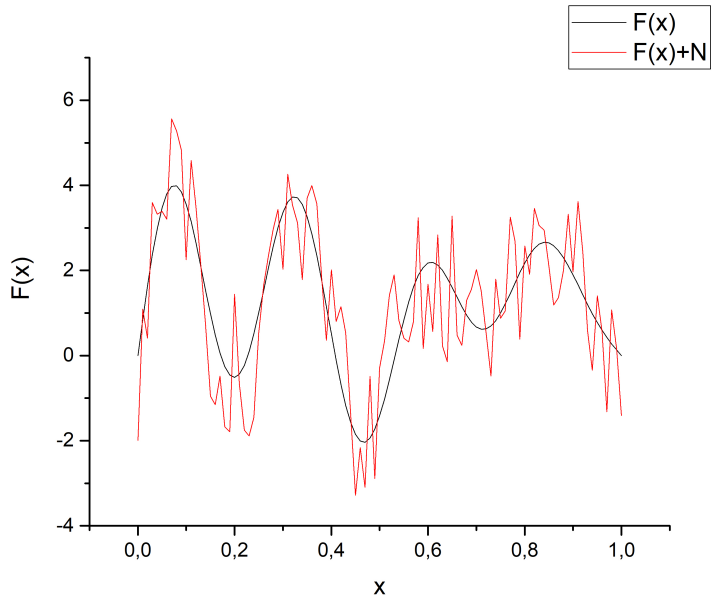


Рис. 2. Пример малого искажения передаваемого сообщения

1. Выбрать n функций: $f_1, f_2, \ldots, f_k$ из семейства ортогональных функций;
2. Построить на отрезке ортогональности [a, b] функцию

$$F(x) = \sum_{i=1}^{n} b_i f_i(x).$$

Процедура извлечения скрытого сообщения состоит из последовательного вычисления интегралов

$$d_i = \int_a^b f_i(x) F(x) w(x) dx \qquad i = (1, \ldots, n).$$

Если $d_i > 0.5$, то соответствующий бит b_i выходного сообщения равен единице, в противном случае ноль.

2. Компьютерный эксперимент

В компьютерном эксперименте было использовано семейство простых тригонометрических функций, обладающих свойством ортогональности на отрезке $[0, 1]$:

$$\{f_n(x) = \sqrt{2}\cos(\pi n x) | n \in N\}.$$

Весовая функция для такого семейства ортогональных функций $w(x) = 1$. Для вычисления интегралов использовался численный метод трапеций. При дискретизации сигнала интервал ортогональности разбивался на 100 частей. Интенсивность скрываемого сигнала выбиралась равной единице. Интенсивность

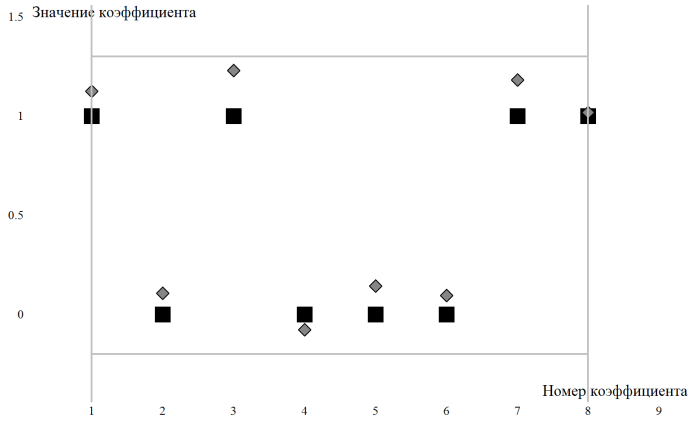


Рис. 3. Исходные и восстановленные коэффициенты

шума варьировалась, начиная от нуля, до значений, не позволяющих извлечь скрытое сообщение с шагом 0.1. Хаотический сигнал моделировался с помощью генератора псевдослучайной последовательности с равномерным распределением. В качестве скрытого сообщения передавались все возможные значения, кодируемые 8-ю битами. Для всех значений передаваемого сообщения вычислялось максимальное количество неверно извлечённых битов при заданной интенсивности хаотического сигнала. График полученной зависимости представлен на рис. 4.

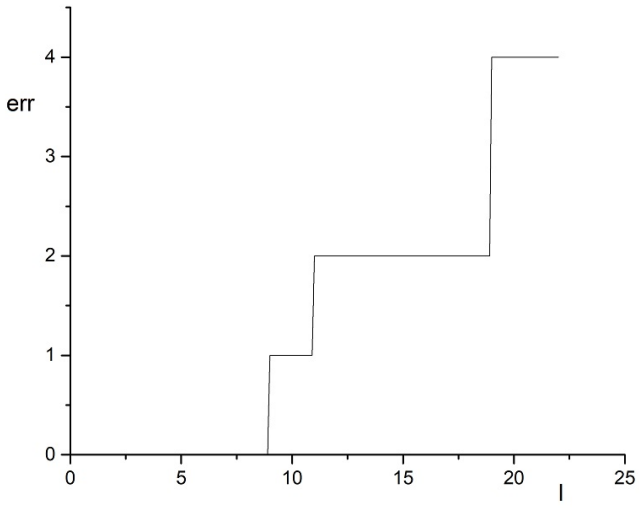


Рис. 4. Зависимость количества изменённых битов в извлечённом сообщении *err* в зависимости от интенсивности хаотического сигнала *I*

Как видно из рис. 4, если интенсивность хаотического сигнала превышает интенсивность скрытого сигнала не более чем в 8 раз, то сообщение восстанавливается без потерь. Если интенсивность хаотического сигнала превышает интенсивность сигнала сообщения не более чем в 18 раз, то неверно опреде-

ляется не более 2-х бит. Следует отметить, что 2 бита — это максимальное значение. Часть сообщений восстанавливается без потерь, а часть — с одним изменённым битом. Пример общего вида сигнала, кодирующего сообщение, и его суммы с хаотическим сигналом при отношении интенсивностей, равном 18, представлены на рис. 5.

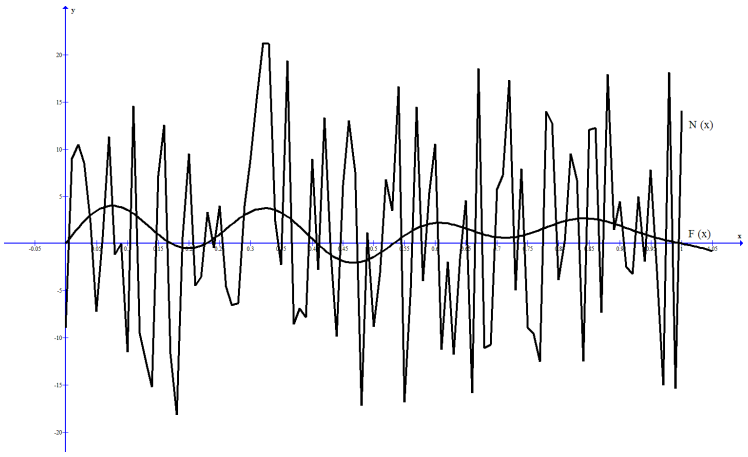


Рис. 5. Сигнал скрытого сообщения $F(x)$ и его сумма с хаотическим сигналом $N(x)$.

Предложенная схема обладает низкой конфиденциальностью. Злоумышленник, зная общий вид используемого семейства ортогональных функций, легко может определить как факт наличия скрытого сообщения, так и его содержимое. Данная проблема может быть решена с помощью использования семейства ортогональных функций с параметром. Причём параметр должен быть вещественным. Например, может быть использовано семейство функций:

$$\{f_n(x) = \sqrt{2}\cos(\pi nax) | n \in N\},$$

где a — вещественный параметр, который держится в секрете обоими абонентами. Как показал компьютерный эксперимент, изменение a на 10 % приводит в среднем к замене 12 % битов в извлекаемом на выходе сообщении. Прямой перебор практически невозможен.

Заключение

В данной статье показана возможность реализации метода хаотической маскировки скрытого сообщения без синхронизации генераторов, передающей и принимающей сторон. Рассмотренный пример простых тригонометрических функций не обеспечивает достаточного уровня целостности сообщения и конфиденциальности. Однако эти проблемы могут быть решены при дальнейшем развитии предложенного метода. Прежде всего при кодировании сообщения могут быть использованы коды, исправляющие ошибки, что существенно повысит устойчивость метода к уровню шума. Во-вторых, вместо тригонометрических функций могут быть использованы другие семейства ортогональных функций.

Проблема конфиденциальности также может быть решена с помощью выбора семейства ортогональных функций с параметрами. В этом случае для обнаружения и извлечения скрытого сообщения необходимо знать набор параметров, которые играют роль ключевых материалов и держатся в секрете. Предложенный метод хаотической маскировки обладает тремя существенными преимуществами перед остальными. Во-первых, не требуется синхронизация генераторов динамического хаоса передающей и принимающей сторон. Во-вторых, может быть использован любой генератор хаотического сигнала. В-третьих, схема устойчива к зашумлению в канале связи, что является большой проблемой для всех предложенных ранее схем.

Литература

1. Белим С.В., Федосеев А.М. Исследование скрытых каналов передачи информации в алгоритме цифровой подписи ГОСТ Р34.10-2001 // Известия ЧНЦ. 2007. № 2(36). С. 20–23.
2. Атмашкин М.И., Белим С.В. Скрытые каналы передачи информации в алгоритме электронной цифровой подписи ГОСТ Р 34.10-2001 // Проблемы информационной безопасности. Компьютерные системы. 2010. № 4. С. 26–35
3. Дмитриев А.С., Панас А.И. Динамический хаос: новые носители информации для систем связи. М. : Физматлит, 2002.
4. Cuomo K.M. Oppenheim A.V., Strogatz S.H. Synchronization of Lorenz-based chaotic circuits with applications to communications // IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing. 1993. V. 40, No. 10. P. 626–633.
5. Dedieu H., Kennedy M.P., Hasler M. Chaos shift keying: modulation and demodulation of a chaotic carrier using self-synchronizing Chua's circuits // IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing. 1993. V. 40, No. 10. P. 634–642.
6. Dmitriev A.S., Panas A.I., Starkov S.O. Experiments on speech and music signals transmission using chaos // Int. J. Bifurcation and chaos. 1995. V. 5, No. 4. P. 1249–1254.
7. Yang T., Chua L. O. Secure communication via chaotic parameter modulation // IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications. 1996. V. 43, No. 9. P. 817–819.
8. Downes P.T. Secure communication using chaotic synchronization. // SPIE. 1993. V. 2038. P. 227–234.
9. Perez G., Cerderia H.A. Extracting Messages Masked by Chaos // Phys Rev. Lett. 1995. V. 74. P. 1970–1973.
10. Short K.M. Unmasking a modulated chaotic communication scheme // Int. J. Bifurcation and chaos. 1996. V. 6. No. 2. P. 367–375.
11. Ponomarenko V.I., Prokhorov M.D. Extracting information masked by the chaotic signal of a time-delay system // Phys. Rev. E. 2002. V. 66. 026215.
12. Baptista M.S., Macau E.E., Grebogi C. Conditions for efficient chaos-based communication // Chaos. 2003. P. 145–150.
13. Дмитриев А.С., Панас А.И., Старков С.О. Динамический хаос как парадигма современных систем связи // Зарубежная радиоэлектроника. Успехи современной радиоэлектроники. 1997. No. 10. С. 4–26.

CHAOTIC MASKING SCHEME FOR MESSAGES BASED ON ORTHOGONAL FUNCTIONS

S.V. Belim
Dr.Sc. (Phys.-Math.), Professor, e-mail: sbelim@mail.ru
Y.S. Rakitskiy
Ph.D. (Eng.), Associate Professor, e-mail: yrakitsky@gmail.com

Dostoevsky Omsk State University, Omsk, Russia

Abstract. The method of chaotic masking of a discrete signal is proposed in the article. The proposed approach does not require the synchronization of chaotic generators and is resistant to noise in the communication channel. These properties remove the main problem of chaotic masking associated with the desynchronization of control parameters. The basic idea is to use message coding using orthogonal functions. Next, a simple summation of a hidden message with a chaotic signal is applied. The extraction of the message can be performed on the basis of the orthogonality property without subtraction of the chaotic component.

Keywords: orthogonal functions, signal, masking.

Дата поступления в редакцию: 20.11.2018

Авторам

Предоставляемые данные и документы

Автор предоставляет в редакцию:
- рукопись статьи в формате LaTeX (см. требования к оформлению);
- список из трёх экспертов по тематике статьи, давших согласие написать рецензию на представленную работу[1];
- экспертное заключение о возможности открытого опубликования.

Лицензирование

Согласно ГК РФ ст. 1286 лицензионный договор с автором для публикации в периодических изданиях может быть заключён в устной форме. Сам факт получения рукописи статьи редколлегией журнала «Математические структуры и моделирование» является акцептом (принятием) лицензионного договора.

Все статьи в журнале «Математические структуры и моделирование» публикуются под лицензией Creative Commons Attribution 4.0 International (CC-BY). Текст лицензии находится по адресу `https://creativecommons.org/licenses/by/4.0/legalcode`.

Требования к оформлению рукописи

К публикации принимаются рукописи объёмом не более 16 страниц.

Авторам необходимо предоставить следующую информацию на русском и английском языках:
- название статьи;
- список авторов с указанием
  - фамилии, имени и отчества,
  - учёного звания,
  - учёной степени,
  - должности,
  - места работы или учёбы,
  - действующего адреса электронной почты;
- аннотация (абстракт) объёмом от 100 до 250 слов;
- список ключевых слов.

Автор также указывает УДК (универсальный десятичный код) статьи. Его можно подобрать по тематике статьи в справочнике `http://msm.univer.omsk.su/udc/`.

Библиографические ссылки оформляются согласно ГОСТ 7.0.5–2008.

Рукопись статьи представляется в редакцию по электронной почте в двух форматах pdf и tex. Статья должна быть набрана с использованием макропакета LaTeXи стиля msmb.cls, предоставляемого редакцией `http://msm.univer.omsk.su/files/msmb.zip`. Рекомендуется установить компилятор MiKTeX, так как именно им пользуются в редакции.

Отклонения в оформлении рукописи от приведённых правил позволяют редколлегии принять решение о снятии статьи с публикации. Статья может быть отклонена по причинам несоответствия тематике журнала или в связи с низким уровнем качества научного исследования.

В статье запрещается переопределять стандартные команды и окружения.

Нумеруемые формулы необходимо выделять в отдельную строку.

Нумерация только арабскими цифрами в порядке возрастания с единицы. Нумеровать следует только те формулы, на которые в тексте имеются ссылки.

[1]Необходимы полные данные экспертов (место работы, учёная степень, должность), с указанием способа связи с ними (e-mail, телефон). Редколлегия может обратиться к одному из экспертов из предложенного списка с просьбой написать рецензию или может назначить рецензента из собственного списка.

Запрещается использовать в формулах буквы русского алфавита. Если без них никак не обойтись, то следует использовать команду `\mbox{...}`.

Все рисунки и таблицы должны иметь подпись, оформленную с помощью команды `\caption{...}`.

Файлы с рисунками необходимо представить в формате PDF или EPS (использовать редакторы векторной графики типа InkScape, Adobe Illustrator или Corel Draw).

Используйте стандартные команды переключения на готический, каллиграфический и ажурный шрифты: `\mathfrak`, `\mathcal` и `\mathbb`.

Не допускается заканчивать статью рисунком или таблицей.

В списке литературы обязательно указание следующих данных: для книг — фамилии и инициалы авторов, название книги, место издания, издательство, год издания, количество страниц; для статей — фамилии и инициалы авторов, название статьи, название журнала, год издания, том, номер (выпуск), страницы начала и конца статьи (для депонированных статей обязательно указать номер регистрации).

Кавычки в русском тексте («абвгд») должны быть угловыми, в английском — прямыми верхними кавычками ("abcdefg" или “abcdefg”).

Обязательна расшифровка сокращений при первом вхождении термина. Например: ... искусственный интеллект (ИИ)...

Порядок рецензирования

Первичная экспертиза проводится главным редактором (заместителем главного редактора). При первичной экспертизе оценивается соответствие статьи тематике журнала, правилам оформления и требованиям, установленным редакцией журнала к научным публикациям.

Все статьи, поступившие в редакцию научного журнала «Математические структуры и моделирование», проходят через институт рецензирования.

Рецензент выбирается главным редактором журнала из числа членов редколлегии или ведущих специалистов по профилю данной работы.

Рецензенты уведомляются о том, что присланные им рукописи являются частной собственностью авторов и относятся к сведениям, не подлежащим разглашению. Рецензентам не разрешается делать копии статей для своих нужд.

Срок для написания рецензии устанавливается по согласованию с рецензентом.

Рецензия должна раскрывать актуальность представленного материала, степень научной новизны исследования, определять соответствие предлагаемого к публикации текста общему профилю издания и стиль изложения.

Рецензент выносит заключение о возможности опубликования статьи: «рекомендуется», «рекомендуется с учётом исправления замечаний, отмеченных рецензентом» или «не рекомендуется». В случае отрицательной рецензии редакция направляет автору мотивированный отказ, заверенный главным редактором или его заместителем.

В случае несогласия с мнением рецензента автор статьи имеет право предоставить аргументированный ответ в редакцию журнала. Статья может быть направлена на повторное рецензирование, либо на согласование в редакционную коллегию.

При наличии в рецензии рекомендаций по исправлению и доработке статьи автору направляется текст рецензии с предложением учесть их при подготовке нового варианта статьи или аргументированно (частично или полностью) их опровергнуть. Доработанная (переработанная) автором статья повторно направляются на рецензирование и рассматривается в общем порядке. В этом случае датой поступления в редакцию считается дата возвращения доработанной статьи.

После принятия редколлегией решения о допуске статьи к публикации автор информируется об этом и указываются сроки публикации.

Оригиналы рецензий хранятся в редакции в течение пяти лет.

Авторская этика

Авторы публикаций должны гарантировать, что в список авторов включены только лица, соответствующие критериям авторства (лица, внёсшие значительный вклад в работу), и что заслуживающие авторства исследователи не исключены из списка авторов.

Должны работать вместе с редакторами или издателями для скорейшего исправления своих работ в случае обнаружения в них ошибок или упущений после публикации.

Обязаны незамедлительно уведомлять редакцию в случае обнаружения ошибки в любой поданной ими на публикацию, принятой для публикации или уже опубликованной работе.

Не вправе копировать из других публикаций ссылки на работы, с которыми они сами не ознакомились; цитаты и ссылки на другие работы должны быть точными и оформленными в соответствии с предъявляемыми требованиями.

Должны ссылаться максимально правильно и точно на имеющие отношение к публикации предыдущие работы как других исследователей, так и самих авторов, обращаясь, прежде всего к первоисточнику; дословное воспроизведение собственных работ и их перефразирование неприемлемы, они могут быть использованы лишь в качестве основы для новых выводов.

Необходимо указывать авторство данных, текста, рисунков и идей, которые автор получил из других источников — они не должны представляться, как принадлежащие автору публикации; прямые цитаты из работ других исследователей должны выделяться кавычками и соответствующей ссылкой.

Должны соблюдать нормы законодательства о защите авторских прав; материалы, защищённые авторским правом (например, таблицы, цифры или крупные цитаты), могут воспроизводиться только с разрешения их владельцев.

Памятка для перевода должностей, учёных степеней и званий на английский язык

Профессор = Professor
Доцент = Associate Professor
Старший преподаватель = Assistant Professor
Преподаватель = Instructor
Ассистент = Instructor
Аспирант = Postgraduate Student или Ph.D. Student
Соискатель = Ph.D. Doctoral Candidate
Магистрант = Master's Degree Student
Студент = Student
д.ф.-м.н. = Dr.Sc. (Phys.-Math.)
к.ф.-м.н. = Ph.D. (Phys.-Math.)
д.т.н. = Dr.Sc. (Eng.)
к.т.н. = Ph.D. (Eng.)
Инженер-программист = Software Engineer
Старший/младший научный сотрудник = Senior/Junior Scientist Researcher

Электронная почта для отправки статей

`lavrov@omsu.ru` — зам. главного редактора (ответственный за выпуск) Д.Н. Лавров.

Научный журнал

Математические структуры и моделирование

№4(48)

Главный редактор
А.К. Гуц

Зам. глав. ред., выпускающий редактор
Д.Н. Лавров

Зам. глав. ред., технический редактор
Н.Ф. Богаченко

Корректор:
И.Н. Баловнева
Проверка корректности перевода:
А.Н. Кабанов

Адрес научной редакции
644077, Омская обл., г. Омск, пр-т Мира, д. 55а,
Омский государственный университет

E-mail: guts@omsu.ru, lavrov@omsu.ru
Электронная версия журнала:
http://msm.univer.omsk.su
http://msm.omsu.ru

Подписано в печать 14.12.2018. Формат 60 × 84 1/8.
Усл. печ. л. 19,18. Тираж 100 экз. Заказ № 315.

Отпечатано на полиграфической базе издательства ОмГУ им. Ф.М. Достоевского
644077, Омская обл., г. Омск, пр-т Мира, д. 55а

ISSN 2222-8772