# ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

**5(213) 2014**

**ТЕОРЕТИЧЕСКИЙ И ПРИКЛАДНОЙ НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ**

**Издается с ноября 1995 г.**

**УЧРЕДИТЕЛЬ**

**Издательство "Новые технологии"**


## СОДЕРЖАНИЕ




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

# INFORMATION TECHNOLOGIES
# INFORMACIONNYE TEHNOLOGII

5(213)
2014

THEORETICAL AND APPLIED SCIENTIFIC AND TECHNICAL JOURNAL

Published since November 1995

ISSN 1684-6400


## CONTENTS




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

УДК 519.233

**Б. Г. Кухаренко,** канд. физ.-мат. наук, ст. науч. сотр., вед. науч. сотр.,
Институт машиноведения РАН, г. Москва, e-mail: kukharenko@imash.ru,
**М. О. Солнцева,** аспирант, Московский физико-технический институт (ГУ),
e-mail: solnceva.chalei@gmail.com

# Кластеризация управляемых объектов на основе сходства их многомерных траекторий

*Для кластеризации многомерных траекторий применяется метод полиномиальных регрессий с обучением параметров посредством алгоритма ожидания и максимизации правдоподобия. Особенностью метода является одновременное выравнивание траекторий в многомерном пространстве и во времени. Эффективность метода кластеризации демонстрируется на примере анализа траекторий движения самолетов в воздушном пространстве аэропорта.*

***Ключевые слова:*** *анализ данных, многомерные траектории, кластеризация, полиномиальная регрессия, алгоритм ожидания и максимизации правдоподобия*

**B. G. Kukharenko, M. O. Solntseva**

# Clustering Objects under Control by Similarity of Their Multidimensional Trajectories

*For clustering multidimensional trajectories polynomial regression method with parameter leaning by the Expectation-Maximization algorithm is in use. The method based on polynomial regression is characterized by the joint clustering and continuous alignment of curve sets in time and space. Efficiency of the clustering method is demonstrated by analysis of flight tracks in airport space.*

***Keywords:*** *data analysis, multidimensional trajectories, clustering, polynomial regression, Expectation-Maximization algorithm*

## Введение

Современное развитие приборов мобильной связи и систем теле- и видеонаблюдения, в том числе использование космических спутников, способствует накоплению громадного количества данных о траекториях перемещения людей, наземного транспорта, морских, воздушных судов и т. п. Классификация траекторий движущихся объектов, т. е. построение моделей для предсказания классов таких объектов (согласно их траекториям и различным дополнительным характеристикам), используется в ряде практических приложений для анализа транспортных сетей [1, 2], создания военных систем наблюдения [3], морского пограничного контроля [4], оптимизации загруженности взлетно-посадочной полосы и обеспечения безопасности воздушного пространства в крупных аэропортах [5]. Задача кластеризации траекторий движения возникает и при управлении мультиробототехническими системами. Она обусловлена необходимостью предсказания движения для организации взаимодействия объектов системы на основе выделяемых патернов в траекториях движения [6]. Кластеризация объектов робототехнических систем также упрощает контроль их согласованного перемещения [7, 8].

Чтобы исключить в любой рассматриваемой системе нежелательную конкуренцию (столкновения) между объектами, совершающими движение по сходным траекториям, следует разнести их во времени и (или) в пространстве. В этом случае также выполняется предварительная кластеризация движущихся объектов системы в соответствии с их траекториями. Для кластеризации кривых, представляющих траектории движения, применяют различные методы, из которых наиболее эффективен метод локальной полиномиальной регрессии [9]. Для

кластеризации траекторий при обучении полиномиальной регрессии в дальнейшем используется итеративный алгоритм ожидания и максимизации правдоподобия (EM-алгоритм) [10]. Этот объединенный подход обеспечивает выравнивание кривых, осуществляемое в координатном пространстве и во времени одновременно. Задача выравнивания кривых с использованием модели априорных вероятностей, заданных на множестве возможных выравниваний, решается EM-алгоритмом, что формализует так называемый подход Прокруста для анализа кривых [11]. Априорные вероятности выравнивания интегрируются в модель смеси конечного числа распределений, на основе которой выполняется кластеризация. В настоящей работе для выполнения одновременной кластеризации и выравнивания кривых используется полиномиальная модель.

Ранее модель смеси гауссовых распределений с обучением посредством EM-алгоритма использовалась для совместной кластеризации и выравнивания при корректировке изображений, которые подвергались различным линейным преобразованиям [12]. Однако такая модель рассматривает преобразования в дискретном пространстве пикселей, в то время как подход, применяемый в настоящей работе, направлен на моделирование кривых и допускает произвольное и непрерывное их выравнивание в пространстве и во времени. Эффективность такого подхода демонстрируется на примере анализа данных радаров международного аэропорта г. Сан-Франциско, находящихся в свободном доступе на сайте https: //c3.nasa.gov/dashlink/resources/132/

## Модель одновременной кластеризации и выравнивания одномерных временных рядов

Рассмотрим одновременные кластеризацию и выравнивание одномерных временных рядов, т. е. векторов переменной длины, представляющих координатные временные зависимости. Каждый вектор $\mathbf{z}_i \in \mathbb{R}^{N_i \times 1}$, $i = 1, 2, ...$, состоит из последовательности измерений координатной зависимости $z_i = z_i(t)$ в моменты времени $\mathbf{t}_i \in \mathbb{R}^{N_i \times 1}$. В работе [13] модели смеси регрессий эффективно используются для кластеризации одномерных векторов переменной длины. Вектор $\mathbf{z}_i$ моделируется регрессионной моделью

$$\mathbf{z}_i = \mathbf{T}_i\boldsymbol{\beta} + \boldsymbol{\varepsilon}_i, \qquad (1)$$

где $\boldsymbol{\beta}$ — вектор коэффициентов регрессии размерности $(q + 1) \times 1$; $\boldsymbol{\varepsilon}_i$ — гауссов шум с нулевым средним; $\mathbf{T}_i$ — регрессионная матрица. Матрица $\mathbf{T}_i$ зависит от типа используемой регрессионной модели. В случае полиномиальной регрессии $\mathbf{T}_i$ имеет вид стандартной матрицы Вандермонта

$$\mathbf{T}_i = \begin{bmatrix} 1 & t_i[1] & (t_i[1])^2 & \dots & (t_i[1])^q \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ 1 & t_i[N_i] & (t_i[N_i])^2 & \dots & (t_i[N_i])^q \end{bmatrix}. \qquad (2)$$

В основе модели одновременной кластеризации и выравнивания лежит модель смеси регрессий, в которую вводятся четыре независимых параметра преобразований выравнивания и масштабирования во времени и пространстве $\{\Phi_i\} = \{a_i, b_i, c_i, d_i\}$ (параметры $a_i$ и $b_i$ описывают масштабирование и сдвиг во времени, а параметры $c_i$ и $d_i$ — масштабирование и смещение в пространстве измерений). Полиномиальная регрессия для одномерного случая имеет вид

$$\mathbf{z}_i = c_i\boldsymbol{\Upsilon}_i\boldsymbol{\beta}_k + d_i + \boldsymbol{\varepsilon}_i, \qquad (3)$$

где матрица $\boldsymbol{\Upsilon}_i$ получается из $\mathbf{T}_i$ (2) подстановкой $\mathbf{t}_i \to a_i\mathbf{t}_i - b_i$; $\boldsymbol{\beta}_k$ определяет модель регрессии для $k$-го кластера ($k = \overline{1, K}$); $\boldsymbol{\varepsilon}_i$ — гауссов шум с нулевым средним и дисперсией $\sigma_k^2\mathbf{I}$. Поэтому распределение плотности условной вероятности имеет вид

$$p_k(\mathbf{z}_i|a_i, b_i, c_i, d_i) = \mathcal{N}(\mathbf{z}_i|c_i\boldsymbol{\Upsilon}_i\boldsymbol{\beta}_k + d_i, \boldsymbol{\sigma}_k^2\mathbf{I}). \qquad (4)$$

Плотность вероятности для кривой $\mathbf{z}_i$ однозначно задается соответствующим множеством параметров $\{\Phi_i\}$, которые подлежат определению. Задача кластеризации кривых решается как стандартная задача оценки значений скрытых переменных. Каждый из параметров преобразования в формулах (3) и (4) рассматривается как характерная для $\mathbf{z}_i$ случайная переменная с заранее известным распределением вероятности для кластера. Параметры преобразования и параметры модели оцениваются одновременно посредством EM-алгоритма.

## Априорные распределения вероятностей для параметров преобразования

Априорные распределения вероятностей для параметров преобразования выбираются таким образом, чтобы тождественное преобразование являлось наиболее вероятным. С учетом этого эффективной априорной вероятностью является Гауссово распределение $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\sigma}^2)$ со средним $\boldsymbol{\mu}$ и дисперсией $\boldsymbol{\sigma}^2$. Поэтому априорные распределения плотности вероятности для параметров преобразования времени определяются как

$$a_i \sim \mathcal{N}(1, r_k^2),\ b_i \sim \mathcal{N}(0, s_k^2), \qquad (5)$$

и априорные распределения вероятности для параметров преобразования координаты задаются как

$$c_i \sim \mathcal{N}(1,\ u_k^2),\ d_i \sim \mathcal{N}(1,\ v_k^2). \qquad (6)$$

В формуле (6) параметры дисперсии $u_k^2$ и $v_k^2$ зависят от кластера. Однако любое подмножество этих параметров может быть выравнено ("сшито") между кластерами, если это требуется для конкретного приложения. Отметим, что априорные распределения вероятности технически допускают отрицательное масштабирование во времени и в пространстве измерений. Хотя этот результат не типичен, можно задать другие априорные распределения вероятности, например, логарифмически нормальные, чтобы не допустить отрицательного масштабирования. Следует заметить, что дисперсии для параметров априорных вероятностей выводятся на основе данных, полученных в результате работы EM-алгоритма. Ниже модель (3)—(6) совместной кластеризации и выравнивания кривых обобщается на случай многомерного пространства измерений.

## Кластеризация и выравнивание многомерных кривых

Рассмотрим выравнивание кривых в многомерном пространстве. Ранее предполагалось, что вектор $\mathbf{z}_i \in \mathbb{R}^{N_i \times 1}$ состоит из последовательности измерений одномерной координатной зависимости в моменты времени $\mathbf{t}_i \in \mathbb{R}^{N_i \times 1}$. Однако во многих приложениях такие зависимости от времени являются многомерными. Таким образом, каждому моменту времени $t_i[j], j = \overline{1, N_i}$, соответствует многомерный вектор размерности $D$. Обозначим многомерные кривые как $\mathbf{Z}_i \in \mathbb{R}^{N_i \times D}$, полученные в результате измерений в моменты времени $\mathbf{t}_i \in \mathbb{R}^{N_i \times 1}$. Тогда матрица $\mathbf{Z}_i = \{Z_i[j\ ;\ l], j = \overline{1, N_i}, l = \overline{1, D}\}$ состоит из $D$ столбцов, таких что каждый $l$-й столбец $\mathbf{z}_i^{(l)} = \{Z_i[j; l], j = \overline{1, N_i}\}$, $l = \overline{1, D}$ содержит последовательность измерений $l$-й одномерной координатной зависимости для $i$-й рассматриваемой переменной. То есть столбец $\mathbf{z}_i^{(l)}$, соответствующий одномерному вектору $\mathbf{z}_i \in \mathbb{R}^{N_i \times 1}$ в формуле (1), оказывается вложенным в матрицу многомерной кривой $\mathbf{Z}_i \in \mathbb{R}^{N_i \times D}$.

**Регрессионные модели выравнивания в многомерном пространстве.** Для многомерной кривой $\mathbf{Z}_i \in \mathbb{R}^{N_i \times D}$ вектор измерения каждой координаты описывается независимой регрессионной моделью

$$\mathbf{z}_i^{(l)} = \mathbf{T}_i\boldsymbol{\beta}_{kl} + d_{il} + \boldsymbol{\varepsilon}_i;\ d_{il} \propto \mathcal{N}(0,\ \nu_{kl}^2);$$
$$\boldsymbol{\varepsilon}_i \propto \mathcal{N}(\mathbf{0}_{N_i \times 1},\ \sigma_{kl}^2 \mathbf{I}_{N_i \times N_i}), \qquad (7)$$

где $\mathbf{T}_i$ — матрица Вандермонта (2). Матрица $\boldsymbol{\beta}_{kl}$ задает коэффициенты регрессии для $l$-го измерения (т. е. коэффициенты регрессии для $l$-го столбца $\mathbf{Z}_i \in \mathbb{R}^{N_i \times D}$); $d_{il}$ задает смещение для $l$-го измерения; $\mathbf{0}_{N_i \times 1}$ — вектор с нулевыми компонентами размерности $N_i \times 1$ и $\mathbf{I}_{N_i \times N_i}$ — тождественная матрица размерности $N_i \times N_i$. Использование параметров $\nu_{kl}^2$, $\sigma_{kl}^2$ позволяет рассматривать дисперсию по каждому измерению независимо.

На основе модели (7) плотность вероятности многомерной кривой $\mathbf{Z}_i \in \mathbb{R}^{N_i \times D}$ и число параметров $D$ смещения $\{d_{il},\ l = \overline{1, D}\}$ определяется следующим образом:

$$p(\mathbf{Z}_i,\ d_{i1},\ ...,\ d_{iD}) =$$
$$= \prod_{l=1}^{D} \mathcal{N}(\mathbf{z}_i^{(l)} | \mathbf{T}_i\boldsymbol{\beta}_{kl} + d_{il},\ \sigma_{kl}^2 \mathbf{I}_{N_i \times N_i}) \mathcal{N}(d_{il} | 0,\ \nu_{kl}^2). \qquad (8)$$

Плотность вероятности (8) учитывает два необходимых условия: во-первых, все координаты пространства кривых $\mathbf{Z}_i \in \mathbb{R}^{N_i \times D}$ и, во-вторых, предполагается, что для каждого измерения $l$ существует собственное множество параметров смещения $\{\Phi_i\}_l$. Далее предполагается, что эти два условия всегда выполняются.

Плотность безусловного распределения (компонент многомерной случайной величины) $p(\mathbf{Z}_i)$ представляется в виде произведения $p(\mathbf{Z}_i) = \prod_{l=1}^{D} p(\mathbf{z}_i^{(l)})$, поэтому логарифм правдоподобия $\mathbb{Z} = \{\mathbf{Z}_i, i = \overline{1, N}\}$ имеет вид

$$\log(p(\mathbb{Z})) = \sum_{i=1}^{N} \log(p(\mathbf{Z}_i)) =$$
$$\sum_{i=1}^{N} \sum_{l=1}^{D} \log(\int p(\mathbf{z}_i^{(l)} | d_{il}) p(d_{il}) dd_{il}). \qquad (9)$$

Интегрирование в формуле (9) выполняется аналитически, что приводит к следующему виду логарифма правдоподобия:

$$\log(p(\mathbb{Z})) = \\ = \sum_{i=1}^{N}\sum_{l=1}^{D}\log(\mathcal{N}(\mathbf{z}_i^{(l)}|\mathbf{T}_i\boldsymbol{\beta}_{kl},\ \mathbf{1}_{N_i\times N_i}\nu_{kl}^2 + \sigma_{kl}^2\mathbf{I}_{N_i\times N_i})), \quad (10)$$

где $\mathbf{1}_{N_i\times N_i}$ — единичная матрица размерности $N_i \times N_i$.

**Регрессионные модели выравнивания многомерных кривых во времени.** Поскольку независимые координаты многомерного пространства смещаются и масштабируются независимо, для выравнивания в пространстве рассматриваются $D$ отдельных параметров преобразования. Однако изменение во времени каждой координаты траектории происходит в одном и том же временном масштабе, следовательно, параметры преобразования времени должны быть распределены одинаково по всем $D$ измерениям. Поэтому каждому из векторов $\mathbf{z}_i^{(l)} \in \mathbb{R}^{N_i\times 1}$ соответствует единственный параметр $b_i$. Тогда условная плотность вероятности $\mathbf{Z}_i \in \mathbb{R}^{N_i\times D}$ определяется как

$$p(\mathbf{Z}_i|b_i) = \prod_{l=1}^{D} p(\mathbf{z}_i^{(l)}|b_i) = \\ = \prod_{l=1}^{D}\mathcal{N}(\mathbf{z}_i^{(l)}|\boldsymbol{\Upsilon}_i\boldsymbol{\beta}_{kl},\ \sigma_{kl}^2\mathbf{I}_{N_i\times N_i}), \quad (11)$$

где используется одно $b_i$ для всех $l = \overline{1, D}$ и матрица $\boldsymbol{\Upsilon}_i$ получается из $\mathbf{T}_i$ (2) подстановкой $\mathbf{t}_i \to \mathbf{t}_i - b_i$. Условная плотность вероятности разлагается на множители, а безусловная плотность вероятности $p(\mathbf{Z}_i)$ не разлагается, поскольку различные измерения в пространстве оказываются связанными через параметр смещения времени $b_i$. Следовательно, для логарифма правдоподобия $\mathbb{Z} = \{\mathbf{Z}_i,\ i = \overline{1, N}\}$ имеем

$$\log(p(\mathbb{Z})) = \\ = \sum_{i=1}^{N}\log(p(\mathbf{Z}_i)) = \sum_{i=1}^{N}\log(\int p(b_i)\prod_{l=1}^{D} p(\mathbf{z}_i^{(l)}|b_i)db_i). \quad (12)$$

Под знаком интеграла в формуле (12) находится произведение вероятностей по всем измерениям пространства, что приводит к сложным вычислениям. Однако с помощью методов Монте-Карло можно вычислить аппроксимацию логарифма правдоподобия (12) следующим образом:

$$\log(p(\mathbb{Z})) \approx \sum_{i=1}^{N}\log\left(\sum_{m=1}^{M}\prod_{l=1}^{D} p(\mathbf{z}_i^{(l)}|b_i^{(m)})\right) - N\log(M), \quad (13)$$

где

$$b_i^{(m)} \propto \mathcal{N}(0,\ \eta^2),\ m = \overline{1, M}. \quad (14)$$

## EM-алгоритм

Сложность EM-алгоритма, обеспечивающего одновременное обучение параметров модели и преобразования, является линейной функцией от полного числа точек $\sum_{i=1}^{N} N_i$ многомерных траекторий $\mathbf{Z}_i$ [10]. Пусть $\boldsymbol{\pi}_i$ — принадлежность $\mathbf{Z}_i$ к некоторому кластеру. Параметры $\{\Phi_i\}$ и принадлежности к кластеру рассматриваются как скрытые переменные. В таком случае логарифм правдоподобия для полного набора данных определяется как логарифм совместного правдоподобия множества $\mathbb{Z} = \{\mathbf{Z}_i,\ i = \overline{1, N}\}$ и скрытых переменных $\Phi = \{\Phi_i,\ \pi_i\}$, что в соответствии с формулой (11) может быть записано в виде суммы (по всем $N$ кривым) логарифма от произведения веса кластера $\alpha_{\pi_i}$ и совместного распределения вероятности (8), зависящего от кластера

$$L_c = \sum_{i=1}^{N}\sum_{l=1}^{D}\log(\alpha_{\pi_i} p_{\pi_i}(\mathbf{z}_i^{(l)}|\Phi_i)p_{\pi_i}(\Phi_i)). \quad (15)$$

На E-шаге оценивается распределение вероятности $p(\Phi_i,\ \pi_i|\mathbf{Z}_i)$ и затем используется в качестве следующего ожидаемого распределения в (15). На следующей итерации это ожидаемое распределение используется на M-шаге для оценки параметров модели в $p_{\pi_i}(\mathbf{z}_i^{(l)}|\Phi_i)$.

## Численный эксперимент

Описываемый в настоящей работе подход к кластеризации кривых тестируется на реальных данных радара TRACON (Terminal Radar Approach Control), регистрирующего траектории полетов воздушных судов над заливом Сан-Франциско (данные находятся в открытом доступе на сайте https://c3.nasa.gov/dashlink/resources/132/). Рассматривается область воздушного пространства над тремя крупными аэропортами: Окленда, Сан-Франциско и Сан-Хосе (Oakland, San Francisco and San Jose International Airports). Это пространство представляет собой цилиндр радиуса 80 км с центром над аэропортом Окленда и высотой 6 км. Данные содержат трехмерные координаты и абсолютные скорости воздушных судов через равные интервалы времени (5 с) с момента обнаружения радаром и до самой нижней регистрируемой радарами высоты. Помимо основной информации в записях указываются дополнительные

сведения о типе совершаемой операции (прибытии или отправлении), о пункте вылета и назначения и др.

В настоящей работе анализируются траектории первых 30 самолетов, которые приземлились в аэропорту 1 января 2006 г. Чтобы их случайные маневры до начала снижения не искажали общей тенденции движения, анализируются только 160 последних точек каждой траектории. Разница между моментами времени последовательной регистрации самолета (~5 с) определяется угловой скоростью вращения радара. Начало координат (0, 0, 0) совпадает с положением радаров.

На рис. 1 (см. третью сторону обложки) показано трехмерное представление для 30 анализируемых траекторий самолетов (идущих на посадку в трех международных аэропортах, в пространстве над заливом Сан-Франциско). В численном эксперименте эти траектории разделяются на пять кластеров. В каждом из пяти кластеров, с помощью описанной выше регрессионной модели (11) выравнивания кривых в многомерном пространстве, определяется собственный тренд.

На рис. 2 (см. третью сторону обложки) показано изменение координат самолетов отдельно по осям ($x$, $y$, $z$) в соответствии с последовательностью моментов времени регистрации радарами. Как и на рис. 1, число временных точек регистрации и интервалы между ними (5 с) — одинаковые для всех траекторий движения самолетов. Разница в реальном времени для самолетов, начинающих снижение, не учитывается, поэтому по горизонтальной оси на рис. 2 отложен индекс моментов времени регистрации радарами. Это позволяет увидеть сходство формы траекторий самолетов, принадлежащих одному кластеру при изменении каждой координаты во времени. Линия тренда (жирная линия на рис. 2) представляет собой некоторую обобщенную форму траекторий кривых в кластере. Например, на рис. 2, *а* два кластера имеют почти линейную форму линии тренда. На рис. 2, *б* для этих же самых кластеров форма линии тренда напоминает прямой угол. На рис. 2, *в* можно заметить линии тренда с выраженной *s*-образной формой кривой. Несмотря на то, что снижение самолетов происходит не синхронно в реальном времени, их сходные траектории отражают типичные маршруты посадки, когда самолеты выстраиваются в "караван", ожидая своей очереди приземления на посадочную полосу. Рис. 2 является иллюстрацией эффективности метода кластеризации кривых, представленного в настоящей работе.

На рис. 3 (см. третью сторону обложки) данные по кластеризации траекторий самолетов, совершающих посадку (см. рис. 1), показаны в трехмерном пространстве. Как видно на рис. 3, каждый кластер представляется своим "посадочным" паттерном, который проявляет наибольшее сходство со всеми кривыми, вошедшими в кластер, что достигается благодаря поиску максимального значения логарифма правдоподобия (13).

## Заключение

В настоящей работе метод одновременной кластеризации и выравнивания кривых во времени и пространстве демонстрируется на примере кластеризации траекторий самолетов, идущих на посадку в зоне крупного международного аэропорта. Эффективность кластеризации траекторий в трехмерном пространстве достигается благодаря использованию модели полиномиальной регрессии с обучением параметров посредством EM-алгоритма.

### Список литературы


1. **Кухаренко Б. Г., Солнцева М. О.** Принцип минимальной длинны при анализе графов с разреженными матрицами смежности в задачах кластеризации их узлов // Информационные технологии. 2013. № 7. С. 37—42.

2. **Солнцева М. О., Кухаренко Б. Г.** Применение методов кластеризации узлов на графах с разреженными матрицами смежности в задачах логистики // Труды МФТИ. 2013. Т. 5, № 3 (19). С. 75—83.

3. **Zheng Y., Zhou X.,** eds. Computing with Spatial Trajectories. New York; Dordrecht, Heidelberg, London: Springer. 2011.

4. **Greidanus H., Kourti N.** Findings of the DECLIMS project — Detection and classification of marine traffic from space / Sawaya-Lacoste H., Ouwehand L., eds. // Proceedings of SEASAR 2006 "Advances in SAR Oceanography from Envisat and ERS Missions". Frascati, Italy. 23—26 January 2006. Noordwijk: The European Space Agency. 2006. P. 25—33.

5. **Gariel M., Srivastava A. N., Feron E.** Trajectory clustering and an application to airspace monitoring // IEEE Transactions on Intelligent Transportation Systems. 2011. V. 12, Is. 4. P. 1511—1524.

6. **Sung C., Feldman D., Rus D.** Trajectory clustering for motion prediction, Intelligent Robots and Systems (IROS) // Proceedings of 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2012). Vilamoura-Algarve, Portugal: IEEE. 2012. P. 1547—1552.

7. **Солнцева М. О., Кухаренко Б. Г.** Сокращение неинформативной части панорамного кадра по методу швов при управлении группами объектов // Сб. науч. тр. МФТИ "Математические и информационные модели управления". М.: МФТИ. 2013. С. 91—97.

8. **Кухаренко Б. Г., Солнцева М. О.** Использование методов сокращения фона при сегментировании телеметрических изображений для идентификации групп объектов // Информационные технологии. 2014. № 2. С. 3—8.

9. **Fan J., Gijbels I.** Local Polynomial Modelling and its Applications. London: Chapman and Hall. 1996.

10. **Gaffney S., Smyth P.** Joint probabilistic curve clustering and alignment // Saul L., Weiss Y., Bottou L., eds. Proceedings of Neural Information Processing Systems (NIPS 2004). December 13—18, 2004, Vancouver, British Columbia, Canada. Advances in Neural Information Processing Systems. Cambridge, MA: MIT Press. 2005. V. 17. P. 473—480.

11. **Ramsay J. O., Silverman B. W.** Functional Data Analysis. New York: Springer-Verlag. 1997.

12. **Frey B. J., Jojic N.** Transformation-invariant clustering using the EM algorithm // IEEE Transactions on PAMI. January 2003. V. 25, N 1. P. 1—17.

13. **Gaffney S., Smyth P.** Trajectory clustering with mixtures of regression models / Chaudhuri S., Madigan D., eds. // Proceedings of Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. August 15—18 1999. New York: ACM Press. 1999. P. 63—72.

УДК 621.398

**А. С. Мохов,** аспирант, **В. О. Толчеев,** д-р техн. наук, проф., e-mail: tolcheevvo@mail.ru
Национальный исследовательский университет "МЭИ", г. Москва

# Разработка методов высокоточной классификации двуязычных текстовых библиографических документов

*Рассматривается задача повышения точности классификации двуязычных текстовых документов. Исследуются известные процедуры классификации и разрабатываются специальные подходы на основе профильных методов. Приводятся результаты экспериментов на сформированных двуязычных выборках. Эти результаты свидетельствуют, что предлагаемые модификации профильных методов более полно учитывают специфику двуязычных документов и позволяют увеличить точность их классификации. Наилучшая точность достигается при объединении профильных методов в коллектив решающих правил.*

***Ключевые слова:*** *обработка и анализ двуязычных текстовых документов, методы классификации, профильные методы, коллективы решающих правил, ошибка классификации*

**A. S. Mokhov, V. O. Tolcheev**

# The Development of High-Precision Classification Methods for Bilingual Text Documents

*The main problem of the article is an increase of an accuracy of bilingual text classification. We develop and research profile methods and compare them with "classic" methods. Our results confirm that profile methods permit to increase an accuracy of bilingual text classification. Combining profile-classifiers in an ensemble we get the best results.*

***Keywords:*** *text Mining, methods of bilingual text classification, profile methods, ensemble of classifiers, accuracy (and error) of classification*

## Введение

В настоящее время в рамках работ по совершенствованию методов обработки естественного языка (*Natural Language Processing*) и информационного поиска (*Information Retrieval*) формируется новое направление по созданию процедур классификации многоязычных текстовых данных (*Multilingual Text Categorization*). На практике чаще всего проводится классификация двуязычной информации. При этом рассматриваются две постановки задачи:

1. Имеется обучающая выборка, содержащая документы на *исходном языке*, требуется разработать классификатор для правильного определения классов документов другого *целевого языка* (например, в качестве исходного языка может использоваться английский, для которого легко составить обучающую выборку практически для любой предметной области, а целевого — вьетнамский, венгерский, польский и т. п.). Такая постановка задачи в специализированной литературе называется *перекрестной классификацией разноязычных текстовых документов* (*Cross-Language Text Categorization*). Для ее решения разрабатываются процедуры на основе специализированных онтологий и двуязычных словарей [1—3]. При проведении перекрестной классификации используется перевод текстов с одного языка на другой, поэтому результирующая точность существенным образом зависит от качества используемых двуязычных словарей.

2. Имеется *смешанная* обучающая выборка, которая состоит из документов, представленных одновременно на двух языках (т. е. одна и та же информация изложена, например, на русском и английском языках), необходимо построить классификатор для разнесения таких документов по группам.

Первая постановка задачи обычно используется при отсутствии на целевом языке достаточно большого числа рубрицированных документов (текстов с указанием метки класса — рубрики) для формирования обучающих и экзаменационных выборок. Такая ситуация возникает, например, при зарождении новых научных направлений или при анализе узкоспециализированных предметных областей, для которых тематических публикаций мало и они представлены в основном на английском языке.

При второй постановке задачи предполагается, что имеются смешанные выборки — обучающие и экзаменационные массивы на двух языках. Эти массивы могут содержать информацию различного типа: политическую, научную, юридическую, рекламную и т. п. В данной работе рассматривается проблема

классификации научных двуязычных документов, заданных своими *библиографическими описаниями* (название публикации, аннотация и ключевые слова). При этом смешанные выборки формировались из библиографических описаний (одних и тех же статей) на русском и английском языках.

В настоящее время для классификации двуязычных смешанных выборок в основном используются хорошо известные в литературе "классические" методы (наивный байесовский метод (НБ), метод к-ближайших соседей (к-БС), метод центроидов (МЦ)) [4—6]. Однако "классические" методы разрабатывались применительно к анализу одноязычных текстов, поэтому они не всегда могут в полной мере учитывать информацию, которая содержится в документе, написанном на нескольких языках. В связи с этим актуальной представляется задача повышения точности классификации двуязычных выборок за счет разработки новых (или модернизации имеющихся) процедур.

На наш взгляд, к числу наиболее эффективных подходов, способных увеличить точность классификации двуязычных документов, относятся:

- разработка *комплексного решающего правила*, в котором используются закономерности, выявленные в ходе обучения на смешанной выборке (или при раздельном обучении на каждой из двух выборок);
- построение *коллективного решающего правила* (т. е. синтез коллектива решающих правил — КРП) [7—9]. Для членов КРП обучение также может проводиться как по смешанной выборке, так и раздельно на документах каждого из представленных в выборке языков. В случае использования простого голосования документу присваивается та метка класса, за которую проголосовало большинство членов КРП.

## Математическая модель представления двуязычных документов и методы классификации

Для анализа документальной информации выборка из текстовых документов описывается в виде *расширенной матрицы "документ — термин"*, строки которой представляют собой документы, а столбцы — (русскоязычные и англоязычные) термины, содержащиеся в этих документах [6, 10, 11]:

$$X = \begin{array}{c} \begin{matrix} \textit{Русские термины} & & & \textit{Английские термины} \\ & p \neq M/2 & & \end{matrix} \\ \left[\begin{array}{ccc:ccc} x_1^{(1)} & \dots & x_1^{(p)} & x_1^{(p+1)} & \dots & x_1^{(M)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_N^{(1)} & \dots & x_N^{(p)} & x_N^{(p+1)} & \dots & x_N^{(M)} \end{array}\right] \end{array}, \quad (1)$$

где $x_i^{(j)}$ — вес термина $i$ в документе $j$ ($i = 1, ..., M$; $j = 1, ..., N$); $M$ — общее число терминов в смешанной выборке; $N$ — число документов; $p$ — число русских терминов; $(M - p)$ — число английских терминов, в общем случае $p \neq M/2$.

Вес терминов в вышеприведенной матрице рассчитывается по известным в литературе формулам взвешивания, например, по формуле *tf—idf*-взвешивания [6]:

$$x_j^{(i)} = f_{ij} \log\left(\frac{N}{N_i}\right). \quad (2)$$

Здесь $f_{ij}$ — частота слова $i$ в документе $j$; $N_i$ — общее число документов выборки, содержащих слово $i$.

В данной работе исследуется ряд известных методов классификации применительно к анализу двуязычных выборок и анализируются вновь предложенные подходы для повышения точности классификации. Рассмотрим эти процедуры более подробно.

***Метод ближайшего соседа.*** Согласно правилу ближайшего соседа новый документ, представленный в виде вектора $X_{N+1}$, относится к тому же классу $Q_k$ ($k = 1,...,K$), к которому принадлежит его ближайший сосед $X_j^*$ ($j$-я строка матрицы "документ-термин") [4]:

$$d(X_j^*, X_{N+1}) = \min d(X_j, X_{N+1}) \ \forall j = 1,...,N. \quad (3)$$

Здесь $d$ — мера близости. В данном исследовании использовались две меры близости:

- евклидово расстояние

$$d(X_j, X_l) = \sqrt{\sum_{i=1}^{M} (x_j^{(i)} - x_l^{(i)})^2}; \quad (4)$$

- косинусоидальная мера близости (косинус угла между векторами) [6]

$$d(X_j, X_l) = \cos(X_j, X_l) = \frac{\left|\sum_{i=1}^{M} x_j^{(i)} x_l^{(i)}\right|}{\sqrt{\sum_{i=1}^{M} (x_j^{(i)})^2 \sum_{i=1}^{M} (x_j^{(i)})^2}}. \quad (5)$$

На практике чаще применяется метод к-БС, в котором решение о классификации принимается на основе анализа меток классов, приписанных к-БС. Новый документ относится к классу, за который проголосовало большинство соседей.

***Наивный байесовский классификатор.*** В наивном байесовском классификаторе используется вероятностная модель определения класса документов.

В *вероятностной* модели принадлежность документа $X$ к классу $Q_k$ определяется как вероятность принадлежности каждого из терминов $x^{(i)}$ классу $Q_k$ [5, 11]:

$$P(Q_k|X) = P(Q_k) \prod_{i=1}^{M} P(x^{(i)}|Q_k). \quad (6)$$

Согласно вероятностной модели документ принадлежит тому классу, для которого величина $P(Q_k|X)$ максимальна. В формуле (6) использованы следующие обозначения:

$P(Q_k|X)$ — вероятность того, что документ $X$ принадлежит классу $Q_k$;

$P(Q_k)$ — вероятность встретить документ класса $Q_k$ во всей выборке;

$P(x^{(i)}|Q_k)$ — вероятность принадлежности термина $x^{(i)}$ классу $Q_k$.

***Метод центроидов.*** В данном методе для каждого класса рассчитываются центроиды $\mathbf{C}_k$ — вектора со средними значениями весов терминов документов данного класса:

$$\mathbf{C}_k = \frac{1}{N_k} \sum_{j=1}^{N_k} X_j, \qquad (7)$$

где $N_k$ — число документов, принадлежащих классу $Q_k$. Для классификации нового документа $X_{N+1}$ определяется расстояние, например по формуле (4), между ним и центроидами всех классов; $X_{N+1}$ относится к классу с наиболее близким центроидом.

***Профильные методы.*** Рассмотренный выше метод центроидов относится к так называемым профильным методам, в которых рассчитывается некоторый формальный объект — профиль, способный характеризовать все остальные элементы класса при классификации новых документов (в методе центроидов профиль состоит из средневзвешенных значений терминов) [6, 12].

Однако центроидный профиль не всегда обеспечивает высокие значения для наиболее информативных классообразующих терминов. Улучшение точности метода центроидов может быть достигнуто за счет включения в профиль с большими весами самых информативных терминов, для определения которых необходимо применять специальные процедуры.

Рассмотрим более подробно способы выявления информативных терминов на основе *таблиц сопряженности размера* 2 × 2 (табл. 1).

В табл. 1 применяются следующие обозначения: $A$ — число раз, когда термин $x^{(i)}$ и класс $Q_k$ встречаются вместе; $B$ — число раз, когда $x^{(i)}$ встречается без $Q_k$; $C$ — число раз, когда $Q_k$ встречается без $x^{(i)}$; $D$ — число раз, когда ни $Q_k$, ни $x^{(i)}$ не встречаются; $A + B + C + D = N$ — общее число документов в выборке.

Таблица 1

**Таблица сопряженности**

| Признак $x^{(i)}$ | Принадлежность классу $Q_k$ | Непринадлежность классу $Q_k$ |
|---|---|---|
| Имеется | $A$ | $B$ |
| Отсутствует | $C$ | $D$ |

В наших исследованиях применяли три способа выявления информативных терминов, из которых затем создавался соответствующий профиль: *статистический* (на основе $\chi^2$-критерия), *теоретико-информационный* (на основе критерия взаимной информации), *эвристический* (на основе расчета коэффициентов ассоциативности). Профили вычисляют по следующим формулам [13]:

ФИ-профиль:

$$\Phi(x^{(i)}, Q_k) = \frac{\chi^2}{N} = \frac{(AD - BC)^2}{(A+B)(C+D)(A+C)(B+D)}; \qquad (8)$$

РО-профиль:

$$\rho(x^{(i)}, Q_k) = \sqrt{\frac{\chi^2}{N}} = \frac{(AD - CB)}{\sqrt{(A+B)(C+D)(A+C)(B+D)}}; \qquad (9)$$

МИ-профиль:

$$MI^r(x^{(i)}, Q_k) = \log^2\left(\frac{A^r N}{(A+B)(A+C)}\right). \qquad (10)$$

Профиль Соукала—Сниса (С—С):

$$SS(x^{(i)}, Q_k) = \frac{2(A+D)}{2(A+D)+B+C}. \qquad (11)$$

ФИ-, РО-, МИ- и С—С-профили включают термины, которые имеют наибольшие значения, вычисленные по табл. 1 и вышеуказанным формулам. Настраиваемыми параметрами профильных методов являются: длина профиля $L$ (обычно $L < M$), а также порядок $r$ для МИ-профиля.

Веса терминов в МИ-профиле изменяются в произвольном диапазоне. Вместе с тем, результаты классификации лучше интерпретируемы, если веса варьируются в известных пределах, например, от нуля (между термином и классом нет зависимости) до 1 (имеется полная зависимость). В работе для расчетов весов предлагается использовать нормированный МИ-профиль.

*Нормированный МИ-профиль (НМИ):*

$$NMI(x^{(i)}, Q_k) = \frac{A\log\frac{AN}{(A+B)(A+C)}}{(A+B)\log\frac{N}{A+B}}. \qquad (12)$$

На этапе классификации рассчитываются значения весов классов [13]:

$$\omega_k = \sum_{i=1}^{M_k} tf_i \cdot \mathrm{Pro}f(x^{(i)}, Q_k), \qquad (13)$$

где $tf_i$ — частота встречаемости $i$-го слова в классифицируемом документе $X_{N+1}$; $M_k$ — число наиболее информативных терминов, включенных в профиль $k$-го класса (в наших исследованиях все классы имели профиль одинакового размера $L = M_k$);

$\Pr of(x^{(i)}, Q_k)$ — означает профиль, вычисленный по одной из формул (8)—(12).

Решающее правило в профильных методах имеет вид: классифицируемый документ $X_{N+1}$ относится к тому классу ($X_{N+1} \in Q_k$), которому соответствует наибольший вес $\omega_k = \max (k = 1, ..., K)$, т. е. в $X_{N+1}$ наиболее часто встречаются термины, которые входят в профиль $k$-го класса.

### Сравнительный анализ методов классификации

Для проведения исследований были сформированы 21 обучающая и экзаменационная выборки, состоящие из 7 классов каждая (7 — русскоязычные выборки, 7 — англоязычные и 7 — смешанные). Обучающие и экзаменационные выборки между собой не пересекаются и состоят соответственно из 385 ($N = 385$) и 84 ($n = 84$) библиографических документов. Для формирования документальных массивов использовались публикации по различным тематикам информатики (*Computer Science*), полученные по фиксированным запросам (ключевым словам) из электронной библиотеки *eLibrary.ru*. На этапе предварительной обработки данных были исключены стоп-термины — слова, которые не несут смысловой нагрузки, и для обоих языков был проведен стемминг — выделение корня слова.

Сравнительный анализ "классических" и профильных процедур на сформированных выборках показал, что профильные методы обладают более высокой точностью. На рисунке приводятся ошибки профильных методов для длин профилей $L = 200$ на английской ("e"), русской ("r") и смешанной ("a") выборках, а также ошибки "классических" методов — метода к-БС и метода центроидов (МЦ) на тех же выборках (размер словаря $M$ — 650 терминов для каждого языка, число ближайших соседей $k = 5$, косинусоидальная мера близости). На диаграммах "ящик с усами", показаны медианы, квартили и разброс результатов по 7 (русскоязычным, англоязычным и смешанным) выборкам для исследуемых методов (см. рисунок). Значение длины профиля ($L = 200$) выбрано экспериментально, дальнейшее увеличение $L$ не приводит к заметному уменьшению ошибки классификации [14].

В ходе экспериментальных исследований были отмечены следующие особенности методов:

- в РО-профиле происходит завышение веса высокочастотных слов (данное свойство характерно для всех профилей, основанных на $\chi^2$-критерии);
- в НМИ-профиле, в противоположность РО-, большие веса получают редкие термины (этим свойством обладают профили, базирующиеся на критерии взаимной информации);
- в эвристических профилях, рассчитываемых на основе коэффициентов ассоциативности, наблюдается слабая вариативность весов терминов, наилучшую точность обеспечивает С—С-профиль, который присваивает самые высокие значения весов по сравнению с другими коэффициентами ассоциативности.

Таблица 2

**Средние ошибки профильных методов, метода к-БС и метода центроидов на русскоязычной, англоязычной и смешанной выборках**

| Выборка | РО-профиль | НМИ-профиль | С-С-профиль | к-БС | МЦ |
|---|---|---|---|---|---|
| Английская | 22,63 | 19,73 | 23,14 | 21,60 | 30,76 |
| Русская | 17,51 | 12,93 | 14,80 | 17,53 | 25,34 |
| Смешанная | 14,63 | 11,21 | 13,09 | 14,97 | 23,49 |

Из результатов, приведенных на рисунке и в табл. 2, следует, что профильные методы обладают более высокой точностью по сравнению с "классическими" процедурами. Дополнительные преимущества профильных процедур связаны с достаточно высоким быстродействием, которое достигается за счет компактного представления профилей классов (используется 200 наиболее информативных терминов) по сравнению с "классическими" процедурами, для которых требуется словарь, состоящий из 650 слов.

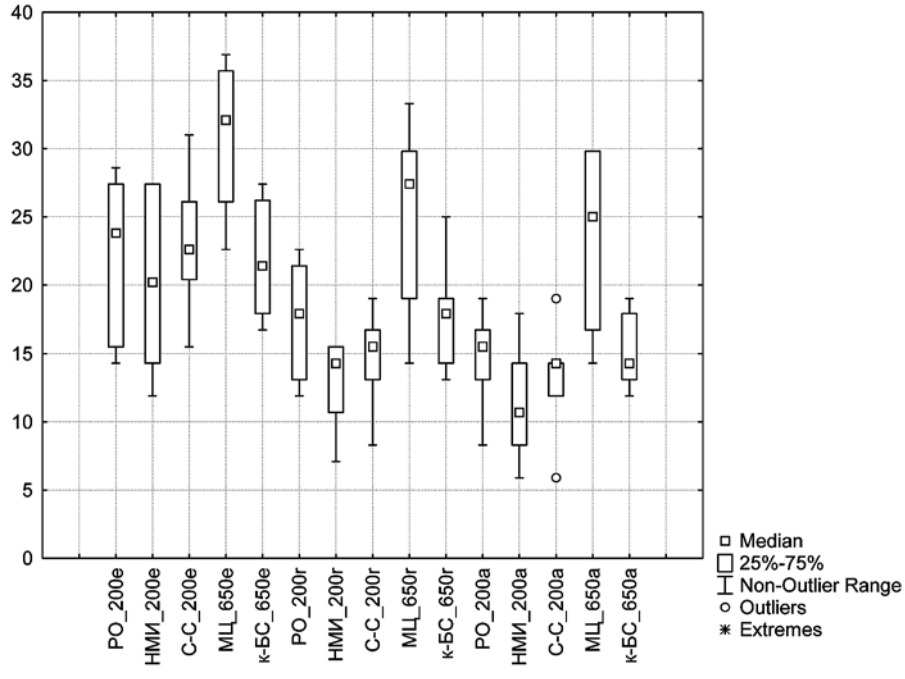


**Диаграмма "Ящик с усами" для ошибок классификации различными методами (диаграмма выполнена в ППП STATISTICA)**

К числу наиболее важных результатов, полученных при экспериментальном исследовании профильных методов, следует отнести увеличение точности на большинстве смешанных выборок по сравнению с русско- и англоязычными текстовыми массивами. Это дает возможность утверждать, что при правильном выборе метода использование двуязычных текстовых массивов обеспечивает более высокую точность, чем классификация тех же документов, представленных только на одном языке.

Важной особенностью профильных процедур является их "гибкость" при настройке, простота модификации за счет объединения в новом профиле терминов, информативность которых определена с помощью различных формул (8)—(12). Задача состоит в том, чтобы во вновь формируемых профилях учесть сильные стороны каждого из способов выявления классообразующих признаков и скомпенсировать их недостатки.

## Исследование новых профильных методов и коллективов решающих правил для классификации двуязычной текстовой информации

В работе для разработки новых профилей взяты три исходных метода (РО-, НМИ- и С—С-процедуры), основанные на различных принципах определения информативности терминов. Были сформированы и экспериментально проверены несколько комбинаций новых профилей UNI1, UNI2 и UNI5 [14].

1. В методе UNI1 исследовалась целесообразность построения смешанного (русско-английского) профиля, в который включались самые информативные (имеющие наибольший вес) термины обоих языков, рассчитанные по формулам РО- и НМИ-профилей ($L = 200$). Таким образом, в UNI1 должны попасть наиболее частотные термины, отобранные по формуле РО-профиля, и специфические достаточно редкие слова, отражающие терминологические особенности предметной области (они будут выявлены с помощью НМИ-профиля).

2. В основу метода UNI2 было положено предположение, что русскоязычные и англоязычные тексты неравнозначны. Так как русский язык является "родным" для авторов статей, изложение материала на нем более квалифицированное и информативное, чем на английском (у авторов уровень знания иностранной терминологии ниже и, как следствие, изложение темы менее качественное). В профиль UNI2 включались $h$ классообразующих терминов из русскоязычных РО- и НМИ-профилей, дополненных $t$ наиболее информативными англоязычными словами из соответствующих РО- и НМИ-профилей. Длина профиля $L = h + t$ (в наших исследованиях $h = 100$, $t = 100$ терминов).

3. UNI5 — элементы профиля рассчитываются как сумма весов теоретико-информационного и эвристического подходов (НМИ- и С—С-профилей). За счет высоких значений С—С-профиля результирующие веса информативных терминов существенно возрастают (становятся больше 1) и усиливается их влияние на определение класса нового документа.

Таблица 3

**Средние ошибки классификации профильных методов**

| Метод классификации | Ошибки классификации, % |
|---|---|
| РО | 14,97 |
| НМИ | 11,21 |
| С—С | 13,09 |
| UNI1 | 13,79 |
| UNI2 | 12,41 |
| UNI5 | 12,59 |

Ошибки классификации на смешанных выборках для РО-, НМИ-, UNI1-, UNI2-, UNI5-профилей при $L = 200$ приведены в табл. 3.

Эксперименты на выборках показали, что модифицированные профили (семейство процедур UNI) не позволяют уменьшить ошибку по отношению к наиболее точному НМИ-профилю. Вместе с тем, они демонстрируют лучшую (или сопоставимую) точность в сравнении с РО- и С—С-профилями.

Важным результатом проведенных исследований является то, что получена группа практически равноточных методов, способных обучаться на русскоязычных, англоязычных и смешанных выборках, максимально используя всю имеющуюся в исходных данных информацию. Причем большинство методов разнородно, т. е. основано на различных способах построения профиля класса. Это позволяет рассматривать их в качестве кандидатов для построения коллективов решающих правил (КРП). При объединении в коллектив можно ожидать, что разнородные профильные процедуры будут "исправлять" ошибки друг друга и увеличивать результирующую точность классификации.

В данной работе рассматриваются КРП, которые включают несколько (обычно не меньше трех) равноточных, но разнородных классификаторов, объединенных для выработки общего решения. Принятие решений в КРП во многом аналогично процедуре согласования мнений нескольких специалистов в экспертных системах. К числу существенных преимуществ КРП относят, прежде всего, возможность увеличения точности группировки документов по классам в сравнении с использованием индивидуального классификатора. КРП также обладает хорошей интерпретируемостью результатов и высокой устойчивостью решений (незначительные изменения в выборках несущественно влияют на получаемые оценки) [7—9, 14].

Нами были сформированы и экспериментально проверены три КРП, состоящие из разного числа

членов (в скобках указаны входящие в состав классификаторы):

1. В КРП1 (РО, НМИ, С—С) были включены три наиболее разнородных классификатора: статистический РО-профиль, теоретико-информационный нормированный МИ-профиль и эвристический С—С-профиль. Обучение проводилось на смешанных выборках, длина профиля $L = 200$.

2. КРП2 (РО, НМИ, С—С, UNI2, UNI5) — представляет собой КРП1, расширенный за счет включения UNI2- и UNI5- профилей. Обучение проводилось на смешанных выборках, длина профиля $L = 200$.

3. КРП3 (РО, НМИ, С—С, метод центроидов, к-БС) — представляет собой КРП1, расширенный "классическими" методами: методом центроидов и методом к-БС. Обучение проводилось на смешанных выборках, для профильных методов длина профиля $L = 200$, размер русско-английского словаря для "классических" процедур $M = 1300$, использовалась косинусоидальная мера близости и пять ближайших соседей в методе к-БС.

В табл. 4 представлены средние ошибки сформированных КРП на семи выборках и средняя ошибка наиболее точного индивидуального метода — НМИ-профиля.

Таблица 4

**Средние ошибки классификации на смешанных выборках**

| Классификатор | Ошибка классификации, % |
|---|---|
| НМИ-профиль | 11,21 |
| КРП1 (РО, НМИ, С—С) | 11,04 |
| КРП2 (РО, НМИ, С—С, UNI2, UNI5) | 9,84 |
| КРП3 (РО, НМИ, С—С, Центроид, к-БС) | 10,36 |

Все три сформированных КРП на двуязычных документальных массивах обеспечивают прирост в точности по сравнению с наиболее точным членом коллектива — НМИ-профилем. Как показали экспериментальные исследования (см. табл. 4), для заметного уменьшения ошибки в КРП необходимо включать не менее пяти разнородных и равноточных методов.

## Заключение

Задача повышения точности является ключевой в теории классификации. В данной работе эта задача рассматривается в контексте обработки и анализа двуязычной информации. Наши экспериментальные исследования показали, что использование смешанных выборок, которые содержат терминологическую информацию на русском и английском языках, в большинстве случаев обеспечивает более высокую точность классификации по сравнению с одноязычными выборками.

На основе экспериментальных результатов можно сделать вывод о хороших точностых характеристиках профильных методов. Эти методы, в том числе и предложенные в данной статье, за счет более эффективного выявления информативных терминов позволяют улучшить точность классификации на смешанных (двуязычных) выборках по сравнению с известными "классическими" методами.

Приблизительно одинаковые ошибки всех профильных методов при их существенной разнородности позволяют объединять эти процедуры в КРП — коллективный классификатор, обладающий наиболее высокой точностью классификации двуязычных документов.

### Список литературы


1. **Bel N., Koster C. H., Villegas M.** Cross-Lingual Text Categorization // Proceedings of the 7th European Conference on Research and Advanced Technology for Digital Libraries. 2003. P. 126—139.

2. **Yejun Wu and Douglas W.** Oard Bilingual Aspect Classification Based on Cross-language Text Classification // Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Singapore. 2008. P. 203—210.

3. **Prettenhofer P., Stein B.** Cross-Language Text Categorization Using Structural Correspondence Learning // Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics. 2010. P. 1118—1127.

4. **Дуда Р., Харт П.** Распознавание образов и анализ сцен. М.: Мир, 1976. 511 с.

5. **Барсегян А. А., Куприянов М. С., Степаненко В. В., Холод И. И.** Методы и модели анализа данных: Data Mining. СПб.: БХВ—Петербург, 2004. 336 с.

6. **Сэлтон Г.** Автоматическая обработка, хранение и поиск информации. М.: Советское радио, 1973. 560 с.

7. **Растригин Л. А., Эренштейн Р. Х.** Метод коллективного распознавания. М.: Энергоиздат, 1981. 79 с.

8. **Толчеев В. О.** Синтез коллективов решающих правил для проведения классификации текстовых документов // Информационные технологии. 2007. № 10. С. 32—38.

9. **Kuncheva L. I., Whitaker C. J., Shipp C. A., Duin R. P.** Limits on the Majority Vote Accuracy in Classifier Fusion // Pattern Analysis and Applications, 2003. V. 6. P. 22—31.

10. **Толчеев В.О.** Модели и методы классификации текстовой информации // Информационные технологии. 2004. № 5. С. 6—14.

11. **Aas K., Eikvil L.** Text Categorization: A Survey. Oslo: Norwegian Computing Center. 1999. P. 1—37.

12. **Толчеев В. О.** Методы выявления закономерностей из эмпирических данных. М.: Изд-во МЭИ, 2010. 88 с.

13. **Толчеев В. О.** Основы теории классификации многомерных наблюдений. М.: Изд-во МЭИ, 2012. 122 с.

14. **Мохов А. С., Толчеев В. О.** Разработка профильных методов классификации двуязычных текстовых документов // Матер. 6-й Всероссийской мультиконференции по проблемам управления, МКПУ-2013. Дивноморское: Изд-во ЮФУ. Т. 1. С. 75—79.

УДК 62-50:519.7/8

**В. И. Левин,** д-р техн. наук, проф.,
Пензенский государственный технологический университет, e-mail: vilevin@mail.ru

# Методология оптимизации в условиях неопределенности методом детерминизации

*Рассмотрены существующие подходы к оптимизации и оптимальному проектированию систем в условиях неопределенности. Дана точная постановка задачи условной оптимизации при интервальной неопределенности параметров целевой функции и ограничений. Изложена математическая теория сравнения интервалов, включающая точное определение максимального и минимального интервалов. На ее основе сформулирован и обоснован метод детерминизации, позволяющий решить поставленную задачу путем сведения к двум полностью определенным задачам условной оптимизации того же типа.*

***Ключевые слова:*** *оптимизация, неопределенность, оптимизация при интервальной неопределенности, метод детерминизации*

**V. I. Levin**

# The Methodology of Optimization in Condition of Uncertainty by Determination Method

*Existing approaches to system optimization (optimal design) under uncertainty are considered. The exact formulation of constrained optimization problem with interval uncertainty of the objective function and constraints parameters is given. The mathematical theory of intervals comparison including precise definition of minimum and maximum ranges is stated. On the base of this theory the determination method is formulated and justified which allows to solve problem by reduction to a two fully certain constrained optimization problems of same type.*

***Keywords:*** *optimization, uncertainty, optimization with interval uncertainty, determination method*

## Введение

Задачи оптимизации имеют большое прикладное значение: на их основе построены методы оптимального проектирования различных систем — технических, экономических, социальных и т. д., обеспечивающие достижение наилучшего, в определенном смысле, результата работы создаваемой системы. В связи с этим к настоящему времени создано огромное число методов решения задач оптимизации как универсальных, рассчитанных на применение к задачам различных классов, так и специализированных, позволяющих эффективно решать лишь узкие классы задач [1—6]. Однако при всем различии существующих методов все они имеют одно общее свойство — применимость только к тем задачам оптимизации, в которых оптимизируемая функция известна точно (детерминирована). Между тем встречающиеся на практике задачи оптимизации и оптимального проектирования обычно таковы, что их оптимизируемые функции известны не точно, а с той или иной степенью неопределенности (недетерминированы). Это вызвано следующими фактами: 1) большинству реальных процессов свойственна естественная неопределенность; 2) параметры большинства систем ввиду погрешности вычислений и измерений известны неточно; 3) многие параметры систем изменяются во времени.

В связи с этим возникает проблема оптимизации неполностью определенных (недетерминированных) функций. Эта проблема достаточно сложна по сравнению с традиционной оптимизацией полностью определенных функций, поскольку для нее дополнительно необходимо:

- обобщить понятие экстремума функции;
- выяснить условия существования экстремума, связанные с неполной определенностью (недетерминированностью) функции;
- разработать специальные методы поиска экстремума функций.

Реально эта проблема еще сложнее, поскольку имеющаяся информация об оптимизируемой функции может быть не только неполностью определенной, но и неоднозначной, неточной, противоречивой и т. д. В такой ситуации многие авторы считают, что модели для описания сложных систем могут быть смысловыми, носящими содержательно-описательный, словесный характер. Такой взгляд представляется не вполне логичным. Действительно, математика, как хорошо известно, строится как полностью определенная, однозначная, точная и непротиворечивая наука. Поэтому правильное применение математики к описанию сложных систем — неопределенных, противоречивых, неточных, неоднозначных и т. д. — вполне способно давать адекватные математические модели этих систем, лишенные неопределенности, неоднозначности, неточности и противоречивости. Для этого требуется всего лишь подобрать математический аппарат, который позволяет оперировать с неопределенностью и другими НЕ-свойствами исследуемой сложной системы так же точно и однозначно, как классическая математика оперирует с полностью определенными системами.

Существуют различные подходы к нахождению оптимума неполностью определенных (недетерминированных) функций, различающиеся достоинствами и недостатками [7]. Первый подход — детерминированный — состоит в решении задач оптимизации для определенных значений или сочетаний значений параметров оптимизируемой функции, взятых внутри заданных областей их неопределенности [7]. Например, можно взять наихудшее сочетание значений параметров внутри областей неопределенности (пессимистический подход) [7—10], наилучшее сочетание (оптимистический подход) [11], центры (середины) областей неопределенности параметров (центральный подход) [12] и др. Основное достоинство этого подхода — простота интерпретации полученного решения, основной недостаток — слабомотивированная ориентировка на какое-то одно значение (сочетание значений) параметров, которое на практике реализуется редко, что может обернуться неоправданной сложностью решения. Второй подход — вероятностный — состоит в решении задачи оптимизации для усредненных (ожидаемых, в смысле математического ожидания) значений параметров оптимизируемой функции или для таких значений параметров, которые обеспечивают достаточно высокую вероятность получения оптимума [13—16]. Этот подход предполагает задание вероятностных распределений указанных параметров внутри областей их неопределенности. Основное достоинство этого подхода — ориентировка получаемого решения хотя и на одно, но зато наиболее часто встречающееся (наиболее подходящее для получения оптимума) значение (сочетание значений) параметров функции, основной недостаток — необходимость знания вероятностных распределений параметров, что зачастую бывает невозможно. Третий подход — нечеткий — идейно близок второму, но вместо вероятностных распределений параметров оптимизируемой неполностью определенной функции, являющихся объективными характеристиками значений этих параметров, используют нечеткие распределения параметров, получаемые экспертным путем, т. е. субъективно [12].

В работах автора [17—24] был предложен и детально описан применительно к различным оптимизационным задачам детерминизационный подход к нахождению оптимума неполностью определенных функций. Данный подход принципиально отличается от трех предыдущих тем, что оптимизация неполностью определенной функции ведется с учетом всего множества возможных значений недетерминированных параметров функции.

Указанный подход позволяет для любой функции, неопределенность которой заключается в том, что ее параметры известны лишь с точностью до интервалов возможных значений, свести нахождение оптимума такой функции к нахождению одноименных оптимумов двух полностью определенных функций. Таким образом, для нахождения оптимума неполностью определенных функций становится возможно применять многочисленные хорошо известные и эффективно работающие методы точного нахождения оптимума полностью определенных (детерминированных) функций. При этом собственно сам алгоритм нахождения оптимума неполностью определенной функции оказывается полностью определенным, точным, однозначным и непротиворечивым. Другой причиной выбора неопределенности именно интервального типа было то, что интервальные оценки неизвестных параметров систем наиболее просты и доступны для получения. В этом и заключаются основные достоинства предложенного нами подхода к оптимизации неполностью определенных функций — метода детерминизации. У этого метода есть и другие достоинства и недостатки. Они подробно рассмотрены в разд. 4.

В настоящей работе детерминизационный подход к оптимизации неполностью определенных функций излагается и обосновывается в наиболее общем виде, не зависящем от особенностей оптимизируемых функций.

## 1. Постановка задачи

Рассмотрим следующую ситуацию. Пусть задана некоторая произвольная непрерывная функция $n$ переменных

$$y = F(x_1, \ldots, x_n), \tag{1}$$

причем все параметры (коэффициенты) ее явного представления известны точно, и она существует в области, определяемой ограничениями

$$\Phi_i(x_1, \ldots, x_n) \leqslant b_i,\ i = \overline{1,m}. \tag{2}$$

Тогда для функции (1) можно сформулировать полностью определенную задачу условной оптимизации:

$$F(x_1, ..., x_n) = \max,$$
$$\text{при } \Phi_i(x_1, ..., x_n) \leqslant b_i,\ i = \overline{1, m}. \qquad (3)$$

Конечно, возможен и вариант задачи (3), где необходимо не максимизировать, а минимизировать функцию $F$. В современном математическом программировании разработано множество различных методов эффективного решения задач (3), ориентирующихся на тип функций $F$ и $\Phi_i$, $i = \overline{1, m}$.

Пусть теперь параметры $p_k$, $k = \overline{1, l}$ явного представления функции $F$ известны не точно, а с точностью до интервалов своих значений, т. е. имеют вид интервалов $\tilde{p}_k = [p_{k1}, p_{k2}]$. Пусть аналогичным образом неточно заданы параметры $q_s$ явного представления функций $\Phi_i$ в левых частях ограничений и параметры $b_i$ в правых частях, т. е. $\tilde{q}_{si} = [q_{si1}, q_{si2}]$, $s = \overline{1, t}$, $\tilde{b}_i = [b_{i1}, b_{i2}]$, $i = \overline{1, m}$. Тогда функции $F$ и $\Phi_i$, $i = \overline{1, m}$, становятся интервальными (т. е. принимающими вид интервалов $\tilde{F}$ и $\tilde{\Phi}_i$, $i = \overline{1, m}$), определяемыми с точностью до интервалов возможных значений, равно как и параметры $\tilde{b}_i$, $i = \overline{1, m}$ (т. е. принимающие вид интервалов $\tilde{b}_i$, $i = \overline{1, m}$). В итоге полностью определенная задача условной оптимизации (3) переходит в неполностью определенную — интервальную задачу условной оптимизации

$$\tilde{F}(x_1, ..., x_n) = \max,$$
$$\text{при } \tilde{\Phi}_i(x_1, ..., x_n) \leqslant \tilde{b}_i,\ i = \overline{1, m}. \qquad (4)$$

Конечно же, возможен вариант задачи (4), где требуется не максимизировать, а минимизировать функцию $\tilde{F}$. Итак, необходимо разработать методику решения оптимизационной задачи вида (4).

## 2. Математика сравнения интервалов

В основе решения поставленной выше интервальной задачи условной оптимизации (4) лежит математическая теория сравнения интервалов.

Рассмотрим два интервала $\tilde{a} = [a_1, a_2]$ и $\tilde{b} = [b_1, b_2]$. Попытаемся сравнить их по величине, рассматривая как интервальные числа. Первое, что приходит в голову, — сравнить интервалы $\tilde{a}$ и $\tilde{b}$ на основе отношений в отдельных парах вещественных чисел $(a_i, b_j)$, где $a_i \in \tilde{a}$, $b_j \in \tilde{b}$. Но такой подход сразу ведет к провалу, поскольку в общем случае, при произвольных интервалах $\tilde{a}$ и $\tilde{b}$, некоторые пары интервалов $(a_i, b_j)$ будут находиться в отношении $a_i > b_j$, а другие — в противоположном отношении $a_i < b_j$. По этой причине единственное, что нам остается, — реализовать сравнение интервалов на теоретико-множественном уровне, рассматривая их как единое целое, которое не подлежит дроблению на более мелкие части. Этот путь был реализован автором статьи в 1990-е годы. Ниже приводится краткое изложение полученных результатов [25—28].

Операции взятия максимума $\vee$ и минимума $\wedge$ двух интервалов $\tilde{a} = [a_1, a_2]$ и $\tilde{b} = [b_1, b_2]$ введем в виде теоретико-множественных конструкций

$$\tilde{a} \vee \tilde{b} = \{a \vee b | a \in \tilde{a},\ b \in \tilde{b}\},$$
$$\tilde{a} \wedge \tilde{b} = \{a \wedge b | a \in \tilde{a},\ b \in \tilde{b}\}. \qquad (5)$$

Согласно (5), взятие максимума (минимума) интервалов $\tilde{a}$ и $\tilde{b}$ есть нахождение максимума (минимума) двух точечных величин $a$ и $b$ при условии, что конкретные значения величин пробегают все возможные значения из интервалов $\tilde{a}$ и $\tilde{b}$ соответственно. Для того чтобы интервалы $\tilde{a}$ и $\tilde{b}$ можно было сравнить по величине, установив их отношение — $\tilde{a} \geqslant \tilde{b}$ или $\tilde{a} \leqslant \tilde{b}$, необходимо, во-первых, чтобы введенные операции $\vee$, $\wedge$ над этими интервалами существовали, во-вторых, чтобы эти операции давали своим результатом один из операндов — $\tilde{a}$ или $\tilde{b}$, и в-третьих, чтобы эти операции являлись согласованными в том смысле, что если большим (меньшим) является один из интервалов, то меньшим (большим) является другой. Сформулированное условие сравнимости интервалов по величине является, очевидно, не только необходимым, но и достаточным.

К счастью, легко доказать, что условие согласованности $\vee$ и $\wedge$ над интервалами выполняется для любой пары интервалов $(\tilde{a}, \tilde{b})$. Очевидно также, что всегда выполняется условие существования введенных операций взятия максимума $\vee$ и минимума $\wedge$ двух интервалов, причем результатом операции оказывается некоторый, вообще говоря, новый интервал. Таким образом, необходимым и достаточным условием сравнимости $\tilde{a}$ и $\tilde{b}$ оказывается условие, по которому операции $\tilde{a} \vee \tilde{b}$ и $\tilde{a} \wedge \tilde{b}$ должны иметь своим результатом один из интервалов — $\tilde{a}$ или $\tilde{b}$. Последняя формулировка условия сравнимости интервалов открывает возможность получения его в конструктивной форме, пригодной для практического применения. Основной результат здесь формулируется следующим образом.

**Теорема 1.** Для того чтобы два интервала $\tilde{a} = [a_1, a_2]$ и $\tilde{b} = [b_1, b_2]$ являлись сравнимыми по величине (по отношению $\geqslant$) и находились при этом в отношении $\tilde{a} \geqslant \tilde{b}$, необходимо и достаточно, чтобы границы этих интервалов подчинялись следующим условиям:

$$a_1 \geqslant b_1,\ a_2 \geqslant b_2, \qquad (6)$$

а для того чтобы они были сравнимы по величине (отношению $\leqslant$) и находились в отношении $\tilde{a} \leqslant \tilde{b}$, необходимо и достаточно выполнения условий

$$a_1 \leqslant b_1,\ a_2 \leqslant b_2. \qquad (7)$$

Эта теорема показывает, что интервалы $\tilde{a}$ и $\tilde{b}$ являются сравнимыми по отношению $\geqslant$ или $\leqslant$ (и находятся именно в этом отношении) только в том

случае, когда в таком же отношении находятся одноименные границы этих интервалов $a_1$, $b_1$ и $a_2$, $b_2$. Иными словами, два интервала $\tilde{a}$ и $\tilde{b}$ находятся в отношении $\tilde{a} \geqslant \tilde{b}$ только тогда, когда $\tilde{a}$ сдвинут обеими своими границами вправо относительно $\tilde{b}$ и в отношении $\tilde{a} \leqslant \tilde{b}$ только тогда, когда интервал $\tilde{a}$ сдвинут обеими границами влево относительно $\tilde{b}$.

Значение теоремы 1 заключается в том, что она сводит сравнение двух интервалов и выбор большего (меньшего) из них к сравнению их одноименных границ, являющихся обычными числами. Так разрешается проблема сравнения интервалов.

**Теорема 2.** Для того чтобы два интервала $\tilde{a} = [a_1, a_2]$ и $\tilde{b} = [b_1, b_2]$ были не сравнимы по величине (по отношению $\geqslant$ и $\leqslant$), т. е. не находились в отношении $\tilde{a} \geqslant \tilde{b}$ или $\tilde{a} \leqslant \tilde{b}$, необходимо и достаточно выполнения условий

$$(a_1 < b_1, a_2 > b_2) \text{ или } (b_1 < a_1, b_2 > a_2). \qquad (8)$$

Эта теорема показывает, что интервалы $\tilde{a}$ и $\tilde{b}$ не сравнимы по отношению $\geqslant$ и $\leqslant$ лишь тогда, когда один из них полностью "накрывает" другой.

Результат теоремы 2 состоит в том, что она показывает существование случаев несравнимости интервалов по отношениям $\geqslant$ и $\leqslant$, в отличие от вещественных чисел, которые всегда сравнимы по указанным отношениям. Несравнимость величин некоторых интервалов есть естественный результат того факта, что интервальные числа, в отличие от обыкновенных вещественных чисел, задаются не точно, а с неопределенностью (например, известно, что число принимает некоторое значение в заданном интервале, но не уточняется, какое именно это значение). На основе теорем 1 и 2 можно доказать нижеследующие положения.

**Теорема 3.** Для того чтобы существовал максимальный интервал в некоторой системе интервалов $\tilde{a}(1) = [a_1(1), a_2(1)]$, $\tilde{a}(2) = [a_1(2), a_2(2)]$, ... (который находится со всеми остальными интервалами в отношении $\geqslant$), необходимо и достаточно, чтобы его границы относительно одноименных границ остальных интервалов были расположены согласно следующим условиям:

$$\left.\begin{array}{l} a_1(1) \geqslant a_1(2),\ a_1(1) \geqslant a_1(3),\ \ldots \\ a_2(1) \geqslant a_2(2),\ a_2(1) \geqslant a_2(3),\ \ldots \end{array}\right\}. \qquad (9)$$

Здесь необходимо сделать уточнение, что условия (9) даны конкретно для того случая, когда максимальным является интервал $\tilde{a}(1)$. Однако хорошо видно, что эта конкретизация не ограничивает общности.

**Теорема 4.** Для того чтобы существовал минимальный интервал в некоторой системе интервалов $\tilde{a}(1) = [a_1(1), a_2(1)]$, $\tilde{a}(2) = [a_1(2), a_2(2)]$ ... (который находится со всеми остальными интервалами в отношении $\leqslant$), необходимо и достаточно, чтобы его границы относительно одноименных границ остальных интервалов были расположены согласно следующим условиям:

$$\left.\begin{array}{l} a_1(1) \leqslant a_1(2),\ a_1(1) \leqslant a_1(3),\ \ldots \\ a_2(1) \leqslant a_2(2),\ a_2(1) \leqslant a_2(3),\ \ldots \end{array}\right\}. \qquad (10)$$

Аналогично теореме 3 условия (10) записаны для случая, когда минимальным является интервал $\tilde{a}(1)$, что не ограничивает общности.

Теоремы 3 и 4 показывают, что интервал является максимальным (минимальным) среди множества имеющихся интервалов только в том случае, когда максимальны (минимальны) его нижняя граница — среди нижних границ всех интервалов — и его верхняя граница — среди верхних границ всех интервалов.

### 3. Идея решения

В задаче (4) целевая функция $\tilde{F}(x_1, ..., x_n)$, функции $\tilde{\Phi}_i(x_1, ..., x_n)$, $i = \overline{1, m}$, в левых частях ограничений и параметры $\tilde{b}_i$, $i = \overline{1, m}$, в правых частях являются интервальными и поэтому могут быть записаны в виде интервалов

$$\begin{array}{l} \tilde{F}(x_1, ..., x_n) = [F_1(x_1, ..., x_n),\ F_2(x_1, ..., x_n)]; \\ \tilde{\Phi}_i(x_1, ..., x_n) = [\Phi_{i1}(x_1, ..., x_n), \Phi_{i2}(x_1, ..., x_n)],\ i = \overline{1, m}; \\ \tilde{b}_i = [b_{i1}, b_{i2}],\ i = \overline{1, m}. \end{array} \qquad (11)$$

После этого интервальную задачу условной оптимизации (4) можно перезаписать в явном интервальном виде:

$$\begin{array}{l} [F_1(x_1, ..., x_n),\ F_2(x_1, ..., x_n)] = \max; \\ [\Phi_{i1}(x_1, ..., x_n), \Phi_{i2}(x_1, ..., x_n)] \leqslant [b_{i1}, b_{i2}],\ i = \overline{1, m}, \end{array} \qquad (12)$$

который уже поддается решению с помощью каких-либо известных методов оптимизации. Действительно, согласно теореме 3 интервальное уравнение в верхней строке сформированной выше системы (12) можно записать в виде эквивалентной пары обычных (детерминированных) уравнений

$$F_1(x_1, ..., x_n) = \max,\ F_2(x_1, ..., x_n) = \max. \qquad (13)$$

Далее, согласно теореме 1, систему интервальных неравенств в задаче условной оптимизации (12) можно записать в виде эквивалентной системы обычных (детерминированных) неравенств:

$$\Phi_{i1}(x_1, ..., x_n) \leqslant b_{i1},\ \Phi_{i2}(x_1, ..., x_n) \leqslant b_{i2},\ i = \overline{1, m}. \qquad (14)$$

Закончим построение следующим очевидным образом. Рассматривая совместно пару детерминированных уравнений оптимизации (13) с системой неравенств-ограничений (14), имеем две детерминированные задачи условной оптимизации вида (3), при этом задачу (15) естественно назвать нижней

граничной задачей исходной интервальной задачи (4), а задачу (16) — ее верхней граничной задачей:

$$\left.\begin{aligned} &F_1(x_1, ..., x_n) = \max, \\ &\Phi_{i1}(x_1, ..., x_n) \leqslant b_{i1},\ i = \overline{1, m}, \\ &\Phi_{i2}(x_1, ..., x_n) \leqslant b_{i2},\ i = \overline{1, m}, \end{aligned}\right\} \quad (15)$$

$$\left.\begin{aligned} &F_2(x_1, ..., x_n) = \max, \\ &\Phi_{i1}(x_1, ..., x_n) \leqslant b_{i1},\ i = \overline{1, m}, \\ &\Phi_{i2}(x_1, ..., x_n) \leqslant b_{i2},\ i = \overline{1, m}. \end{aligned}\right\} \quad (16)$$

Из построения следует, что пара задач (15), (16), рассматриваемых в совокупности, эквивалентна исходной интервальной задаче (4). Таким образом, для получения решения задачи (4) надо решить ее нижнюю (15) и верхнюю (16) граничные задачи. В общем случае решения нижней и верхней задач: $\{M_н(x), F_{1,\max}\}$, $\{M_в(x), F_{1,\max}\}$, где $M_н(x)$, $M_в(x)$ — множества точек решений $x = (x_1, ..., x_n)$ нижней и верхней граничной задач; $F_{1,\max}$, $F_{2,\max}$ — полученные максимальные значения целевых функций этих задач. Решение задачи (4) составляется из решений ее нижней и верхней граничных задач в виде

$$\{x^* \in M_н(x) \cap M_в(x),\ \widetilde{F}_{\max} = [F_{1,\max}, F_{2,\max}]\}. \quad (17)$$

В качестве точки решения $x^*$ в (17) берется любая точка из пересечения множеств $M_н(x)$, $M_в(x)$, а в качестве максимального значения целевой функции $\widetilde{F}_{\max}$ — интервал от максимума целевой функции нижней граничной задачи $F_{1,\max}$ до максимума целевой функции верхней граничной задачи $F_{2,\max}$.

Очевидно, что преимущество нашего подхода к решению интервальной задачи условной оптимизации заключается в возможности применения традиционных, хорошо разработанных методов решения детерминированных задач оптимизации. Основанный на этом подходе метод решения назовем методом детерминизации, поскольку он сводит решение недетерминированной задачи вида (4) к решению двух детерминированных задач (15) и (16).

### 4. Алгоритм решения

Для решения интервальной задачи (4) методом детерминизации необходимо действовать по следующему алгоритму.

*Шаг 1.* Используя формулы интервальной математики, выражающие результаты элементарных преобразований интервалов [18],

$$\begin{aligned} &[a_1, a_2] + [b_1, b_2] = [a_1 + b_1, a_2 + b_2]; \\ &[a_1, a_2] - [b_1, b_2] = [a_1 - b_1, a_2 - b_2]; \\ &k[a_1, a_2] = \begin{cases} [ka_1, ka_2], & k > 0, \\ [ka_2, ka_1], & k < 0; \end{cases} \\ &[a_1, a_2] \cdot [b_1, b_2] = [\min_{i,j}(a_i \cdot b_j), \max_{i,j}(a_i \cdot b_j)]; \\ &[a_1, a_2]/[b_1, b_2] = [a_1, a_2] \cdot [1/b_2, 1/b_1], \end{aligned} \quad (18)$$

представляем целевую функцию $\widetilde{F}$ и функции ограничений $\widetilde{\Phi}_i$ задачи (4) в интервальной форме. Так же представляем параметры $\widetilde{b}_i$ в ограничениях задачи. Полученные представления имеют вид (11).

*Шаг 2.* Используя интервальные представления целевой функции задачи, функции ограничений, а также параметров, полученные на шаге 1, формируем нижнюю граничную (15) и верхнюю граничную (16) задачи интервальной задачи условной оптимизации (4).

*Шаг 3.* Используя подходящие методы решения детерминированных задач условной оптимизации, получаем решения нижней $\{M_н(x), F_{1,\max}\}$ и верхней $\{M_в(x), F_{2,\max}\}$ граничных задач. При этом $M_н(x)$ — множество точек решения $x = (x_1, ..., x_n)$ нижней граничной задачи, на котором ее целевая функция $F_1$ достигает своего максимума $F_{1,\max}$, и аналогично, $M_в(x)$ — множество точек решения $x = (x_1, ..., x_n)$ верхней граничной задачи, в которых ее целевая функция $F_2$ достигает максимума $F_{2,\max}$.

*Шаг 4.* Выбирая в качестве точки решения интервальной задачи (4) любую точку $x^*$ из пересечения множеств $M_н(x)$ и $M_в(x)$ точек решения нижней и верхней граничных задач и беря в качестве нижней границы максимума $\widetilde{F}_{\max}$ интервальной целевой функции $\widetilde{F}$ задачи (4) максимум $\widetilde{F}_{1,\max}$ целевой функции нижней граничной задачи, а в качестве верхней границы максимума целевой функции $\widetilde{F}$ задачи (4) соответственно максимум $\widetilde{F}_{2,\max}$ целевой функции верхней граничной задачи, получаем полное решение интервальной задачи (4) в виде (17).

**Пример** (интервальный вариант задачи о назначениях). Пусть в организации имеются три работы и три исполнителя — кандидата на выполнение этих работ. Заданы издержки $\tilde{a}_{ij} = [a_{1,ij}, a_{2,ij}]$ выполнения любой $j$-й работы любым $i$-м исполнителем ($i, j = \overline{1, 3}$), представляющие собой интервалы и составляющие матрицу $\tilde{A} = \|\tilde{a}_{ij}\| = \|[a_{1,ij}, a_{2,ij}]\| = [A_1, A_2]$, где $A_1 = \|a_{1,ij}\|$ и $A_2 = \|a_{2,ij}\|$ — нижняя и верхняя граничные матрицы издержек. Нужно распределить работы между исполнителями таким образом, чтобы каждый из них был занят выполнением ровно одной работы, а суммарные издержки были минимальны.

В целях моделирования задачи введем множество неизвестных булевых матриц назначений $X = \|x_{ij}\|$, $x_{ij} \in \{0,1\}$, где $x_{ij} = 1$, если $i$-й исполнитель выполняет $j$-ю работу, и $x_{ij} = 0$ в противном случае. Тогда изложенная задача записывается математически таким образом:

$$\widetilde{F}(x_{ij}) \equiv \sum_{i=1}^{3}\sum_{j=1}^{3} \tilde{a}_{ij}x_{ij} = \min, \text{ при}$$

$$\Phi_1(x_{ij}) \equiv \sum_{i=1}^{3} x_{ij} = 1,\ j = \overline{1, 3};\ \Phi_2(x_{ij}) \equiv \sum_{j=1}^{3} x_{ij} = 1,\ i = \overline{1, 3}.$$

Очевидно, что выписанная выше задача представляет собой частный случай общей неполностью определенной (интервальной) задачи условной оп-

тимизации (4). Поэтому для решения данной задачи вполне можно применить четырехшаговый алгоритм, описанный выше.

***Шаг 1.*** С помощью формул (18) представляем целевую функцию нашей задачи $\tilde{F}$ в интервальной форме:

$$\tilde{F}(x_{ij}) \equiv \left[ \sum_{i=1}^{3} \sum_{j=1}^{3} a_{1,ij} x_{ij},\ \sum_{i=1}^{3} \sum_{j=1}^{3} a_{2,ij} x_{ij} \right].$$

Представлять в интервальной форме функции $\tilde{\Phi}_1(x_{ij})$, $\tilde{\Phi}_2(x_{ij})$ ограничений задачи и параметры в правых частях ограничений не надо, так как в них не фигурируют интервальные параметры.

***Шаг 2.*** Используя интервальные представления целевой функции, функции ограничений и параметров в правых частях, полученные на шаге 1, формируем нижнюю граничную и верхнюю граничную задачи решаемой интервальной задачи условной оптимизации:

$$F_1(x_{ij}) \equiv \sum_{i=1}^{3} \sum_{j=1}^{3} a_{1,ij} x_{ij} = \min,$$

$$\text{при } \sum_{i=1}^{3} x_{ij} = 1,\ j = \overline{1,3},\ \sum_{j=1}^{3} x_{ij} = 1,\ i = \overline{1,3};$$

$$F_2(x_{ij}) \equiv \sum_{i=1}^{3} \sum_{j=1}^{3} a_{2,ij} x_{ij} = \min,$$

$$\text{при } \sum_{i=1}^{3} x_{ij} = 1,\ j = \overline{1,3},\ \sum_{j=1}^{3} x_{ij} = 1,\ i = \overline{1,3}.$$

**Шаг 3.** Решаем нижнюю граничную и верхнюю граничную задачи, сформированные на предыдущем шаге, принимая следующее конкретное числовое значение интервальной матрицы издержек:

$$\tilde{A} = [A_1, A_2],$$

где $A_1 = \|a_{1,ij}\| = \begin{Vmatrix} 1 & 2 & 2 \\ 1 & 2 & 2 \\ 2 & 2 & 2 \end{Vmatrix}$, $A_2 = \|a_{2,ij}\| = \begin{Vmatrix} 2 & 3 & 3 \\ 4 & 4 & 3 \\ 3 & 4 & 4 \end{Vmatrix}$.

Имеется всего шесть различных значений матриц неизвестных $X = \|x_{ij}\|$, удовлетворяющих ограничениям решаемой задачи. Поэтому решение легко находится перебором на множестве этих матриц. В результате получаем решение нижней граничной задачи в виде $\{M_{\text{н}}(x), F_{1,\min}\}$, где множество решений $M_{\text{н}}$ равно

$$M_{\text{н}}(x) = \left\{ X_{1a} = \begin{Vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{Vmatrix},\ X_{1b} = \begin{Vmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{Vmatrix},\right.$$

$$\left. X_{1c} = \begin{Vmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{Vmatrix},\ X_{1d} = \begin{Vmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{Vmatrix} \right\},$$

а достигнутое минимальное значение целевой функции $F_{1,\min} = 5$. Далее получаем решение верхней граничной задачи $\{M_{\text{в}}(x), F_{1,\min}\}$: множество решений

$$M_{\text{в}}(x) = \left\{ X_{2a} = \begin{Vmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{Vmatrix},\ X_{2b} = \begin{Vmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{Vmatrix} \right\},$$

а соответствующее достигнутое минимальное значение целевой функции верхней граничной задачи составляет $F_{2,\min} = 9$.

***Шаг 4.*** Находим пересечение множеств решений нижней $M_{\text{н}}(x)$ и верхней $M_{\text{в}}(x)$ граничных задач. Оно состоит из одной матрицы назначений

$$x^* = X_{1b} = X_{2a} = \begin{Vmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{Vmatrix},$$

которая есть решение всей задачи. Наконец, находим достигнутое на этом решении минимальное значение заданной интервальной целевой функции, в нашем случае оно представляет собой интервал $\tilde{F}_{\min} = [F_{1,\min}, F_{2,\min}] = [5, 9]$.

Итак, оптимальное решение нашей интервальной задачи таково: назначить 1-го исполнителя на 1-ю работу, 2-го — на 3-ю, 3-го — на 2-ю. При этом необходимые издержки оцениваются минимальным интервалом [5, 9].

Другие примеры решения оптимизационных задач с использованием изложенного алгоритма даны в работах [7—9, 15, 19].

## 5. Сравнение предлагаемого подхода с существующими

Как было сказано во введении, проблема оптимизации неполностью определенных функций, по сравнению с традиционной оптимизацией полностью определенных функций, требует дополнительно:

1) обобщения понятия экстремума функции;

2) выяснения условий существования экстремума функций, связанных с ее неполной определенностью;

3) разработки специальных методов поиска экстремума таких функций.

Именно по этой схеме разработан предлагаемый в статье детерминизационный подход к оптимизации. Конкретно, обобщение понятия экстремума функции на случай неполностью определенных (интервальных) функций дано в разд. 2 (формулы (5)). Далее условия существования (несуществования) экстремума интервальной функции приведены в теоремах 1—4 того же разд. 2. Наконец, специальный метод поиска экстремума интервальной функции разработан в разд. 4. Необходимость проведения этой работы очевидна. Действительно, оптимизация полностью определенных функций основана на сравнении вещественных чисел с выделением большего и меньшего, причем на вещественной оси большее число сдвинуто вправо относительно

меньшего. Однако для оптимизации неполностью определенных функций такой подход не работает, так как неполностью определенные числа (например интервальные), в отличие от вещественных, в общем случае не находятся в отношении "сдвинуто вправо (влево) на вещественной оси" и потому не могут сравниваться непосредственно, с выделением большего и меньшего числа. Вследствие этого для таких функций и приходится обобщать понятие экстремума.

Далее неполнота информации, которой характеризуются неполностью определенные числа и функции, при достижении определенного уровня может привести к ситуации несравнимости таких чисел и невозможности выделить из них большее и меньшее и, как следствие, — к отсутствию экстремума таких функций. В связи с этим и возникает необходимость нахождения условий существования экстремума неполностью определенных функций.

Наконец, вследствие иного, более общего, чем для полностью определенных функций, понятия экстремума неполностью определенной функции и возможности несуществования этого экстремума, вызванной неполнотой информации, приходится разрабатывать специальные методы отыскания экстремума таких функций. Важно понимать, что невозможность в определенных случаях с помощью предложенного в статье алгоритма нахождения экстремума неполностью определенной функции не связана с качеством самого алгоритма, а является следствием объективной реальности, а именно — отсутствия экстремума вследствие недостатка информации об анализируемой функции.

Охарактеризуем теперь другие существующие подходы к оптимизации неполностью определенных функций. Коротко об их основных достоинствах и недостатках сказано во введении. Рассмотрим вопрос подробнее. Начнем с детерминированного подхода. Здесь задача оптимизации неполностью определенной функции (исходная) заменяется другой задачей — оптимизации полностью определенной функции. Причем конструирование этой новой задачи путем выбора определенных значений параметров внутри областей неопределенности параметров функции исходной задачи зачастую выполняется на основе чисто эвристических соображений и не опирается ни на какие математически ясные обобщения понятия экстремума на случай неполностью определенных функций. Вследствие этого новая задача оказывается, как правило, математически неэквивалентной исходной, а интерпретация ее решения в терминах исходной задачи — проблематичной. Кроме того, ввиду большой сложности некоторых критериев оптимизации, используемых в новой задаче (maxmin, minmax), трудоемкость алгоритмов поиска экстремума неполностью определенных функций при детерминированном подходе может оказаться высокой. Зато при этом подходе обычно не возникает проблемы выяснения условий существования экстремума функций, так как полностью определенные функции практически всегда имеют экстремум.

Теперь о вероятностном подходе.

При первом варианте данного подхода исходная задача оптимизации неполностью определенной функции заменяется, как и в случае детерминированного подхода, другой задачей, а именно, оптимизации полностью определенной функции, которая теперь получается из исходной функции путем замены ее случайных параметров их математическими ожиданиями (центрами). Сразу ясно, что эта новая задача неэквивалентна исходной в еще большей степени, чем при детерминированном подходе, поскольку она, не опираясь ни на какие обобщения понятия экстремума для неполностью определенных функций, не учитывает не только неопределенность возможных значений параметров анализируемой функции, но также и случайный характер реализации конкретных значений параметров на практике.

При втором варианте подхода задача оптимизации неполностью определенной функции с интервальными параметрами фактически заменяется задачей оптимизации неполностью определенной функции со случайными параметрами. Последние получаются из интервальных параметров исходной задачи принятием, например, гипотезы о равномерном распределении значений параметров внутри своих интервалов. Принятие указанной гипотезы сразу упрощает выбор экстремального интервала. Например, для выбора большего из двух интервалов $\tilde{a}$ и $\tilde{b}$ достаточно лишь вычислить вероятности $P(\tilde{a} > \tilde{b})$ и $P(\tilde{b} > \tilde{a})$ и взять тот из интервалов, для которого вероятность превышения им второго интервала больше. Данный подход гарантирует существование решения, полученного с помощью гипотезы модельной задачи оптимизации функции со случайными параметрами. Но проблема заключается в том, что модельная задача неэквивалентна исходной задаче оптимизации функции с интервальными параметрами, так как одно лишь задание неопределенности функции в форме интервалов возможных значений ее параметров не предполагает задания дополнительной информации, например, о вероятностных распределениях внутри интервалов. Все, что было сказано о вероятностном подходе, можно повторить и для нечеткого, с заменой термина "вероятностное распределение параметров неполностью определенной функции" термином "нечеткое распределение" этих параметров.

На практике для решения конкретных задач оптимизации неполностью определенных функций, в зависимости от условий, можно применять различные подходы. В общем случае рекомендуется начинать с детерминизационного подхода, поскольку он базируется на точном определении понятия максимума и минимума неопределенной величины (интервала), что упрощает интерпретацию полученного решения и делает его более прозрачным. Если детерминизационный подход не приводит к решению, вследствие недостаточной информации об оптимизируемой функции, целесообразно эту информацию пополнить путем сужения интервалов

возможных значений параметров этой функции с помощью дополнительных измерений, наблюдений, привлечения более квалифицированных экспертов, после чего снова применить данный подход. Если и это не помогло получить решение, рекомендуется использовать другие подходы. В первую очередь, целесообразно попытаться применить вероятностный подход, который достаточно прост в реализации. При этом нужно иметь в виду, что используемые при этом подходе вероятностные распределения параметров неполностью определенной функции должны быть известны с достаточной точностью, так как в противном случае найденное предположительно оптимальное значение функции может оказаться далеким от настоящего оптимума. Надо еще учитывать, что при вероятностном подходе получение оптимума функции вообще строго не гарантируется, а только "обещается" с определенной вероятностью, притом не обязательно близкой к единице, что не всегда приемлемо. Поэтому на практике часто обращаются к детерминированному подходу в оптимизации неполностью определенных функций. Этот подход, в отличие от детерминизационного, всегда обеспечивает существование оптимума для неполностью определенной функции, и в отличие от вероятностного гарантирует получение этого оптимума. К сожалению, при этом подходе, как говорилось ранее, вследствие преобразования исходной неполностью определенной функции в полностью определенную, новая задача оптимизации оказывается неэквивалентна исходной, а интерпретация ее решения в терминах исходной задачи проблематичной. Например, выбор с помощью детерминированного подхода минимального из двух интервалов $\tilde{a} = [4,5]$, $\tilde{b} = [3,15]$ по критерию оптимальности "нижняя граница интервала минимальна" дает следующее решение: $\min(\tilde{a}, \tilde{b}) = \tilde{b} = [3, 15]$. Но это решение проблематично интерпретировать практически, поскольку оно противоречит эвристическим представлениям. Любой автомобилист уверенно предпочтет, как более экономную, машину с расходом топлива от 4 до 5 л на 100 км машине с расходом от 3 до 15 л!

## Заключение

В настоящей работе показано, что проблема оптимизации неполностью определенных функций достаточно просто разрешима, если неопределенность задавать в интервальной форме и, кроме того, использовать конструктивную теорию сравнения интервальных величин, которая сводит указанное сравнение к сравнению одноименных границ этих интервалов. Тем самым поиск оптимума неполностью определенной функции сводится к нахождению одноименного оптимума двух полностью определенных (детерминированных) функций. Наш подход (его естественно назвать детерминизационным) примечателен тем, что позволяет вполне строго свести оптимизацию неполностью определенных функций к хорошо известным и эффективным методам оптимизации полностью определенных функций.

### Список литературы


1. **Юдин Д. Б., Гольдштейн Е. Г.** Задачи и методы линейного программирования. М.: Сов. радио, 1964.
2. **Вентцель Е. С.** Введение в исследование операций. М.: Сов. радио, 1964.
3. **Уайлд Д. Дж.** Методы поиска экстремума. М.: Наука, 1967.
4. **Корбут А. А., Финкельштейн Ю. Ю.** Дискретное программирование. М.: Наука, 1969.
5. **Моисеев Н. Н., Иванилов Ю. П., Столярова Е. М.** Методы оптимизации. М.: Наука, 1978.
6. **Левин В. И.** Структурно-логические методы исследования сложных систем с применением ЭВМ. М.: Наука, 1987.
7. **Первозванский А. А.** Математические модели в управлении производством. М.: Наука, 1975. 616 с.
8. **Libura M.** Integer Programming Problems with Inexact Objective Function // Control and Cybernetics. 1980. V. 9, N 4. P. 189—202.
9. **Тимохин С. Г., Шапкин А. В.** О задачах линейного программирования в условиях неточных данных // Экономика и математические методы. 1981. № 5. С. 955—963.
10. **Рощин В. А., Семенова Н. В., Сергиенко И. В.** Вопросы решения и исследования одного класса задач неточного целочисленного программирования // Кибернетика. 1989. № 2. С. 42—46.
11. **Семенова Н. В.** Решение одной задачи обобщенного целочисленного программирования // Кибернетика. 1984. № 5. С. 25—31.
12. **Вощинин А. П., Сотиров Г. Р.** Оптимизация в условиях неопределенности. М.: Изд-во МЭИ, 1989. 224 с.
13. **Ащепков Л. Т., Давыдов Д. В.** Универсальные решения интервальных задач оптимизации и управления. М.: Наука, 2006. 285 с.
14. **Давыдов Д. В.** Интервальные методы и модели принятия решений в экономике: дис. д-ра экон. наук. Владивосток, 2009.
15. **Островский Г. М., Волин Ю. М.** Технические системы в условиях неопределенности. Анализ гибкости и оптимизация. М.: Бином, 2008.
16. **Островский Г. М., Зиятдинов Н. Н., Лаптева Т. В.** Оптимизация технических систем. М.: Кнорус, 2012.
17. **Левин В. И.** Дискретная оптимизация в условиях интервальной неопределенности // Автоматика и телемеханика. 1992. № 7.
18. **Левин В. И.** Булево линейное программирование с интервальными коэффициентами // Автоматика и телемеханика. 1994. № 7. С. 111—122.
19. **Левин В. И.** Интервальное дискретное программирование // Кибернетика и системный анализ. 1994. № 6. С. 92—103.
20. **Левин В. И.** Оптимизация расписаний в системах с неопределенными временами обработки. I, II // Автоматика и телемеханика. 1995. № 2, 3.
21. **Левин В. И.** Задача трех станков с неопределенными временами обработки // Автоматика и телемеханика. 1996. № 1. С. 109—120.
22. **Левин В. И.** Интервальная модель общей задачи линейного программирования. I, II // Вестник Тамбовского университета. Серия: Естественные и технические науки. 1998. Т. 3, № 4; 1999. Т. 4, № 1.
23. **Левин В. И.** Нелинейная оптимизация в условиях интервальной неопределенности // Кибернетика и системный анализ. 1999. № 2. С. 138—146.
24. **Левин В. И.** Антагонистические игры с интервальными параметрами // Кибернетика и системный анализ. 1999. № 4. С. 149—159.
25. **Левин В. И.** О недетерминистской дискретной оптимизации // Принятие решений в условиях неопределенности: сб. статей. Уфа: Изд-во Уфимского авиационного ин-та, 1990.
26. **Левин В. И.** Математическая теория сравнения интервальных величин и ее применение в задачах измерения // Измерительная техника. 1998. № 5.
27. **Левин В. И.** Математическая теория сравнения интервальных величин и ее применение в задачах измерения, контроля и управления // Измерительная техника. 1998. № 9.
28. **Левин В. И.** Интервальная математика и исследование систем в условиях неопределенности. Пенза: Изд-во Пензенского технол. ин-та, 1998. 95 с.

УДК 004.272.43:004.4

**С. А. Инютин,** д-р техн. наук, проф.,
МАТИ — РГТУ имени К. Э. Циолковского, e-mail: inyutin_int@mail.ru

# Особенности вычисления характеристик модулярной величины

*Исследуются итерационные методы вычисления количественных характеристик отношения порядка для компьютерных модулярных форматов в параллельных реконфигурируемых вычислительных системах*

***Ключевые слова:*** *многопроцессорные реконфигурируемые системы, вычислительный процесс, сложность вычисления, модулярные компьютерные форматы, числовые характеристики.*

**S. A. Inyutin**

# Peculiarity Calculation Characteristics for Computer Modular Value

*Research iteration methods calculation position characteristics for computer modular formats at parallel reconfiguration calculation systems.*

***Keywords:*** *multiprocessor systems, modular calculation process, figure characteristic for computer formats, complexity calculation fault tolerance, cellular algorithm of the routing*

## Введение

Разработка специальных компьютерных арифметик является теоретической базой программного вычислительного инструментария для организации вычислений в больших и сверхбольших компьютерных диапазонах [1, 2].

Представляет интерес класс специализированных арифметик, использующий множество-носитель с модулярными представлениями числовых данных. Это дает возможность распараллеливания вычислительного процесса в SIMD- и SIMP-архитектурах на множество реальных или моделируемых процессорных элементов, выполняющих независимые вычисления по отдельным модулям [3]. В специализированной компьютерной арифметике в производных форматах отображаются векторы с компонентами, являющимися вычетами числовых величин по простым модулям или по степеням простых модулей [4]. Вычеты целесообразно считать разрядами модулярного представления числовой величины.

Под компьютерной модулярной арифметикой будем понимать множество форматов модулярных данных, способ введения арифметического аддитивного и мультипликативного диапазонов, алгоритмы выполнения модульных операций, методы вычисления количественной характеристики отношения порядка для модулярных представлений числовых величин, а также методы и алгоритмы выполнения немодульных операций [5].

Известно, что аддитивные и мультипликативные операции в модулярной компьютерной арифметике (сложение, вычитание, умножение, целочисленное деление) выполняются параллельно и независимо по каждому из модулей — разрядов. При этом не возникает переносов между модульными разрядами, что позволяет выполнять параллельные вычисления в независимых вычислительных трактах [6, 7].

## 1. Количественные характеристики модулярных представлений

При разработке вычислительных средств (технических устройств или комплекса программ) параллельной структуры на основе модулярной арифметики центральной является проблема эффективного вычисления количественных характеристик отношения порядка для модулярных представлений, называемых позиционными характеристиками [8]. Вычисление таких характеристик позволяет определить порядок и арифметический знак на множестве модулярных представлений числовых величин. Сложность выполнения немодульных операций в модулярной арифметике линейно зависит от сложности вычисления этих характеристик.

К позиционным характеристикам предъявляются требования минимальной (линейной, логарифмической) сложности вычисления, универсальности или применимости для выполнения на их основе немодульных операций в модулярной арифметике. Рассмотрим методы уменьшения сложности вычис-

ления позиционных характеристик для одинарного модулярного вычислительного диапазона — $P$ [7, 8].

Для компьютерной одинарной модулярной арифметики первая введенная позиционная характеристика модулярной величины $A(\mathrm{mod}P) \leftrightarrow (\alpha_1 \mathrm{mod} p_1, ..., \alpha_n \mathrm{mod} p_n)$ называлась рангом, затем был введен нормированный ранг $Z_A$ [5, 9].

Система модулярных оснований (простых или взаимно-простых чисел) может быть упорядочена $p_1 < p_2 < ... p_n$, $\forall i, j\ (p_i, p_j) = 1$ и хранится в кэш-памяти процессора [10].

Для модулярной величины выполняется аддитивное разложение или равенство, связывающее значение числовой величины с нормированными компонентами вектора модулярного представления:

$$A = \sum_{i=1}^{n} \alpha_i \frac{P}{p_i} - Z_A P, \qquad (1)$$

где $\tilde{\alpha}_i = |A|_{p_i}$ — компонента классического модулярного представления;

$\alpha_i = \left| \tilde{\alpha}_i \left| \frac{P}{p_i} \right|_{p_i}^{-1} \right|_{p_i}$ — нормированная компонента модулярного представления;

$Z_A = \left[ \sum_{i=1}^{n} \frac{\alpha_i}{p_i} \right]$ — позиционная характеристика — нормированный ранг;

$P = \prod_{i=1}^{n} p_i$ — максимум одинарного модулярного диапазона;

$p_i$ — основания одинарной модулярной системы.

Точное вычисление нормированного ранга требует суммирования рациональных дробей со знаменателем $P$, т. е. вычислений в компьютерном диапазоне с максимальным значением $P$. Значение нормированного ранга — целочисленного функционала от нормированных компонент модулярного представления принадлежит отрезку $[0, n-1]$, причем обычно $n \ll P$, $Z_A \in [0, n-1]$.

Вследствие неравенства областей значений и определения нормированного ранга сложность вычисления позиционной характеристики увеличивается до квадратичной. Для уменьшения сложности вычисления позиционной характеристики до линейной требуется поиск методов вычисления нормированного ранга $Z_A$ модулярной величины с областью определения, близкой к области значений [11].

## 2. Итерационный метод вычисления количественной характеристики

Рассмотрим метод вычисления нормированного ранга модулярной величины. Пусть параметр $D \in N$, $D > 1$, тогда

$$Z_A = \left[ \sum_{i=1}^{n} \frac{\alpha_i}{p_i} \right] = \left[ \frac{1}{D} \left( \sum_{i=1}^{n} \left[ \frac{\alpha_i D}{p_i} \right] + \sum_{i=1}^{n} \frac{|\alpha_i D|_{p_i}}{p_i} \right) \right]. \qquad (2)$$

В качестве приближения нормированного ранга используется целая часть (небольшая) от первого слагаемого выражения (2), полученная позиционная характеристика названа "неточный" ранг [6]:

$$Z_A^I = \left[ \frac{1}{D} \sum_{i=1}^{n} \left[ \frac{\alpha_i D}{p_i} \right] \right]. \qquad (3)$$

Возникающая погрешность в вычислении функционала (2) с использованием выражения (3) определяется вторым слагаемым. Оценим значение этой погрешности.

*Лемма 1.* Верхняя и нижняя оценки погрешности в приближении нормированного ранга модулярной величины задаются неравенствами

$$\frac{\sum_{i=1}^{n} |\alpha_i D|_{p_i}}{p_n} \leqslant \sum_{i=1}^{n} \frac{|\alpha_i D|_{p_i}}{p_i} \leqslant \frac{\sum_{i=1}^{n} |\alpha_i D|_{p_i}}{p_1}.$$

**Доказательство.** Система модулярных оснований упорядоченная, поэтому выполняется двойное неравенство для правильных дробей:

$$\frac{|\alpha_i D|_{p_i}}{p_n} \leqslant \frac{|\alpha_i D|_{p_i}}{p_i} \leqslant \frac{|\alpha_i D|_{p_i}}{p_1}.$$

Суммирование соответствующих рациональных дробей и отдельно их числителей позволяет получить утверждение леммы.

Далее для вычисления нормированного ранга используем уточняющие итерационные соотношения. Для оценки возникающих погрешностей определим пределы для суммы правильных дробей:

$$\frac{1}{p_i} \leqslant \sum_{i=1}^{n} \xi_i = \frac{\sum_{i=1}^{n} |\alpha_i D|_{p_i}}{p_i} \leqslant \frac{p_i - 1}{p_i} < 1.$$

Для анализа процесса итерационного вычисления уточненного нормированного ранга введем понятие критичности модулярной величины.

*Определение.* Модулярная величина $A(\mathrm{mod}P)$ называется критической (обладает свойством критичности), если не равны нормированный ранг модулярной величины и уточненный ранг, вычисляемый при некотором параметре $D < P$.

*Определение.* Областью критичности в диапазоне $[0, P)$ назовем совокупность модулярных величин $A(\mathrm{mod}P)$, являющихся критическими при некотором значении параметра $D < P$.

При параллельном вычислительном процессе на множестве процессорных элементов, вычисления выполняются по модулям — основаниям модулярной системы. Для уменьшения разрядности процессорных элементов, т. е. вычислительного диапазона при вычислении позиционной характеристики в модулярной арифметике, целесообразно использовать единый целочисленный параметр $n - 1 \leqslant D < p_n$, близкий к значению $p_n$.

Рассмотрим варианты приближения снизу нормированного ранга, просуммировав его с нижней

оценкой значения погрешности правильной дроби из *Леммы 1.*

Целую часть от полученной суммы назовем уточненным нормированным рангом (его первой итерацией):

$$Z_A^{II} = \left[\frac{1}{D}\left(\sum_{i=1}^{n}\left[\frac{\alpha_i D}{p_i}\right] + \frac{\sum_{i=1}^{n}|\alpha_i D|_{p_i}}{p_n}\right)\right] =$$

$$= \left[\frac{1}{Dp_n}\left(p_n\sum_{i=1}^{n}\left[\frac{\alpha_i D}{p_i}\right] + \sum_{i=1}^{n}|\alpha_i D|_{p_i}\right)\right].$$

Для вычисления первой итерации уточненного нормированного ранга требуется компьютерный диапазон с максимальным значением $n \cdot p_n$.

Оценим погрешность при вычислении уточненного нормированного ранга.

*Лемма 2.* Погрешность числителя рациональной дроби при вычислении нормированного ранга с использованием в уточняющем соотношении $Z_A^{II}$, будет не больше

$$\sum_{i=1}^{n}\xi_i\left(1 - \frac{p_i}{p_n}\right).$$

**Доказательство.** Действительно, выполняется

$$\sum_{i=1}^{n}|a_i D|_{p_i}/p_i - \sum_{i=1}^{n}|a_i D|_{p_i}/p_n =$$

$$= \sum_{i=1}^{n}|a_i D|_{p_i}/(1/p_i - 1/p_n) = \sum_{i=1}^{n}|a_i D|_{p_i}/p_i(p_n - p_i)/p_n =$$

$$= \sum_{i=1}^{n}\xi_i(p_n - p_i)/p_n \leqslant \sum_{i=1}^{n}(p_n - p_i)p_n.$$

Получим верхнюю оценку погрешности, учитывая, что $\xi_i = |a_i D|_{p_i}/p_i < 1$.

*Следствие 1.* Погрешность при вычислении нормированного ранга с использованием уточняющего соотношения $Z_A^{II}$ не больше $\left(n - 1 - \sum_{i=1}^{n-1} p_i/p_n\right)/D$.

Усилим приближение нормированного ранга, просуммировав его с нижней оценкой значения погрешности из *Леммы 2.* В результате получена вторая итерация вычисления уточненного нормированного ранга:

$$Z_A^{III} =$$

$$= \left[\frac{1}{D}\left(\sum_{i=1}^{n}\left[\frac{\alpha_i D}{p_i}\right] + \frac{\sum_{i=1}^{n}|\alpha_i D|_{p_i}}{p_n} + \frac{\sum_{i=1}^{n}|\alpha_i D|_{p_i}(p_n - p_i)}{p_n p_n}\right)\right] =$$

$$= \left[\frac{1}{Dp_n}\left(p_n\sum_{i=1}^{n}\left[\frac{\alpha_i D}{p_i}\right] + \sum_{i=1}^{n}|\alpha_i D|_{p_i} + \frac{\sum_{i=1}^{n}|\alpha_i D|_{p_i}(p_n - p_i)}{p_n}\right)\right]. \quad (4)$$

Для вычисления уточненного ранга требуется компьютерный диапазон с максимальным значением $np_n^2$.

Оценим сверху погрешность при вычислении второй итерации уточненного нормированного ранга.

*Лемма 3.* Погрешность при вычислении нормированного ранга с использованием соотношения $Z_A^{III}$ не больше

$$\sum_{i=1}^{n}\xi_i\left(1 - \frac{2p_i}{p_n} + \frac{p_i^2}{p_n^2}\right).$$

**Доказательство.** Действительно, выполняется соотношение

$$\left|\frac{\sum_{i=1}^{n}|\alpha_i D|_{p_i}}{p_i} - \frac{\sum_{i=1}^{n}|\alpha_i D|_{p_i}}{p_n} - \frac{\sum_{i=1}^{n}|\alpha_i D|_{p_i}(p_n - p_i)}{p_n^2}\right| =$$

$$= \left|\sum_{i=1}^{n}|\alpha_i D|_{p_i}\left(\frac{1}{p_i} - \frac{1}{p_n} - \frac{(p_n - p_i)}{p_n^2}\right)\right| =$$

$$= \frac{\sum_{i=1}^{n}|\alpha_i D|_{p_i}}{p_i}\left(1 - \frac{2p_i}{p_n} + \frac{p_i^2}{p_n^2}\right).$$

Получим верхнюю оценку погрешности, учитывая, что $\xi_i = |a_i D|_{p_i}/p_i < 1$.

Далее усилим приближение нормированного ранга, просуммировав его с нижней оценкой значения погрешности из *Леммы 3.* Ограничимся тремя итерациями, что позволяет установить необходимые закономерности.

В результате будет получена третья итерация уточненного нормированного ранга:

$$Z_A^{IV} = \left[\frac{1}{D}\left(\sum_{i=1}^{n}\left[\frac{\alpha_i D}{p_i}\right] + \frac{\sum_{i=1}^{n}|\alpha_i D|_{p_i}}{p_n} + \right.\right.$$

$$\left.\left. + \frac{\sum_{i=1}^{n}|\alpha_i D|_{p_i}(p_n - p_i)}{p_n p_n} + \frac{\sum_{i=1}^{n}|\alpha_i D|_{p_i}(p_n - p_i)^2}{p_n^3}\right)\right] =$$

$$= \left[\frac{1}{Dp_n^2}\left(p_n^2\sum_{i=1}^{n}\left[\frac{\alpha_i D}{p_i}\right] + p_n\sum_{i=1}^{n}|\alpha_i D|_{p_i} + \right.\right.$$

$$\left.\left. + \sum_{i=1}^{n}|\alpha_i D|_{p_i}(p_n - p_i) + \frac{\sum_{i=1}^{n}|\alpha_i D|_{p_i}(p_n - p_i)^2}{p_n}\right)\right]. \quad (5)$$

Для вычисления уточненного ранга требуется компьютерный диапазон с максимальными значениями $n \cdot p_n^3$. Оценим сверху погрешность при вычислении третьей итерации уточненного нормированного ранга.

*Лемма 4.* Погрешность при вычислении нормированного ранга с использованием соотношения $Z_A^{IV}$ не больше

$$\sum_{i=1}^{n} \xi_i \left( 1 + \frac{2p_i^2}{p_n^2} - \frac{3p_i}{p_n} + \frac{p_i^3}{p_n^3} \right).$$

**Доказательство.** Действительно выполняются соотношения

$$\left| \frac{\sum_{i=1}^{n} |\alpha_i D|_{p_i}}{p_i} - \frac{\sum_{i=1}^{n} |\alpha_i D|_{p_i}}{p_n} - \frac{\sum_{i=1}^{n} |\alpha_i D|_{p_i}(p_n - p_i)}{p_n^2} - \frac{\sum_{i=1}^{n} |\alpha_i D|_{p_i}(p_n - p_i)^2}{p_n^3} \right| =$$

$$= \left| \sum_{i=1}^{n} |\alpha_i D|_{p_i} \left( \frac{1}{p_i} - \frac{1}{p_n} - \frac{(p_n - p_i)}{p_n^2} - \frac{(p_n - p_i)^2}{p_n^3} \right) \right|.$$

Получим верхнюю оценку погрешности, учитывая, что $\xi_i = |\alpha_i D|_{p_i} / p_i < 1$.

Вычислим третью итерацию, приближающую сверху уточненный нормированный ранг, сохранив для его вычисления компьютерный диапазон с максимальными значениями $n \cdot p_n^3$:

$$\tilde{Z}_A^{IV} = \left[ \frac{1}{D} \left( \sum_{i=1}^{n} \left[ \frac{\alpha_i D}{p_i} \right] + \frac{\sum_{i=1}^{n} |\alpha_i D|_{p_i}}{p_n} + \frac{\sum_{i=1}^{n} |\alpha_i D|_{p_i}(p_n - p_i)}{p_n p_n} + \frac{\sum_{i=1}^{n} |\alpha_i D|_{p_i}(p_n - p_i)^2}{p_1 p_n^2} \right) \right] =$$

$$= \left[ \frac{1}{D p_n^2} \left( p_n^2 \sum_{i=1}^{n} \left[ \frac{\alpha_i D}{p_i} \right] + p_n \sum_{i=1}^{n} |\alpha_i D|_{p_i} + \sum_{i=1}^{n} |\alpha_i D|_{p_i}(p_n - p_i) + \frac{\sum_{i=1}^{n} |\alpha_i D|_{p_i}(p_n - p_i)^2}{p_1} \right) \right].$$

Оценим сверху погрешность при вычислении этого варианта третьей итерации уточненного нормированного ранга модулярной величины.

*Лемма 5.* Погрешность при вычислении нормированного ранга с использованием соотношения $\tilde{Z}_A^{IV}$ не больше

$$\sum_{i=1}^{n} \xi_i \left( 1 - \frac{2p_i}{p_n} + \frac{p_i^2}{p_n^2} - \frac{p_i}{p_1} + \frac{2p_i^2}{p_1 p_n} - \frac{p_i^3}{p_1 p_n^2} \right).$$

**Доказательство.** Действительно, выполняются соотношения

$$\left| \frac{\sum_{i=1}^{n} |\alpha_i D|_{p_i}}{p_i} - \frac{\sum_{i=1}^{n} |\alpha_i D|_{p_i}}{p_n} - \frac{\sum_{i=1}^{n} |\alpha_i D|_{p_i}(p_n - p_i)}{p_n^2} - \frac{\sum_{i=1}^{n} |\alpha_i D|_{p_i}(p_n - p_i)^2}{p_1 p_n^2} \right| =$$

$$= \left| \sum_{i=1}^{n} |\alpha_i D|_{p_i} \left( \frac{1}{p_i} - \frac{1}{p_n} - \frac{(p_n - p_i)}{p_n^2} - \frac{(p_n - p_i)^2}{p_1 p_n^2} \right) \right|.$$

Получим верхнюю оценку погрешности, учитывая, что $\xi_i = |a_i D|_{p_i} / p_i < 1$.

**Теорема 1.** Критерием равенства уточненного и нормированного рангов является выполнение равенства $Z_A^{III} = \tilde{Z}_A^{III}$.

**Доказательство.** Вычислим уточненный нормированный ранг в третьей итерации с использованием оценок погрешности снизу:

$$\tilde{Z}_A^{III} = \left[ \frac{1}{D} \left( \sum_{i=1}^{n} \left[ \frac{\alpha_i D}{p_i} \right] + \frac{\sum_{i=1}^{n} |\alpha_i D|_{p_i}}{p_n} + \frac{\sum_{i=1}^{n} |\alpha_i D|_{p_i}(p_n - p_i)}{p_n p_n} + \frac{\sum_{i=1}^{n} |\alpha_i D|_{p_i}(p_n - p_i)^2}{p_n p_n^2} \right) \right] =$$

$$= \left[ \frac{1}{D p_n^2} \left( p_n^2 \sum_{i=1}^{n} \left[ \frac{\alpha_i D}{p_i} \right] + p_n \sum_{i=1}^{n} |\alpha_i D|_{p_i} + \sum_{i=1}^{n} |\alpha_i D|_{p_i}(p_n - p_i) + \frac{\sum_{i=1}^{n} |\alpha_i D|_{p_i}(p_n - p_i)^2}{p_n} \right) \right].$$

Вычислим уточненный нормированный ранг в третьей итерации с использованием оценок погрешности сверху:

$$\tilde{Z}_A^{III} = \left[\frac{1}{D}\left(\sum_{i=1}^{n}\left[\frac{\alpha_i D}{p_i}\right] + \frac{\sum_{i=1}^{n}\left|\alpha_i D\right|_{p_i}}{p_n} + \right.\right.$$

$$\left.\left. + \frac{\sum_{i=1}^{n}\left|\alpha_i D\right|_{p_i}(p_n - p_i)}{p_n p_n} + \frac{\sum_{i=1}^{n}\left|\alpha_i D\right|_{p_i}(p_n - p_i)^2}{p_1 p_n^2}\right)\right] =$$

$$= \left[\frac{1}{Dp_n^2}\left(p_n^2\sum_{i=1}^{n}\left[\frac{\alpha_i D}{p_i}\right] + p_n\sum_{i=1}^{n}\left|\alpha_i D\right|_{p_i} + \right.\right.$$

$$\left.\left. + \sum_{i=1}^{n}\left|\alpha_i D\right|_{p_i}(p_n - p_i) + \frac{\sum_{i=1}^{n}\left|\alpha_i D\right|_{p_i}(p_n - p_i)^2}{p_1}\right)\right].$$

Равенство третьих итераций уточненного нормированного ранга с использованием верхней и нижней оценок погрешностей свидетельствует об их равенстве нормированному рангу модулярной величины. Следовательно, теорема доказана.

Проанализируем проблему критичности для модулярных величин в диапазоне $[0, P)$. Из введенных определений критичности модулярной величины $A(\mathrm{mod}P) \in [0, P)$ следует:

- критичность модулярной величины $A(\mathrm{mod}P) \in [0, P)$ зависит от значения модулярной величины и от значения параметра $(n-1) \leqslant D \leqslant P$, в частности, при $D = P$ критических модулярных величин в диапазоне $[0, P)$ нет;
- при $n - 1 \leqslant D < P$ критическими являются только некоторые модулярные величины;
- выполняются неравенства $Z^I \leqslant Z_A^{II} \leqslant Z_A^{III} \leqslant Z_A^{IV} \leqslant Z_A \leqslant \tilde{Z}_A^{II}$, следующие из свойств разрывной функции — целая часть (небольшая) числа;
- для уточненных нормированных рангов выполняется один из двух вариантов: $Z_A^I = Z_A^{II} = Z_A^{III} = Z_A^{IV} = Z_A$ или уточненный нормированный ранг меньше нормированного ранга ровно на единицу, в частности $Z_A^I = Z_A - 1$, причем последнее равенство является достаточным признаком критичности модулярной величины.

Интерес представляет исследование наличия в полном вычислительном модулярном диапазоне областей, в которых заведомо отсутствуют критические модулярные величины.

**Теорема 2.** В диапазоне $[P(n-1)/D, P)$ нет критических модулярных величин.

**Доказательство.** Из равенства (1), используя разложение слагаемых под символом суммы на целую и дробную части, получим

$$\frac{A}{P} + Z_A = \frac{1}{D}\Sigma\left[\frac{\alpha_i D}{p_i}\right] + \frac{\Sigma\left|\alpha_i D\right|_{p_i}}{Dp_i},$$

$$\left[\frac{A}{P} - \frac{\Sigma\left|\alpha_i D\right|_{p_i}}{p_i D}\right] = \left[\frac{1}{D}\Sigma\left[\frac{\alpha_i D}{p_i}\right]\right] - Z_A = 0|(-1).$$

Знаком "|" здесь обозначены альтернативные утверждения.

Ноль в правой части равенства будет получен при выполнении условия

$$A \geqslant \left[\frac{P}{D}\Sigma\frac{\left|\alpha_i D\right|_{p_i}}{p_i}\right].$$

Следовательно, теорема доказана.

*Следствие 1.* При значении параметра $D = g(n-1)$ для модулярных величин $A \in [P/g, P]$ свойство критичности отсутствует.

*Следствие 2.* В модулярной системе при одном избыточном основании с минимальным значением $g = 2$ не являются критическими модулярные величины, удовлетворяющие соотношению $A \in [P/g, P)$.

## Заключение

Введение избыточности в модулярное представление уменьшает сложность алгоритмов вычисления позиционных характеристик модулярной величины, но при этом появляется дополнительное условие: использовать модулярные величины только из диапазонов, в которых отсутствует критичность.

Предложенный метод устранения критичности при вычислении нормированного ранга основан на выборе значения параметра $D$. Необходим учет следующих рекомендаций:

- простые (или взаимно-простые) модулярные основания $p_i$ следует выбрать с близкими значениями;
- параметр $D$ необходимо выбрать кратным или равным наибольшему основанию модулярной системы;
- целесообразен выбор следующего вычислительного аддитивного модулярного диапазона $[0,8P)$, где $P$ — максимальное значение основного арифметического аддитивного диапазона. Указанный числовой коэффициент учитывает: инструментальное значение основного аддитивного диапазона, позволяющее устранить использование критических модулярных величин, а также расширение диапазона, необходимое для определе-

ния знака результата аддитивной операции и обнаружения аддитивного переполнения.

Приведенные выше методы позволяют организовать эффективное вычисление позиционной характеристики — нормированного ранга, необходимого для оценки значения модулярной величины и выполнения немодульных операций.

**Список литературы**


1. **Инютин С. А.** Основы модулярной алгоритмики. Ханты-Мансийск: Полиграфист, 2009. 237 с.

2. **Ноден П., Китте К.** Алгебраическая алгоритмика. М.: Мир, 1999. 720 с.

3. **Инютин С. А.** Модулярные вычисления в сверхбольших компьютерных диапазонах // Известия вузов. Электроника. 2001. № 6. С. 65—73.

4. **Барский А. Б., Желенков Б. В., Шилов В. В.** Реализация вычислительной системы SPMD-архитектуры: схемотехнические решения и параллельные алгоритмы // Зарубежная радиоэлектроника. Успехи современной радиоэлектроники. 2002. № 7. С. 62—70.

5. **Акушский И. Я., Юдицкий Д. И.** Машинная арифметика в остаточных классах. М.: Советское радио, 1968. 440 с.

6. **Амербаев В. М.** Теоретические основы машинной арифметики. Алма-Ата: Наука, 1976. 20 с.

7. **Инютин С. А.** Компьютерная модулярная алгебра одинарного диапазона и область ее приложения // Вестник Тюменского госуниверситета. 2001. № 2. С. 56—65.

8. **Inyutin S. A.** Parallel Square Modular Computer Algebra // Transaction of Parallel Processing and Applied Mathematics PPAM-2003. Poland-Denmark: Springer, 2003. P. 117—123.

9. **Инютин С. А.** Итерационные процессы в классах вычетов по модулю // Сборник научных трудов. Вып. 33. Сургут: ИЦ СурГУ, 2011. С. 13—19.

10. **Шилов В. В., Столярский Е. З.** Организация и работа кэш-памяти // Информационные технологии. 2000. № 7. С. 2—8.

11. **Инютин С. А.** Система оценки сложности "трудных" алгоритмов // Сборник научных трудов. Вып. 32. Сургут: ИЦ СурГУ, 2010. С. 61—67.


---


УДК 51-74



**А. В. Вишнеков,** д-р техн. наук, проф., зав. каф.,
**Е. М. Иванова,** канд. техн. наук, доц., зам. зав. каф., e-mail: theposte@mail.ru,
**В. А. Филиппов,** канд. техн. наук, ст. науч. сотр., проф.,
Московский институт электроники и математики
Национального исследовательского университета "Высшая школа экономики"


# Выбор среды передачи данных при проектировании локальных вычислительных сетей


*Одна из наиболее часто решаемых сегодня задач в области информационных технологий — построение локальных вычислительных сетей. В статье рассматриваются основные этапы разработки проекта локальной вычислительной сети как отдельные многокритериальные задачи выбора частного решения каждого этапа. Предлагается подход как для выбора базового варианта проекта, так и для частного решения, основанный на методах теории принятия решений. Излагается решение частной задачи на основе метода аналитических иерархий и метода ELECTRE. Рассмотрены два примера применения этих методов для выбора типа среды передачи данных по критериям пользователя.*



***Ключевые слова:*** *проект, локальные вычислительные сети, принятие решений*



**A. V. Vishnekov, E. M. Ivanova, V. A. Filippov**


# The Data Transfer Environment Choice in the Local Area Networks Design


*The local networks development is one of the most frequently solved today tasks in the field of information technologies. The article considers the main stages of the local computer network development as a number of separate multicriteria partial solutions of selection tasks. We propose a new approach for the base project variant selection and for private solutions, based on the methods of the decision-making theory. The authors detail the decision of these problems on the basis of the analytical hierarchy method and the ELECTRE method. The article describes two usage examples of these methods for the solution of a problem of types of communication lines selecting on user's criteria.*



***Keywords:*** *project, computer networks, decision making*

## Введение

Задача разработки проекта локальной вычислительной сети (ЛВС) является сложной многоэтапной, многовариантной задачей, включающей формирование логической и физической структуры ЛВС. Качество решения этой задачи всегда оценивается по множеству критериев. Если разбить весь проект на отдельные этапы проектирования и представить каждый этап как отдельную частную многокритериальную задачу, то для автоматизации выбора рационального частного проектного решения можно использовать методы теории принятия решений, давшие позитивный результат в ряде смежных областей, таких как разработка программного и аппаратного обеспечения вычислительной системы [1].

Задача построения ЛВС может служить примером иерархической задачи проектирования, для которой возможно применение *технологии нисходящего принятия решений* [2]. Согласно данной технологии выбирается общая базовая концепция решения, которая может быть получена с помощью групповых методов принятия решений и методов принятия решений в условиях неопределенности исходной информации [3]. Затем общее решение последовательно детализируется, и частные решения принимаются экспертами с помощью индивидуальных методов поддержки принятия решений.

Покажем, как можно разбить проектирование ЛВС на отдельные частные задачи по принятию решений. С точки зрения разнообразных методов принятия решений важно наличие нескольких альтернатив выбора и некоторого перечня критериев выбора. Далее в зависимости от числа альтернатив и критериев, типа и способов описания критериев, от необходимости привлечения группы экспертов или возможности принятия решений самим разработчиком, от необходимости ранжирования альтернатив или возможности принятия единственного решения, а также от некоторых других особенностей решаемой задачи следует предложить и обосновать конкретный метод принятия решения.

## 1. Постановка задачи

Задача выбора рациональных решений при построении ЛВС решается в двух основных направлениях:

- ***формирование логической структуры*** ЛВС, определяющей ее функциональность (способы передачи, обработки, хранения и защиты данных) и способы реализации требуемых динамических характеристик (время и вероятность доведения информации до получателя);
- ***формирование физической структуры*** ЛВС, выполняемое чаще методами комплексирования из типовых элементов (частей), каждый из которых может быть выбран по определенным правилам из большого многообразия, предлагаемого сегодня на рынке, опять же в соответствии с определенными требованиями, предъявляемыми к ЛВС в целом и ее элементам.

Формирование рациональной логической структуры ЛВС в отношении способов получения требуемых динамических характеристик ЛВС опирается на достаточно широко известный и разработанный аппарат графоаналитических, вероятностных и имитационно-статистических методов. Важнейшей и наиболее общей группой критериев выбора рациональных решений при построении логической структуры ЛВС является *полнота ее функциональности*, к которым следует отнести:

- обеспечение совместного доступа, передачи и защиты данных;
- обеспечение совместного использования приложений;
- обеспечение заданных способов взаимодействия пользователей сети друг с другом;
- обеспечение совместного использования периферийных устройств.

Далее этот список может расширяться в соответствии с требованиями заказчиков сетей и тенденциями развития ЛВС по предоставлению новых сервисных услуг для пользователей.

К другой группе немаловажных критериев следует отнести *соответствие требованиям вероятностно-временных характеристик* ЛВС. Не всегда, но достаточно часто в качестве критерия может рассматриваться и предельно возможная *стоимость принятых решений*.

Что касается формирования рациональной физической структуры ЛВС, то здесь пока не просматриваются относительно четкие проработанные подходы, дающие достаточно строгую количественную оценку принятых решений. Основная причина, видимо, кроется в том, что применяемые методы комплексирования носят качественный, часто интуитивный характер, не позволяют учесть противоречивый компромиссный характер показателей элементов, образующих ЛВС. Задача формирования рациональной физической структуры ЛВС может быть разбита на отдельные частные подзадачи:

- выбор абонентского оборудования (рабочих станций);
- выбор серверного оборудования;
- выбор коммуникационного оборудования (мостов, коммутаторов, маршрутизаторов);
- выбор среды передачи данных (каналов или линий связи);
- выбор периферийного оборудования (принтеров, внешней памяти, средств ввода/вывода и отображения информации);
- выбор средств электропитания и различной оснастки (шкафов, коробов) и т. д.

Приведенные выше критерии для выбора логической структуры могут быть также применены как при построении физической структуры ЛВС в це-

лом, так и при выборе элементов ЛВС и составных частей каждого элемента ЛВС.

## 2. Задача выбора канальной среды

Каждый элемент ЛВС вносит свой вес в качество функционирования ЛВС, но наибольшее влияние на качество работы ЛВС из приведенной совокупности элементов оказывает среда передачи данных (канальная среда). При работе ЛВС большое число сбоев и отказов приходится именно на канальную среду [4]. Поэтому далее целесообразно остановиться на выборе рациональных решений по построению канальной среды ЛВС.

Анализ используемых подходов к рациональному выбору канальной среды показывает, что эти подходы основываются на задании некоторых частных показателей или групп показателей, которые, по мнению разработчиков, являются доминирующими [5, 6]. При рациональном выборе канальной среды (проводной, беспроводной), как правило, учитываются следующие факторы:

- уровень организации канальной среды (структурированная или неструктурированная);
- виды и уровень воздействий внешней среды (помехи, влажность, загазованность, несанкционированный доступ (НСД), механические повреждения и т. д.);
- физико-технические, эксплуатационные, стоимостные характеристики каналов.

Предлагается разбить задачу выбора канальной среды на два этапа:

1) решение вопроса об уровне организации канальной среды;

2) определение конкретного типа (типов) используемых каналов с учетом внешних факторов, физико-технических, эксплуатационных и экономических (стоимостных) характеристик выбранных каналов.

Рассмотрим указанные этапы подробнее с точки зрения применения методов теории принятия решений.

Таблица 1

**Приоритеты альтернатив выбора канальной системы**

| Критерии (показатели) выбора альтернативной канальной системы | Относительная важность альтернатив по критериям | |
|---|---|---|
| | НКС | СКС |
| Стоимость установки и развертывания | 1 | 2 |
| Универсальность | 2 | 1 |
| Срок службы | 2 | 1 |
| Стоимость модернизации и развития | 2 | 1 |
| Возможность легкого расширения сети | 2 | 1 |
| Обеспечение более эффективного обслуживания | 2 | 1 |
| Надежность | 2 | 1 |

**На этапе 1** решается вопрос выбора между двумя альтернативами: неструктурированной (НКС) и структурированной (СКС) канальными системами. При этом можно рассмотреть, например, следующие основные показатели и приоритеты (табл. 1). Перечень критериев, значение относительной важности критериев и альтернатив может варьироваться в зависимости от специфики решаемой задачи или от предпочтений разработчика. В табл. 1 показан пример задания относительной важности двух альтернатив НКС и СКС по нескольким предложенным критериям.

**На этапе 2** осуществляется выбор конкретных типов каналов (линий) связи. При этом учитывается множество показателей [5, 6]: пропускная способность (скорость передачи), электромагнитная помехозащищенность, уровень затухания сигналов на единицу длины линии, протяженность линии (дальность связи), физическая защищенность (от разрыва, коррозии, несанкционированного доступа), перспективность применения, тип подсистемы СКС (горизонтальная, вертикальная, кампус), целесообразность применения на прогнозируемый период, пожелания заказчика, личные предпочтения и опыт разработчиков, стоимость и др.

Таблица 2

**Относительная важность альтернатив (типов каналов)**

| Критерии выбора типа каналов | Тип линии связи | | | | | |
|---|---|---|---|---|---|---|
| | Проводные | | | | Беспроводные | |
| | Витая пара | | Опто-волокно | Коаксиал | Радио | Инфра-красная |
| | UTP | STP | | | | |
| | ***A1*** | ***A2*** | ***A3*** | ***A4*** | ***A5*** | ***A6*** |
| ***K1.*** Пропускная способность (скорость передачи) | 3 | 2 | 1 | 4 | 5 | 6 |
| ***K2.*** Электромагнитная помехозащищенность | 3 | 2 | 1 | 4 | 6 | 5 |
| ***K3.*** Уровень затухания сигналов на единицу длины линии | 3 | 2 | 1 | 4 | 5 | 6 |
| ***K4.*** Протяженность линии (дальность связи) | 4 | 3 | 2 | 5 | 1 | 6 |
| ***K5.*** Физическая защищенность (разрыв, коррозия, несанкционированный доступ) | 4 | 3 | 1 | 2 | 5 | 6 |
| ***K6.*** Перспективность применения | 2 | 3 | 1 | 6 | 4 | 5 |
| ***K7.*** Предпочтения заказчика | 1 | 1 | 1 | 6 | 1 | 1 |

Наиболее часто разработчиками рассматриваются несколько основных критериев (табл. 2), однако при решении любой частной задачи набор критериев может изменяться в зависимости от предпочтений разработчиков и конкретных условий. При этом методика применения рассмотренных методов теории принятия решений останется прежней.

## 3. Пример решения задачи выбора типа каналов

В качестве примера предлагается рассмотреть принятие одного из частных решений, а именно — выбора типа каналов. При этом критериями выбора решения не всегда являются числовые оценки, они могут быть и лингвистическими. Одним из наиболее широко применяемых методов поддержки принятия решений при проектировании в технических областях является метод аналитических иерархий [7], который лежит в основе большинства современных систем поддержки принятия решений (СППР), таких как Expert choice, "Оценка и выбор" и др. Можно отметить множество достоинств этого метода. Он наиболее прозрачен и понятен широкому кругу не подготовленных пользователей, он позволяет контролировать ход принятия решений (в отличие от прочих, например метода ELECTRE). Метод основан на построении иерархии критериев, что дает возможность при необходимости решать задачи с вложенной структурой критериев. Метод позволяет работать как с численными, так и с лингвистическими критериями и направлен на индивидуальное принятие решения без необходимости привлечения группы экспертов, как в методах мозгового штурма или Делфи.

### *3.1. Пример решения задачи выбора типа каналов с применением метода аналитических иерархий*

Рассмотрим пример применения метода для выбора одной из шести альтернативных типов линий связи (*A1-A6*) по нескольким критериям (*K1-K7*). В табл. 2 показан один из вариантов распределения приоритетов альтернатив по указанным критериям.

Коэффициенты относительной важности критериев могут рассчитываться как на основе их попарного сравнения, так и на основе экспертных процедур (метод предпочтения и метод ранга) [3]. Рассмотрим первый способ — *попарного сравнения критериев*, где задаются величины $x_{ij}$ — важность критерия *Ki* по сравнению с критерием *Kj*. Тогда очевидно, что $x_{ij} = 1/x_{ji}$. Используем следующую шкалу относительной важности критериев сравнительной оценки:

- равная важность — 1;
- умеренное превосходство — 3;
- существенное и сильное превосходство — 5;
- значительное превосходство — 7;
- очень большое превосходство — 9.

Таблица 3

**Взаимная важность критериев**

| $x_{ij}$ | | $j$ | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| $i$ | 1 | 1 | 3 | 5 | 5 | 7 | 7 | 9 |
| | 2 | 1/3 | 1 | 3 | 5 | 5 | 7 | 7 |
| | 3 | 1/5 | 1/3 | 1 | 3 | 5 | 5 | 7 |
| | 4 | 1/5 | 1/5 | 1/3 | 1 | 3 | 5 | 5 |
| | 5 | 1/7 | 1/5 | 1/5 | 1/3 | 1 | 3 | 5 |
| | 6 | 1/7 | 1/7 | 1/5 | 1/5 | 1/3 | 1 | 3 |
| | 7 | 1/9 | 1/7 | 1/7 | 1/5 | 1/5 | 1/3 | 1 |

Будем считать, что критерии в табл. 2 расположены по приоритетности (*K1* — наиболее приоритетный), при этом таблица взаимной важности критериев выглядит как табл. 3.

Тогда, согласно методу аналитических иерархий [7], рассчитываются:

- цена каждого критерия по формуле $C_{Ki} = \sqrt[7]{\prod_j x_{ij}}$:

$$C_{K1} = 4{,}42;\ C_{K2} = 2{,}76;\ C_{K3} = 1{,}66;\ C_{K4} = 1{,}00;$$
$$C_{K5} = 0{,}60;\ C_{K6} = 0{,}36;\ C_{K7} = 0{,}23;$$

- сумма цен критериев $K = 11{,}03$;
- вес каждого критерия $W_{Ki} = \dfrac{C_{Ki}}{K}$:

$$W_{K1} = 0{,}40;\ W_{K2} = 0{,}25;\ W_{K3} = 0{,}15;\ W_{K4} = 0{,}09;$$
$$W_{K5} = 0{,}05;\ W_{K6} = 0{,}03;\ W_{K7} = 0{,}02.$$

Затем следует провести сравнение альтернатив. Для этого строятся семь таблиц взаимного предпочтения альтернатив по каждому из семи критериев (табл. 4—10). Элементом каждой таблицы $y_{ij}^{Kl}$ является относительная важность альтернативы *Ai* по сравнению с альтернативой *Aj* по критерию *Kl*. При составлении матриц воспользуемся приоритетом каждой альтернативы согласно табл. 2. Иногда для заполнения этих матриц может потребоваться привлечение экспертов, а значит, использование групповых методов поддержки принятия решений [3].

Таблица 4

**Взаимное предпочтение альтернатив по критерию *K1***

| $y_{ij}^{K1}$ | | $j$ | | | | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 |
| $i$ | 1 | 1 | 1/3 | 1/5 | 3 | 7 | 7 |
| | 2 | 3 | 1 | 1/3 | 5 | 7 | 9 |
| | 3 | 5 | 3 | 1 | 7 | 9 | 9 |
| | 4 | 1/3 | 1/5 | 1/7 | 1 | 3 | 5 |
| | 5 | 1/7 | 1/7 | 1/9 | 1/3 | 1 | 3 |
| | 6 | 1/7 | 1/9 | 1/9 | 1/5 | 1/3 | 1 |

Таблица 5

**Взаимное предпочтение альтернатив по критерию *K2***

| $y_{ij}^{K2}$ | | $j$ | | | | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 |
| $i$ | 1 | 1 | 1/3 | 1/5 | 3 | 7 | 5 |
| | 2 | 3 | 1 | 1/3 | 5 | 9 | 7 |
| | 3 | 5 | 3 | 1 | 7 | 9 | 9 |
| | 4 | 1/3 | 1/5 | 1/7 | 1 | 5 | 3 |
| | 5 | 1/7 | 1/9 | 1/9 | 1/5 | 1 | 1/3 |
| | 6 | 1/5 | 1/7 | 1/9 | 1/3 | 3 | 1 |

Таблица 6

**Взаимное предпочтение альтернатив по критерию *K3***

| $y_{ij}^{K3}$ | | $j$ | | | | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 |
| $i$ | 1 | 1/3 | 1/5 | 3 | 7 | 7 | 1/3 |
| | 2 | 1 | 1/3 | 5 | 7 | 9 | 1 |
| | 3 | 3 | 1 | 7 | 9 | 9 | 3 |
| | 4 | 1/5 | 1/7 | 1 | 3 | 5 | 1/5 |
| | 5 | 1/7 | 1/9 | 1/3 | 1 | 3 | 1/7 |
| | 6 | 1/9 | 1/9 | 1/5 | 1/3 | 1 | 1/9 |

Таблица 7

**Взаимное предпочтение альтернатив по критерию *K4***

| $y_{ij}^{K4}$ | | $j$ | | | | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 |
| $i$ | 1 | 1 | 1/3 | 1/5 | 3 | 1/7 | 5 |
| | 2 | 3 | 1 | 1/3 | 5 | 1/5 | 7 |
| | 3 | 5 | 3 | 1 | 7 | 1/3 | 9 |
| | 4 | 1/3 | 1/5 | 1/7 | 1 | 1/9 | 3 |
| | 5 | 7 | 5 | 3 | 9 | 1 | 1/3 |
| | 6 | 1/5 | 1/7 | 1/9 | 1/3 | 3 | 1 |

Таблица 8

**Взаимное предпочтение альтернатив по критерию *K5***

| $y_{ij}^{K5}$ | | $j$ | | | | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 |
| $i$ | 1 | 1 | 1/3 | 1/7 | 1/5 | 3 | 5 |
| | 2 | 3 | 1 | 1/5 | 1/3 | 5 | 7 |
| | 3 | 7 | 5 | 1 | 3 | 9 | 9 |
| | 4 | 5 | 3 | 1/3 | 1 | 7 | 9 |
| | 5 | 1/3 | 1/5 | 1/9 | 1/7 | 1 | 3 |
| | 6 | 1/5 | 1/7 | 1/9 | 1/9 | 1/3 | 1 |

Таблица 9

**Взаимное предпочтение альтернатив по критерию *K6***

| $y_{ij}^{K6}$ | | $j$ | | | | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 |
| $i$ | 1 | 1 | 3 | 5 | 7 | 9 | 9 |
| | 2 | 1/3 | 1 | 3 | 5 | 7 | 9 |
| | 3 | 1/5 | 1/3 | 1 | 3 | 5 | 7 |
| | 4 | 1/7 | 1/5 | 1/3 | 1 | 3 | 5 |
| | 5 | 1/9 | 1/7 | 1/5 | 1/3 | 1 | 3 |
| | 6 | 1/9 | 1/9 | 1/7 | 1/5 | 1/3 | 1 |

Таблица 10

**Взаимное предпочтение альтернатив по критерию *K7***

| $y_{ij}^{K7}$ | | $j$ | | | | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 |
| $i$ | 1 | 1 | 1 | 1 | 9 | 1 | 1 |
| | 2 | 1 | 1 | 1 | 9 | 1 | 1 |
| | 3 | 1 | 1 | 1 | 9 | 1 | 1 |
| | 4 | 1/9 | 1/9 | 1/9 | 1 | 1/9 | 1/9 |
| | 5 | 1 | 1 | 1 | 9 | 1 | 1 |
| | 6 | 1 | 1 | 1 | 9 | 1 | 1 |

Далее требуется рассчитать ряд показателей:

- цена альтернатив по каждому критерию

$$C_{Ki,\,Aj} = \sqrt[6]{\prod_j y_{ij}^{Ki}}:$$

$C_{A1,\,K1} = 1{,}46;\ C_{A2,\,K1} = 2{,}61;\ C_{A3,\,K1} = 4{,}52;$
$C_{A4,\,K1} = 0{,}72;\ C_{A5,\,K1} = 0{,}36;\ C_{A6,\,K1} = 0{,}22;$

$C_{A1,\,K2} = 1{,}38;\ C_{A2,\,K2} = 2{,}61;\ C_{A3,\,K2} = 4{,}52;$
$C_{A4,\,K2} = 0{,}72;\ C_{A5,\,K2} = 0{,}22;\ C_{A6,\,K2} = 0{,}38;$

$C_{A1,\,K3} = 1{,}46;\ C_{A2,\,K3} = 2{,}61;\ C_{A3,\,K3} = 4{,}52;$
$C_{A4,\,K3} = 0{,}72;\ C_{A5,\,K3} = 0{,}36;\ C_{A6,\,K3} = 0{,}22;$

$C_{A1,\,K4} = 0{,}87;\ C_{A2,\,K4} = 1{,}38;\ C_{A3,\,K4} = 2{,}61;$
$C_{A4,\,K4} = 0{,}38;\ C_{A5,\,K4} = 2{,}61;\ C_{A6,\,K4} = 0{,}38;$

$C_{A1,\,K5} = 0{,}72;\ C_{A2,\,K5} = 1{,}38;\ C_{A3,\,K5} = 4{,}52;$
$C_{A4,\,K5} = 2{,}61;\ C_{A5,\,K5} = 0{,}38;\ C_{A6,\,K5} = 0{,}22;$

$C_{A1,\,K6} = 4{,}52;\ C_{A2,\,K6} = 2{,}61;\ C_{A3,\,K6} = 1{,}38;$
$C_{A4,\,K6} = 0{,}72;\ C_{A5,\,K6} = 0{,}38;\ C_{A6,\,K6} = 0{,}22;$

$C_{A1,\,K7} = 1{,}44;\ C_{A2,\,K7} = 1{,}44;\ C_{A3,\,K7} = 1{,}44;$
$C_{A4,\,K7} = 0{,}16;\ C_{A5,\,K7} = 1{,}44;\ C_{A6,\,K6} = 1{,}44;$

- сумма цен альтернатив по каждому критерию $A_{Ki} = \sum_j C_{Ki,\,Aj}$:

$A_{K1} = 9{,}89;\ A_{K2} = 9{,}83;\ A_{K3} = 9{,}89;\ A_{K4} = 8{,}23;$
$A_{K5} = 9{,}83;\ A_{K6} = 9{,}83;\ A_{K7} = 7{,}36;$

- ценность каждой альтернативы по каждому критерию $V_{Aj,\,Ki} = \dfrac{C_{Aj,\,Ki}}{A_{Ki}}$:

$V_{A1,\,K1} = 0{,}15;\ V_{A2,\,K1} = 0{,}26;\ V_{A3,\,K1} = 0{,}46;$
$V_{A4,\,K1} = 0{,}07;\ V_{A5,\,K1} = 0{,}04;\ V_{A6,\,K1} = 0{,}02;$

$V_{A1,\,K2} = 0{,}14;\ V_{A2,\,K2} = 0{,}27;\ V_{A3,\,K2} = 0{,}46;$
$V_{A4,\,K2} = 0{,}07;\ V_{A5,\,K2} = 0{,}02;\ V_{A6,\,K2} = 0{,}04;$

$V_{A1,\,K3} = 0{,}15;\ V_{A2,\,K3} = 0{,}26;\ V_{A3,\,K3} = 0{,}46;$
$V_{A4,\,K3} = 0{,}07;\ V_{A5,\,K3} = 0{,}02;\ V_{A6,\,K3} = 0{,}87;$

$V_{A1,\,K4} = 0{,}11;\ V_{A2,\,K4} = 0{,}17;\ V_{A3,\,K4} = 0{,}32;$
$V_{A4,\,K4} = 0{,}04;\ V_{A5,\,K4} = 0{,}32;\ V_{A6,\,K4} = 0{,}04;$

$V_{A1,\,K5} = 0{,}07;\ V_{A2,\,K5} = 0{,}14;\ V_{A3,\,K5} = 0{,}46;$
$V_{A4,\,K5} = 0{,}27;\ V_{A5,\,K5} = 0{,}04;\ V_{A6,\,K5} = 0{,}02;$

$V_{A1,\,K6} = 0{,}14;\ V_{A2,\,K6} = 0{,}27;\ V_{A3,\,K6} = 0{,}14;$
$V_{A4,\,K6} = 0{,}07;\ V_{A5,\,K6} = 0{,}04;\ V_{A6,\,K6} = 0{,}02;$

$V_{A1,\,K7} = V_{A2,\,K7} = V_{A3,\,K7} = V_{A5,\,K7} = V_{A6,\,K6} = 0{,}196;$
$V_{A4,\,K7} = 0{,}02;$

- итоговая ценность альтернатив

$$U_{Ai} = \sum_{j=1}^{7} (V_{Ai,\,Kj} W_{Kj}),\ i = 1, 2, \ldots, 6:$$

$U_{A1} = 0{,}14;\ U_{A2} = 0{,}24;\ \mathbf{U_{A3} = 0{,}59};\ U_{A4} = 0{,}07;$
$U_{A5} = 0{,}06;\ U_{A6} = 0{,}16.$

Очевидно, что предпочтительней альтернатива *A3* с максимальной ценностью, т. е. оптоволоконная линия связи. Ранжирование по степени предпочтительности дает следующий результат (в порядке убывания): *A3* (ценность = 0,59), *A2* (ценность = 0,24), *A6* (ценность = 0,16), *A1* (ценность = 0,14), *A4* (ценность = 0,07), *A5* (ценность = 0,06).

### *3.2. Пример решения задачи выбора типа каналов с применением метода ELECTRE*

Если бы все критерии можно было бы выразить численно, то одним из возможных методов принятия решений в данном случае был бы метод ELECTRE, позволяющий исключить неэффективные альтернативы при их попарном сравнении. Этот метод менее прозрачен, чем метод аналитических иерархий. Но он имеет относительно невысокую вычислительную сложность и позволяет уяснить и поэтапно корректировать требования к окончательной цели проекта. Допустим, согласно рассмотренным в табл. 2 приоритетам требуется выбрать наилучшую из четырех альтернатив *A1*, *A3*, *A4*, *A6* по критериям *K1*, *K3*, *K5*. Тогда составим новую таблицу взаимной важности типов каналов (табл. 11).

Допустим, что 10 экспертов отдали свои голоса за наибольшую важность одного критерия следующим образом: *K1* — 5 голосов, *K3* — 3 голоса, *K5* — 2 голоса, т. е. веса критериев соответственно равны $W_{K1} = 5$, $W_{K3} = 3$, $W_{K5} = 2$, а общий вес критериев равен 10. Тогда, пользуясь табл. 2 и проведя попарное сравнение альтернатив [7], составим таблицы индексов согласия $c_{ij}$ (табл. 12) и несогласия $d_{ij}$ (табл. 13) с тем, что альтернатива *Ai* превосходит альтернативу *Aj*, которые вычисляются по следующим правилам. Для $c_{ij}$ в числителе суммируются веса критериев, по которым альтернатива *Ai* предпочтительнее (+) или равноценна *Aj* (=), а в знаменателе — все веса критериев. Для $d_{ij}$ в числителе имеем максимум из разности оценок ($l$) альтернатив *Aj* и *Ai* по $i$-му критерию, а в знаменателе — длину шкалы оценок ($L$) $i$-го критерия: $c_{ij} = \dfrac{\sum\limits_{i \in I^+, I^=} W_{Ki}}{\sum\limits_{1}^{n} W_{Ki}}$, $d_{ij} = \max\limits_{i \in I^-} \dfrac{l_{i,Aj} - l_{i,Ai}}{L_i}$.

Таблица 11

**Взаимная важность (приоритеты) типов каналов по критериям**

| $l_{i,Aj}$ приоритет типов каналов | | *Aj* альтернатива выбора типа канала | | | |
|---|---|---|---|---|---|
| | | *A1* | *A3* | *A4* | *A6* |
| *i* критерий выбора типа каналов | 1 (*K1*) | 3 | 1 | 4 | 6 |
| | 3 (*K3*) | 3 | 1 | 4 | 6 |
| | 5 (*K5*) | 4 | 1 | 2 | 6 |

Таблица 12

**Индексы согласия**

| $c_{ij}$ | | *j* | | | |
|---|---|---|---|---|---|
| | | 1 | 3 | 4 | 6 |
| *i* | 1 | * | 0 | 0,8 | 1 |
| | 3 | 1 | * | 1 | 1 |
| | 4 | 0,2 | 0 | * | 1 |
| | 6 | 0 | 0 | 0 | * |

Таблица 13

**Индексы несогласия**

| $d_{ij}$ | | *j* | | | |
|---|---|---|---|---|---|
| | | 1 | 3 | 4 | 6 |
| *i* | 1 | * | 0,6 | 0,4 | 0 |
| | 3 | 0 | * | 0 | 0 |
| | 4 | 0,2 | 0,6 | * | 0 |
| | 6 | 0,6 | 0,6 | 0,6 | * |

Далее проводится несколько циклов попарного сравнения альтернатив. Для первого цикла задаем уровни согласия $c = 0{,}8$ и несогласия $d = 0{,}6$ и определяем лучшую альтернативу в паре (*Ai* — *Aj*), если $c_{ij} \geqslant 0{,}8$ и $d_{ij} \leqslant 0{,}6$. Уже на первом цикле выясняем, что

*A3* лучше *A1*, *A1* лучше *A4*, *A1* лучше *A6*,
*A3* лучше *A4*, *A3* лучше *A6*,
*A4* лучше *A6*,
} *A3* лучше всех.

Если бы пришлось проводить еще один цикл сравнения, тогда следовало бы выбрать новые более жесткие уровни согласия $c$ и несогласия $d$. Применив метод ELECTRE, выясняем, что альтернатива *A3* — оптоволоконные линии связи, лучше остальных альтернатив по предложенным критериям.

## Заключение

Предложенный подход позволяет провести обоснованный выбор одного из альтернативных типов каналов с учетом заданных критериев. Как было показано выше, задача проектирования ЛВС многоэтапная и многовариантная. Число альтернатив и критериев оценки качества проектных решений достаточно велико, что делает оправданным использование методов теории принятия решений, которые просты в применении и экономят вычислительные ресурсы.

Кроме того, предложенные методы позволяют автоматизировать процедуры выбора проектных решений ЛВС. А это, в свою очередь, приведет к сокращению сроков разработки проекта, повышению объективности принимаемых решений, а также поможет сформировать базу по ранее принятым решениям, которые могут впоследствии использоваться разработчиками.

### Список литературы


1. **Вишнеков А. В., Иванова Е. М.** Сравнительная оценка вычислительных систем по критериям пользователя // Качество. Инновации. Образование. 2013. № 4. С. 63—68.
2. **Вишнеков А. В., Иванова Е. М.** Интеграция методов поддержки принятия решений в автоматизированных СППР при разработке сложных проектов // Проблемы информатики. 2013. № 2. С. 56—64.
3. **Трахтенгерц Э. А.** Компьютерная поддержка принятия решений. М.: СИНТЕК, 1998. 376 с.
4. **Сайт** проекта "Локальные и пиринговые сети Новосибирска". Раздел "статьи". Оптимизация локальных сетей. URL: http://netnsk.ru/publica/inet/mbr_23.php
5. **Гуров И. П.** Основы теории информации и передачи сигналов. СПб.: BHV-Санкт-Петербург, 2000. 97 с. URL: www.ict.edu.ru/ft/000004/HTML/3.htm
6. **Сайт** СК Энерго. Средства промышленной автоматизации. Критерии выбора типа сети. URL: www.siemens71.ru/RUS_6047.shtml
7. **Ларичев О. И.** Теория и методы принятия решений, а также Хроника событий в Волшебных странах: учебник. Изд. 2-е, перераб. и доп. М.: Логос, 2002. 392 с.

УДК 621.391

**С. В. Дворников**[1], д-р техн. наук, доц., проф. каф., e-mail: practicdsv@yandex.ru,
**В. В. Цветков**[2], нач. отд., **А. А. Устинов**[1], д-р техн. наук, проф., ст. науч. сотр.
[1] Военная академия связи, г. Санкт-Петербург,
[2] ЗАО "НПФ "ТИРС", г. Санкт-Петербург

# Компенсация движения при кодировании подвижных изображений на основе разбиения кодируемого блока кадров на непересекающиеся группы

*Изложено развитие известного метода оптимального кодирования трехмерных фрагментов подвижных изображений с различной степенью подвижности на основе изменения их размеров. Осуществлена постановка задачи разбиения кодируемых трехмерных фрагментов подвижных изображений на непересекающиеся группы. Целью такого разбиения является снижение межкадровых различий внутри каждой группы, что приводит к уменьшению числа значимых коэффициентов трехмерного преобразования, а это в свою очередь приводит к увеличению коэффициента сжатия. Обоснован подход к определению оптимального числа областей разбиения. Предложен алгоритм решения задачи разбиения на основе метода случайного поиска с наилучшей пробой. Приведены результаты имитационного моделирования.*

***Ключевые слова:*** *метод оптимального кодирования, трехмерный фрагмент подвижного изображения, коэффициент сжатия, межкадровые различия*

**S. V. Dvornikov, V. V. Cvetkov, A. A. Ustinov**

# The Compensation of the Motion when Coding of the Mobile Scenes by Method of Fission of the Coded Frames on the Separating Groups

*The development of the known method of the optimum, encoding three-dimensional fragment of the mobile scenes, having other degree in transportability, on count of the change of their size is described. The purpose of the new method consists in reduction the external difference of the frames difference inwardly each groups of the frames. This reduces the numbers significant factor three-dimensional transformation. The reduction of the factor of the transformation enlarges the factor to compression is proved. The active method of the optimum amount of the determination, separating area is motivated. The algorithm of the decision of the problem to sections in the base of the method of casual searching for of the best result is offered. Results of simulation modeling are introduced.*

***Keywords:*** *method of the optimum coding, three-dimensional fragment of the mobile scene, factor to compression, differences between frames*

## 1. Сжатие подвижных изображений на основе трехмерных ортогональных преобразований

Кадры подвижного изображения характеризуются как внутрикадровой, или пространственной избыточностью, так и межкадровой, или временной избыточностью. Как правило, в известных стандартах сжатия, например, Н.263, Н.264 для устранения внутрикадровой избыточности используют какое-либо декоррелирующее преобразование, например, двумерное дискретное косинусное преобразование (ДКП-2). Для устранения межкадровой избыточности используют межкадровое предсказание на основе передачи векторов движения. В способах кодирования на основе трехмерных ортогональных преобразований устранение внутрикадровой и межкадровой избыточности осуществляется путем декорреляции пикселей исходного изображения как по пространственным координатам, так и по оси времени. В результате декорреляции большая часть коэффициентов трехмерного ортогонального преобразования оказывается нулевой или близкой к нулю, что и обеспечивает уменьшение

требуемого числа бит, необходимого для кодирования коэффициентов ТДКП и, как следствие, достижение высоких коэффициентов сжатия. Одним из наиболее распространенных видов трехмерного ортогонального преобразования, которое используется при сжатии подвижных изображений, является трехмерное дискретное косинусное преобразование (ДКП-3).

Прямое трехмерное дискретное косинусное преобразование для куба размером $N \times N \times N$ задается следующим образом [1]:

$$F(i, j, k) = \sqrt{\frac{8}{N^3}}\, c(i)c(j)c(k) \sum_{z=0}^{N-1} \sum_{y=0}^{N-1} \sum_{x=0}^{N-1} f(x, y, z) \times$$

$$\times \cos\left[\frac{(2x+1)\pi i}{2N}\right] \cos\left[\frac{(2y+1)\pi j}{2N}\right] \cos\left[\frac{(2z+1)\pi k}{2N}\right],$$

где $f(x, y, z)$ — значение яркостной или цветоразностной компоненты пикселя с координатами $x, y, z \in [0, ..., N-1]$; $F(i, j, k)$ — коэффициент преобразования с координатами $i, j, k \in [0, ..., N-1]$; функция $c(k)$ определяется как

$$c(k) = \begin{cases} 1/2, & k = 0, \\ 1, & k \neq 0. \end{cases}$$

Обратное дискретное косинусное преобразование вычисляется следующим образом:

$$f(x, y, z) = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} \sqrt{\frac{8}{N^3}}\, c(i)c(j)c(k)F(i, j, k) \times$$

$$\times \cos\left[\frac{(2x+1)\pi j}{2N}\right] \cos\left[\frac{(2z+1)\pi k}{2N}\right].$$

ДКП-3 является ортонормальным преобразованием и в соответствии с приведенными выражениями может быть декомпозировано в виде последовательного выполнения трех одномерных преобразований. Кроме того, ортонормальность ДКП-3 означает, что энергия кодируемого фрагмента изображения, под которой понимается сумма квадратов значений его пикселей, равна энергии коэффициентов трехмерного преобразования, под которой понимается сумма квадратов их значений. Как видно на рис. 1, большая часть коэффициентов трехмерного ортогонального преобразования оказывается равной нулю. Это означает, что в результате выполнения трехмерного ортогонального преобразования над исходным изображением большая часть его энергии оказывается локализованной в незначительном числе ненулевых коэффициентов преобразования. Следовательно, ДКП-3 обладает следующими важными свойствами:

- локализация большой части энергии сигнала в небольшом числе коэффициентов преобразования. Данное свойство позволяет исключить наименее значимые коэффициенты из рассмотрения при кодировании с потерями [1];
- ДКП-3 представляет собой ортогональное преобразование, которое может быть реализовано путем последовательного выполнения одномерных ДКП по строкам, столбцам и оси времени [1].

Данные свойства определили широкое применение ДКП-3 при сжатии подвижных изображений. Типовая схема кодека на основе ДКП-3 предполагает выполнение следующих основных операций: входная обработка; ДКП-3; квантование коэффициентов преобразования, энтропийное кодирование. На приеме данные процедуры выполняются в обратной последовательности [1].

Однако в условиях кодирования высокодинамичных сцен ДКП-3 может быть не оптимальным в смысле концентрации энергии в незначительном числе коэффициентов, расположенных на временной оси (в различных кадрах кодируемого блока). Для примера на рис. 1 показаны квантованные коэффициенты ДКП-3 для кодируемого блока размером 8 × 8 × 8. Данный блок был намеренно выбран из высокодинамичной области кадров тестового подвижного изображения "Foreman". Анализ пред-

*а*)

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 183 | -3 | 0 | -1 | 0 | 0 | 0 | 0 |
| -12 | -1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | -1 | 0 | 0 | 0 | 0 |
| -1 | -1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

*б*)

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 3 | 5 | 0 | 1 | 0 | 0 | 0 | 0 |
| -11 | -1 | 0 | -1 | 0 | 0 | 0 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| -1 | -1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

*в*)

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 32 | 2 | -1 | 0 | 0 | 0 | 0 | 0 |
| -10 | -2 | 1 | 0 | 0 | 0 | 0 | 0 |
| -2 | -1 | 0 | 0 | 0 | 0 | 0 | 0 |
| -1 | 1 | -1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

*г*)

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 17 | 2 | 0 | 0 | 0 | 0 | 0 | 0 |
| -1 | 2 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | -2 | -1 | 0 | 0 | 0 | 0 | 0 |
| -1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

*д*)

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 16 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 1 | -1 | 0 | 0 | 0 | 0 | 0 |
| -1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| -1 | -1 | 1 | 0 | 0 | 0 | 0 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

*е*)

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 2 | -1 | 2 | 0 | 0 | 0 | 0 | 0 |
| 4 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| -3 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| -1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

*ж*)

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 4 | -2 | -1 | 1 | 0 | 0 | 0 | 0 |
| 4 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| -1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| -2 | -1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

*з*)

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 1 | -3 | -1 | 0 | 0 | 0 | 0 | 0 |
| 2 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Рис. 1. Пример коэффициентов ДКП-3:**
*а* — 1-го слоя куба 8 × 8 × 8; *б* — 2-го слоя куба 8 × 8 × 8; *в* — 3-го слоя куба 8 × 8 × 8; *г* — 4-го слоя куба 8 × 8 × 8; *д* — 5-го слоя куба 8 × 8 × 8; *е* — 6-го слоя куба 8 × 8 × 8; *ж* — 7-го слоя куба 8 × 8 × 8; *з* — 8-го слоя куба 8 × 8 × 8

ставленных коэффициентов ДКП-3 показывает наличие ненулевых коэффициентов в каждой плоскости куба 8 × 8 × 8.

Наиболее распространенным подходом к устранению данного недостатка является изменение размеров кодируемых трехмерных блоков [2—8]. В основе данного подхода лежит оценка степени подвижности каждого трехмерного кодируемого куба по нескольким фиксированным градациям. Далее для каждой градации определяется оптимальный размер кодируемого куба, например, 16 × 16 × 1, 16 × 16 × 8 или 8 × 8 × 8. Недостатки такого подхода очевидны. Во-первых, градация на заданное число уровней подвижности является не вполне обоснованной и носит крайне эмпирический характер. Во-вторых, фиксированное число уровней градации не может описать всего многообразия степени изменения сцен в трехмерных кодируемых фрагментах. В следующем разделе рассмотрим развитие данной идеи на основе оптимального разбиения кодируемого блока на непересекающиеся наборы кадров.

## 2. Постановка задачи поиска оптимального разбиения трехмерных кубов на непересекающиеся области

Пусть на вход кодера поступают $L$ кадров подвижного изображения размером $P_1 \times N_1$ пикселей. Разделим видеоданные на непересекающиеся области (кубы) размером $P \times N \times L$ пикселей в виде трехмерных матриц $\mathbf{A}_{P\times N\times L}$. В целях упрощения дальнейших рассуждений трехмерную матрицу $\mathbf{A}_{P\times N\times L}$ представим множеством из $L$ одномерных векторов размерностью $P \times N$ элементов: $\mathbf{A}_{P\times N\times L} \to \{\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_L\}$. Отметим, что любой $i$-й вектор сформированного множества из $L$ векторов получен путем последовательной $N$-кратной канкатенации $P$-мерных столбцов $i$-го слоя матрицы $\mathbf{A}_{P\times N\times L}$ в единый вектор размерности $P \times N$. Используя терминологию линейной алгебры, множество $\{\mathbf{A}\} = \{\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_L\}$ будем рассматривать как множество из $L$ точек в евклидовом пространстве размерности $P \times N$ ($E^{P\times N}$).

Разделим исходное множество $\{\mathbf{A}\}$ на $R$ подмножеств: $\{\mathbf{A}\}_1$, $\{\mathbf{A}\}_2$, ..., $\{\mathbf{A}\}_R$ так, чтобы выполнялись условия:

1. $\{\mathbf{A}\}_i \cup \{\mathbf{A}\}_j = \{\mathbf{A}\}$;
2. $\{\mathbf{A}\}_i \cap \{\mathbf{A}\}_j = \{0\}$, $i = 1, 2, ..., R$; $j = 1, 2, ..., R$; $i \neq j$.

Первое условие обеспечивает объединение $R$ подмножеств в единое множество $\{\mathbf{A}\}$. Второе условие обеспечивает то, что $R$ образованных подмножеств являются непересекающимися.

Исходя из необходимости достижения максимального сжатия, будем считать некоторое разбиение множества $\{\mathbf{A}\}$ оптимальным, если суммарное число ненулевых коэффициентов преобразования после выполнения ДКП-3 над каждым образованным подмножеством является минимальным.

Введем функцию $f_{\text{ДКП-3}}(\{\mathbf{A}\}_i)$, которая определяет число ненулевых (значимых) коэффициентов преобразования, полученных после выполнения ДКП-3 над $i$-м подмножеством. Тогда формально задачу поиска оптимального разбиения можно записать следующим образом:

$$\sum_{i=1}^{R} f_{\text{ДКП-3}}(\{\mathbf{A}\}_i) \to \min_{\{\mathbf{A}\}_i \ \forall i = \overline{1,R}}. \quad (1)$$

Задача (1) предполагает поиск некоторого разбиения на конечном множестве возможных разбиений, для которого суммарное число ненулевых коэффициентов преобразования по каждому подмножеству разбиения минимально. В этом смысле данная задача является целочисленной. Кроме того, в ходе решения задачи необходимо для каждого разбиения выполнять ДКП-3 по всем сформированным подмножествам. В связи с этим метод решения данной задачи, основанный на переборе возможных разбиений с последующим выполнением ДКП-3 для найденных подмножеств, является сложным. Рассмотрим постановку задачи поиска оптимального разбиения, которая не требует вычисления ДКП-3 для оценки числа значимых коэффициентов.

Известно [9], что ДКП-3 является наиболее эффективным, если корреляция между соседними элементами внутри обрабатываемого куба является относительно высокой. Коэффициент корреляции между соседними элементами можно сопоставить со степенью их похожести или близости. Выберем в качестве меры близости средний квадрат ошибки (СКО). Тогда можно утверждать, что чем меньше СКО между элементами, тем выше коэффициент корреляции между ними. Следовательно, для решения задачи (1) без выполнения ДКП-3 необходимо найти такое разбиение исходного множества на $R$ непересекающихся подмножеств, при котором суммарный СКО между элементами каждого из подмножеств был бы минимальным.

Для формальной постановки такой задачи необходимо математически описать следующее:

- способ разбиения исходного множества $\{\mathbf{A}\}$ на $R$ непересекающихся подмножеств;
- способ вычисления СКО между элементами каждого из подмножеств.

Разбиение исходного множества $\{\mathbf{A}\}$ на $R$ непересекающихся подмножеств зададим в виде матрицы $\mathbf{X}$ размером $R \times L$ элементов. На элементы матрицы $\mathbf{X}$ наложим следующие ограничения:

$$x(i, j) \in \{1, 0\}; \quad (2)$$

$$\sum_{i=1}^{R} x(i, j) = 1, \ \forall j = \overline{1, L}. \quad (3)$$

Ограничение (2) формально описывает способ разбиения:

если $X(i, j) = 1$, то $\mathbf{a}_j \in \{\mathbf{A}\}_i$;

если $X(i, j) = 0$, то $\mathbf{a}_j \notin \{\mathbf{A}\}_i$.

Ограничения (2) и (3) обеспечивают выполнение условия непересекаемости подмножеств: каждый элемент $\mathbf{a}_j$, $j = 1, 2, ..., L$, может принадлежать только одному подмножеству.

Для вычисления СКО между элементами определим средний элемент $i$-го подмножества в виде [2]:

$\hat{\mathbf{A}}_i = \dfrac{\mathbf{A}\mathbf{x}_i^{\mathrm{T}}}{\mathbf{x}_i\mathbf{1}}$, где $\mathbf{A}$ — матрица размером $P \times L$ элементов; $\mathbf{x}_i$ — $i$-я строка матрицы; $\mathbf{1}$ — единичный вектор-столбец размером $L \times 1$. Отметим, что каждый $j$-й столбец матрицы $\mathbf{A}(\mathbf{a}_j)$ является $j$-м элементом исходного множества $\{\mathbf{A}\}$.

Если $j$-й элемент принадлежит $i$-му подмножеству, то данный элемент определим в виде $\mathbf{a}_{j \in \{\mathbf{A}\}_i}$. Тогда СКО в $i$-й группе определим следующим образом:

$$\mathrm{СКО}_i = \sum_{j=1}^{n_i} \left( \mathbf{a}_{j \in \{\mathbf{A}\}_i} - \frac{\mathbf{A}\mathbf{x}_i^{\mathrm{T}}}{\mathbf{x}_i\mathbf{1}} \right)^{\mathrm{T}} \left( \mathbf{a}_{j \in \{\mathbf{A}\}_i} - \frac{\mathbf{A}\mathbf{x}_i^{\mathrm{T}}}{\mathbf{x}_i\mathbf{1}} \right), \quad (4)$$

где $n_i$ — число элементов множества $\{\mathbf{A}\}$, принадлежащих $i$-му подмножеству.

Исходя из физического смысла матрицы $\mathbf{X}$, принадлежность $j$-го элемента $i$-му множеству можно представить в виде произведения элемента на $x(i, j)$. В этом случае суммарное значение СКО по всем $R$ подмножествам запишем в виде

$$\mathrm{СКО} = \sum_{i=1}^{R} \sum_{j=1}^{L} \left( \mathbf{a}_j x(i,j) - x(i,j) \frac{\mathbf{A}\mathbf{x}_i^{\mathrm{T}}}{\mathbf{x}_i\mathbf{1}} \right)^{\mathrm{T}} \times \left( \mathbf{a}_j x(i,j) - x(i,j) \frac{\mathbf{A}\mathbf{x}_i^{\mathrm{T}}}{\mathbf{x}_i\mathbf{1}} \right). \quad (5)$$

Раскрыв скобки в выражении (5), получим:

$$\mathrm{СКО} = \sum_{i=1}^{R} \sum_{j=1}^{L} \left( \mathbf{a}_j^{\mathrm{T}} \mathbf{a}_j x^2(i,j) - 2x^2(i,j) \frac{\mathbf{a}_j^{\mathrm{T}} \mathbf{A}\mathbf{x}_i^{\mathrm{T}}}{\mathbf{x}_i\mathbf{1}} + x^2(i,j) \frac{\mathbf{x}_i \mathbf{A}^{\mathrm{T}} \mathbf{A}\mathbf{x}_i^{\mathrm{T}}}{(\mathbf{x}_i\mathbf{1})^2} \right). \quad (6)$$

Проанализируем каждое слагаемое в выражении (6) с учетом суммирования.

Слагаемое 1:

$$\sum_{i=1}^{R} \sum_{j=1}^{L} \mathbf{a}_j^{\mathrm{T}} \mathbf{a}_j x^2(i,j) = \sum_{j=1}^{L} \mathbf{a}_j^{\mathrm{T}} \mathbf{a}_j \sum_{i=1}^{R} x^2(i,j) = \sum_{j=1}^{L} \mathbf{a}_j^{\mathrm{T}} \mathbf{a}_j,$$

поскольку $\sum_{i=1}^{R} x^2(i,j) = \sum_{i=1}^{R} x(i,j) = 1$ в соответствии с ограничениями (2) и (3).

Слагаемое 2:

$$2 \sum_{i=1}^{R} \sum_{j=1}^{L} x^2(i,j) \frac{\mathbf{a}_j^{\mathrm{T}} \mathbf{A}\mathbf{x}_i^{\mathrm{T}}}{\mathbf{x}_i\mathbf{1}} = 2 \sum_{j=1}^{L} \mathbf{a}_j^{\mathrm{T}} \sum_{i=1}^{R} \frac{x^2(i,j) \mathbf{A}\mathbf{x}_i^{\mathrm{T}}}{\mathbf{x}_i\mathbf{1}} = 2 \sum_{i=1}^{R} \frac{\mathbf{x}_i \mathbf{A}^{\mathrm{T}} \mathbf{A}\mathbf{x}_i^{\mathrm{T}}}{\mathbf{x}_i\mathbf{1}}.$$

Слагаемое 3:

$$\sum_{i=1}^{R} \sum_{j=1}^{L} \mathbf{x}^2(i,j) \frac{\mathbf{x}_i \mathbf{A}^{\mathrm{T}} \mathbf{A}\mathbf{x}_i^{\mathrm{T}}}{(\mathbf{x}_i\mathbf{1})^2} = \sum_{j=1}^{L} x^2(i,j) \sum_{i=1}^{R} \frac{\mathbf{x}_i \mathbf{A}^{\mathrm{T}} \mathbf{A}\mathbf{x}_i^{\mathrm{T}}}{(\mathbf{x}_i\mathbf{1})^2} = \sum_{i=1}^{R} \frac{\mathbf{x}_i \mathbf{A}^{\mathrm{T}} \mathbf{A}\mathbf{x}_i^{\mathrm{T}}}{\mathbf{x}_i\mathbf{1}},$$

поскольку $\sum_{j=1}^{L} x^2(i,j) = \mathbf{x}_i\mathbf{1}$ при выполнении ограничения (2).

С учетом полученных выражений для слагаемых 1—3 выражение (6) окончательно запишем в следующем виде:

$$\mathrm{СКО} = \sum_{j=1}^{L} \mathbf{a}_j^{\mathrm{T}} \mathbf{a}_j - \sum_{i=1}^{R} \frac{\mathbf{x}_i \mathbf{A}^{\mathrm{T}} \mathbf{A}\mathbf{x}_i^{\mathrm{T}}}{\mathbf{x}_i\mathbf{1}}. \quad (7)$$

Как было отмечено ранее, оптимальным разбиением является разбиение, при котором минимизируется значение СКО. Следовательно, задача поиска оптимального разбиения сводится к минимизации выражения (7) по всем возможным матрицам разбиений. Поскольку первое слагаемое в выражении (7) от искомой переменной не зависит, то окончательно задачу поиска оптимального разбиения можно записать следующим образом:

$$\sum_{i=1}^{R} \frac{\mathbf{x}_i \mathbf{A}^{\mathrm{T}} \mathbf{A}\mathbf{x}_i^{\mathrm{T}}}{(\mathbf{x}_i\mathbf{1})} \to \max_{\mathbf{x}_i,\ \forall i = \overline{1,R}} \quad (8)$$

при ограничениях (2) и (3) на искомые значения элементов матрицы $\mathbf{X}$.

Решением задачи (8) является матрица $\mathbf{X}$, задающая оптимальное разбиение исходного трехмерного куба размером $P \times N \times L$ пикселей на $R$ непересекающихся подмножеств.

### 3. Практическое использование задачи оптимального разбиения для компенсации межкадровой избыточности подвижных изображений

Ниже рассмотрим практические аспекты решения задач (1) и (8) для компенсации движения.

Как было отмечено выше, целью поиска оптимального разбиения исходного фрагмента кадров подвижного изображения является минимизация числа ненулевых коэффициентов, получаемых в результате выполнения ДКП-3 над каждой областью разбиения. Формально эта задача записана

в виде выражения (1). Варьируемыми переменными данной задачи являются области разбиения $\{\mathbf{A}\}_i$, $i = 1, 2, ..., R$. При этом каждая $i$-я область разбиения определяется $i$-й строкой ($\mathbf{x}_i$) матрицы назначения $\mathbf{X}$, получаемой в результате решения задачи (8). Однако при решении задачи (1), как и задачи (8), неясно, из каких соображений выбирается параметр $R$, который определяет число областей разбиения.

Логичным способом выбора параметра $R$ может быть следующий.

Число областей разбиения исходного множества $\{\mathbf{A}\}$ удовлетворяет неравенству $1 \leqslant R \leqslant L$. Иными словами, параметр $R$ не может быть меньше единицы и не может превышать общего числа элементов исходного множества. Тогда задачу (1) можно решить $L$ раз:

$$\begin{aligned}
NZ_1 &= \sum_{i=1}^{R} f_{\text{ДКП-3}}(\{\mathbf{A}\}_i) \to \min_{\{\mathbf{A}\}_i,\ R=1};\\
NZ_2 &= \sum_{i=1}^{R} f_{\text{ДКП-3}}(\{\mathbf{A}\}_i) \to \min_{\{\mathbf{A}\}_i,\ R=2};\\
&\ldots\\
NZ_j &= \sum_{i=1}^{R} f_{\text{ДКП-3}}(\{\mathbf{A}\}_i) \to \min_{\{\mathbf{A}\}_i,\ R=j};\\
&\ldots\\
NZ_L &= \sum_{i=1}^{R} f_{\text{ДКП-3}}(\{\mathbf{A}\}_i) \to \min_{\{\mathbf{A}\}_i,\ R=L},
\end{aligned} \tag{9}$$

где $NZ_j$ — число ненулевых коэффициентов ДКП-3, выполняемого над $j$ областями разбиения исходного множества $\{\mathbf{A}\}$, т. е. решение задачи (1) при $R = j$.

Результат последовательного решения задачи (1) $L$ раз представим в виде вектора $\overrightarrow{NZ} = (NZ_1, NZ_2, ..., NZ_j, ..., NZ_L)$. Тогда параметр $R$ можно определить как индекс минимального элемента вектора $\overrightarrow{NZ}$, т. е.

$$R = \arg\min_{i \in \{1,2,...,L\}} (\overrightarrow{NZ}_i). \tag{10}$$

Решение каждой из задач (9) предполагает поиск оптимального разбиения и выполнения ДКП-3 над каждым разбиением. Выполнение ДКП-3 достаточно хорошо описано в работах [1, 9]. В свою очередь, оптимальное разбиение выполняется на основе решения задачи (8).

С учетом ограничений (2) и (3) задача (8) является задачей нелинейной целочисленной оптимизации. Рассмотрим решение данной задачи методом случайного поиска с наилучшей пробой [10].

Известные алгоритмы случайного поиска включают следующие этапы:

- выбор начальной точки поиска (начального приближения);
- выбор пробных точек вокруг текущей точки поиска;
- вычисление значений целевой функции в пробных точках;
- принятие решения о дальнейшем направлении поиска.

Рассмотрим реализацию данных этапов более подробно.

*Выбор начальной точки поиска.* В качестве начальной точки поиска будем использовать матрицу $\mathbf{X}$ размером $R \times L$ элементов, формируемую некоторым регулярным или случайным образом при условии выполнений ограничений (2) и (3).

Например, матрица $\mathbf{X}$ может быть сформирована следующим регулярным способом.

Сначала формируют нулевую матрицу размером $R \times L$ элементов:

$$x(i, j) = 0,\ \forall i = \overline{1, R};\ \forall j = \overline{1, L}.$$

Затем определяют позиции ненулевых элементов в соответствии с выражением

$$x(1 + j \bmod_R, j + 1) = 0,\ \forall j = \overline{0, L - 1}.$$

Так, для $R = 4$ и $L = 16$ матрица $\mathbf{X}$ будет иметь вид:

$$\mathbf{X} = \begin{bmatrix} 1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0\\ 0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&0\\ 0&0&1&0&0&0&1&0&0&0&1&0&0&0&1&0\\ 0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1 \end{bmatrix}_{4 \times 16}.$$

При случайной генерации начальной точки поиска, как и в предыдущем случае, сначала формируют нулевую матрицу размером $R \times L$ элементов. Затем для каждого столбца матрицы определяют позиции ненулевых элементов в соответствии с выражением

$$x(u, j) = 1,\ \forall j = \overline{1, L},$$

где $u = \text{rand}(1, R)$ — случайное целое число в диапазоне от 1 до $R$ включительно.

В качестве оператора rand(*) может выступать генератор случайных чисел в используемом языке программирования. Ниже показан пример формирования начальной точки поиска случайным образом:

$$\mathbf{X} = \begin{bmatrix} 0&0&1&0&0&1&0&0&0&0&1&0&0&0&0&1\\ 0&0&0&0&0&0&1&0&0&0&0&0&0&1&0&0\\ 1&0&0&0&1&0&0&1&0&0&0&0&0&0&1&0\\ 0&1&0&1&0&0&0&0&1&1&0&1&1&0&0&0 \end{bmatrix}_{4 \times 16}.$$

*Выбор пробных точек вокруг текущей точки поиска.* Пробные точки будем формировать вокруг текущей точки путем последовательного пробного присвоения единицы в каждую нулевую компоненту текущего столбца матрицы $\mathbf{X}$. Затем аналогичные присвоения единицы выполняются для нулевых компонент следующего столбца и т. д.

*Вычисление значения целевой функции в пробных точках.* Совершенно очевидно, что сложность решения задачи (8) определяется сложностью вычис-

ления значения целевой функции в пробной точке. Определим аналитические выражения и быстрый алгоритм расчета значения целевой функции в пробной точке.

Пусть некоторая матрица $\mathbf{X}$ есть текущая точка поиска, удовлетворяющая ограничениям (2) и (3), а $F(\mathbf{X})$ — значение целевой функции (8) в этой точке. Предположим, что в $j$-м столбце матрицы $\mathbf{X}$ в $t$-й позиции расположена единица, т. е. $x(t, j) = 1$. Определим пробную точку путем переназначения единицы из $t$-й позиции в $k$-ю текущего $j$-го столбца, т. е. путем выполнения присвоений:

$$x(t, j) = 0 \text{ и } x(k, j) = 1.$$

Значение целевой функции в задаче (8) представляет собой сумму $R$ компонент, причем значение каждой $i$-й компоненты определяется видом соответствующей $i$-й строки ($\mathbf{x}_i$) матрицы $\mathbf{X}$. Обозначим значение целевой функции в точке $\mathbf{X}$ как $F(\mathbf{X})$. Тогда $F(\mathbf{X})$ можно представить в виде следующей суммы:

$$F(\mathbf{X}) = F(\mathbf{x}_1) + F(\mathbf{x}_2) + ... + F(\mathbf{x}_i) + ... + F(\mathbf{x}_R), \quad (11)$$

где $F(\mathbf{x}_i) = \dfrac{\mathbf{x}_i\mathbf{A}^{\mathrm{T}}\mathbf{A}\mathbf{x}_i}{\mathbf{x}_i\mathbf{1}}$ — значение $i$-й компоненты, определяемое $i$-й строкой матрицы $\mathbf{X}$.

Поскольку пробная точка определяется путем переназначения единицы из $t$-й позиции в $k$-ю текущего $j$-го столбца матрицы $\mathbf{X}$, то это означает изменение ее $t$-й и $k$-й строк. Перепишем выражение (11) так, чтобы в правой части присутствовали $t$-я и $k$-я компоненты:

$$F(\mathbf{X}) = F(\mathbf{x}_1) + F(\mathbf{x}_2) + ... + F(\mathbf{x}_t) + \\ + ... + F(\mathbf{x}_k) + ... + F(\mathbf{x}_R). \quad (12)$$

Тогда, располагая значением целевой функции в точке $\mathbf{X}$, для того чтобы рассчитать значение целевой функции в пробной точке при переназначении единицы из $t$-й позиции в $k$-ю текущего $j$-го столбца, достаточно вычислить новые $t$-ю и $k$-ю компоненты в выражении (12).

Определим выражения для расчета этих компонент.

Отметим, что пробная точка характеризуется присвоением в $j$-й компоненте $t$-й строки нуля. С учетом того, что ранее данная компонента равнялась 1, то описанное присвоение можно описать выражением

$$\mathbf{x}_t^{0,j} = \mathbf{x}_t - \mathbf{e}_j, \quad (13)$$

где $\mathbf{x}_t^{0,j}$ — $t$-я строка матрицы $\mathbf{X}$ после присвоения 0 в $j$-ю компоненту; $\mathbf{e}_j$ — нулевой вектор размерности $1 \times L$ за исключением 1 в $j$-й компоненте.

Тогда значение $t$-го слагаемого целевой функции с учетом обнуления $j$-й компоненты $t$-й строки можно записать в виде

$$F(\mathbf{x}_t^{0,j}) = \frac{(\mathbf{x}_t - \mathbf{e}_j)\mathbf{A}^{\mathrm{T}}\mathbf{A}(\mathbf{x}_t^{\mathrm{T}} - \mathbf{e}_j^{\mathrm{T}})}{\mathbf{x}_t\mathbf{1} - 1} = \\ = \frac{\mathbf{x}_t\mathbf{D}\mathbf{x}_t^{\mathrm{T}} - 2\mathbf{e}_j\mathbf{D}\mathbf{x}_t^{\mathrm{T}} + \mathbf{e}_j\mathbf{D}\mathbf{e}_j^{\mathrm{T}}}{\mathbf{x}_t\mathbf{1} - 1}, \quad (14)$$

где $\mathbf{D} = \mathbf{A}^{\mathrm{T}}\mathbf{A}$.

По аналогии с выражением (13) назначение 1 в $j$-ю компоненту $k$-й строки матрицы $\mathbf{X}$ опишем в виде

$$\mathbf{x}_k^{1,j} = \mathbf{x}_k + \mathbf{e}_j, \quad (15)$$

где $\mathbf{x}_k^{1,j}$ — $k$-я строка матрицы $\mathbf{X}$ после присвоения 1 в $j$-ю компоненту. Тогда значение $k$-го слагаемого целевой функции с учетом присвоения 1 в $j$-ю компоненту $k$-й строки можно записать в виде

$$F(\mathbf{x}_k^{1,j}) = \frac{\mathbf{x}_k\mathbf{D}\mathbf{x}_k^{\mathrm{T}} + 2\mathbf{e}_j\mathbf{D}\mathbf{x}_k^{\mathrm{T}} + \mathbf{e}_j\mathbf{D}\mathbf{e}_j^{\mathrm{T}}}{\mathbf{x}_k\mathbf{1} + 1}. \quad (16)$$

Введем следующие обозначения:

$$\mathbf{u} = (u(1), u(2), ..., u(i), ..., u(R)), \text{ где } u(i) = \mathbf{x}_i\mathbf{D}\mathbf{x}_i^{\mathrm{T}}; \quad (17)$$

$$\mathbf{U} = \mathbf{D}\mathbf{X}^{\mathrm{T}}; \quad (18)$$

$$\mathbf{v} = (v(1), v(2), ..., v(i), ..., v(R)), \text{ где } v(i) = \mathbf{x}_i\mathbf{1}. \quad (19)$$

С учетом введенных обозначений (17)—(19) значение $t$-й и $k$-й компонент целевой функции можно вычислить согласно выражений

$$F(\mathbf{x}_t^{0,j}) = \frac{u(t) - 2\mathbf{U}(j, t) + \mathbf{D}(j, j)}{v(t) - 1}; \quad (20)$$

$$F(\mathbf{x}_k^{1,j}) = \frac{u(k) + 2\mathbf{U}(j, k) + \mathbf{D}(j, j)}{v(k) + 1}. \quad (21)$$

Обозначим $X_{ik}^j$ как пробную точку, получаемую путем переназначения единицы из $t$-й позиции в $k$-ю позицию $j$-го столбца матрицы $\mathbf{X}$. Тогда значение целевой функции в пробной точке запишем в виде суммы

$$F(\mathbf{x}_{ik}^j) = F(\mathbf{x}_1) + F(\mathbf{x}_2) + ... + F(\mathbf{x}_t^{0j}) + \\ + ... + F(\mathbf{x}_k^{1j}) + ... + F(\mathbf{x}_R), \quad (22)$$

где $F(\mathbf{x}_t^{0j})$ и $F(\mathbf{x}_k^{1j})$ рассчитаны в соответствии с выражениями (20) и (21), соответственно.

Поскольку $F(\mathbf{X})$ известно, значение целевой функции в пробной точке $\mathbf{x}_{ik}^j$ можно вычислить следующим образом:

$$F(\mathbf{x}_{ik}^j) = F(\mathbf{X}) - F(\mathbf{x}_t) - F(\mathbf{x}_k) + F(\mathbf{x}_t^{0j}) + F(\mathbf{x}_k^{1j}). \quad (23)$$

*Принятие решения о дальнейшем направлении поиска.* После вычисления значений целевых функций в пробных точках вокруг текущей точки поиска в качестве новой пробной точки в качестве новой текущей точки выбирается та, которая соответствует максимальному значению целевой функции, рассчитанному в соответствии с выражением (22). Данный процесс повторяется до тех пор, пока для текущей точки поиска не найдется ни одна пробная точка с большим значением целевой функции.

С учетом описанных этапов метода случайного поиска с наилучшей пробой определим следующий алгоритм решения задачи (8).

*Алгоритм кластеризации $L$ элементов размерности $P$, представленных в виде матрицы $\mathbf{A}_{P \times L}$, на $R$ непересекающихся подмножеств*

**Исходные данные:**

$\mathbf{A}_{P \times L}$ — исходное кластеризуемое множество из $L$ элементов размерности $P$; $R$ — число непересекающихся подмножеств (областей кластеризации).

**Выходные данные:**

$\mathbf{X}$ — матрица назначений размером $R \times L$ элементов, удовлетворяющая ограничениям (2) и (3) и определяющая распределение $L$ элементов по $R$ непересекающимся подмножествам.

**Выполняемые действия.**

*Шаг 1.* Сформировать текущую точку поиска в виде матрицы $\mathbf{X}$, удовлетворяющей ограничениям (2) и (3) (матрица $\mathbf{X}$ может быть сформирована регулярным или случайным способом).

*Шаг 2.* Вычислить $\mathbf{u}$, $\mathbf{v}$ в соответствии с выражениями (17)—(19), $\mathbf{D} = \mathbf{A}^{\mathrm{T}}\mathbf{A}$ и $F(\mathbf{x}_i) = \dfrac{u(i)}{v(i)}$, $i = 1, 2, ..., R$.

*Шаг 3.* Вычислить исходное значение целевой функции в текущей точке поиска $F(\mathbf{X}) = \sum\limits_{i=1}^{R} \dfrac{u(i)}{v(i)}$.

*Шаг 4.* Присвоить $FF = F(\mathbf{X})$.

*Шаг 5.* Положить $j = 1$, $k_{\mathrm{opt}} = 0$.

*Шаг 6.* Определить индекс $t$ — позицию единичного элемента в $j$-м столбце матрицы $\mathbf{X}$.

*Шаг 7.* Если $v(t) > 1$, то перейти к шагу 8, иначе — перейти к шагу 17 (данная проверка выполняется с целью того, чтобы в процессе поиска в матрице $\mathbf{X}$ не образовывались нулевые строки).

*Шаг 8.* Вычислить $F(\mathbf{x}_t^{0,j})$ в соответствии с выражением (20).

*Шаг 9.* Положить $k = 1$.

*Шаг 10.* Если $k \neq t$, то перейти к шагу 11, иначе — перейти к шагу 14.

*Шаг 11.* Вычислить $F(\mathbf{x}_k^{1,j})$.

*Шаг 12.* Вычислить $F(\mathbf{x}_{ik}^{j})$ в соответствии с выражением (23).

*Шаг 13.* Если $F(\mathbf{x}_{ik}^{j}) > FF$, $FF = F(\mathbf{x}_{ik}^{j})$ и $k_{\mathrm{opt}} = k$, иначе — перейти к шагу 14.

*Шаг 14.* Положить $k = k + 1$.

*Шаг 15.* Если $k > R$, то перейти к шагу 16, иначе — перейти к шагу 10.

*Шаг 16.* Если $k_{\mathrm{opt}} \neq 0$, то выполнить:

— вычисление новых вспомогательных значений для быстрого расчета целевой функции в пробных точках:

$$u(t) = u(t) - 2\mathbf{U}(j, t) + \mathbf{D}(j, j);$$
$$u(k_{\mathrm{opt}}) = u(k_{\mathrm{opt}}) + 2\mathbf{U}(j, k_{\mathrm{opt}}) + \mathbf{D}(j, j);$$
$$v(t) = v(t) - 1;\ v(k_{\mathrm{opt}}) = v(k_{\mathrm{opt}}) + 1;$$

$$\mathbf{U}(:, k_{\mathrm{opt}}) = \mathbf{D}\mathbf{x}_{k_{\mathrm{opt}}}^{\mathrm{T}} —$$

пересчет $k_{\mathrm{opt}}$-го столбца матрицы $\mathbf{U}$;

— расчет новых значений компонент целевой функции:

$$F(\mathbf{x}_t) = \frac{u(t)}{v(t)};\ F(\mathbf{x}_{k_{\mathrm{opt}}}) = \frac{u(k_{\mathrm{opt}})}{v(k_{\mathrm{opt}})};$$

— вычитание значений компонент целевой функции и переопределение текущей точки поиска:

$$F(\mathbf{X}) = F(\mathbf{X}) - F(\mathbf{x}_t) - F(\mathbf{x}_{k_{\mathrm{opt}}});$$

$$\mathbf{x}_t = \mathbf{x}_t - \mathbf{e}_j;\ \mathbf{x}_{k_{\mathrm{opt}}} = \mathbf{x}_{k_{\mathrm{opt}}} + \mathbf{e}_j;$$

— запоминание нового значения целевой функции

$$FF = F(\mathbf{X}).$$

*Шаг 17.* Положить $j = j + 1$.

*Шаг 18.* Если $j \leqslant L$, то $k_{\mathrm{opt}} = 0$ и перейти к шагу 6, иначе — перейти к шагу 19.

*Шаг 19.* Если $k_{\mathrm{opt}} \neq 0$, перейти к шагу 5, иначе — выход.

Проведен сравнительный анализ предложенного алгоритма кластеризации с наиболее известным алгоритмом $k$-средних. Анализ проводился по двум критериям: средней квадратической ошибки кластеризации и вычислительной сложности. В ходе экспериментальных исследований установлено, что при сопоставимой сложности предложенный алгоритм характеризуется меньшей, на 1...2 дБ, ошибкой кластеризации по сравнению с алгоритмом $k$-средних.

На рис. 2 и 3 представлены результаты решения задач (1), (8) и (10) при сжатии тестового подвижного изображения "Foreman".

На рис. 2 по оси абсцисс показаны номера обрабатываемых трехмерных кубов изображения. Размер куба выбран $8 \times 8 \times 16$ ($P = 8$, $N = 8$, $L = 16$). При линейных размерах кадров подвижного изображения $576 \times 720$ пикселей число обрабатываемых кубов составило 6480. По оси ординат показано, на сколько уменьшается число ненулевых коэффициентов ДКП-3 при оптимальном разбиении кодируемого трехмерного куба по сравнению со случаем, когда кодируемый куб рассматривается как одна область: $\Delta NZ = NZ_1 - NZ_R$, где параметр $R$ найден

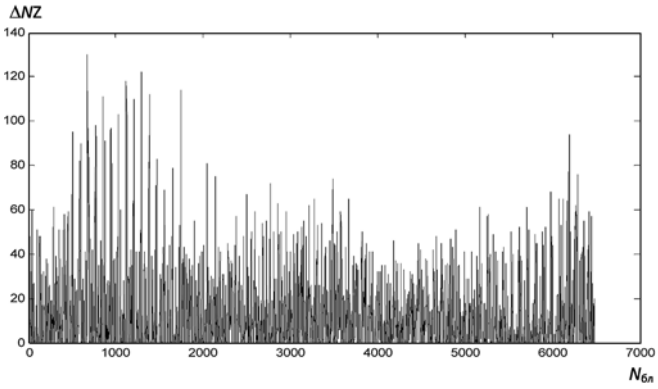


**Рис. 2. Уменьшение числа ненулевых коэффициентов ДКП-3 для каждого трехмерного блока изображения при оптимальном разбиении**

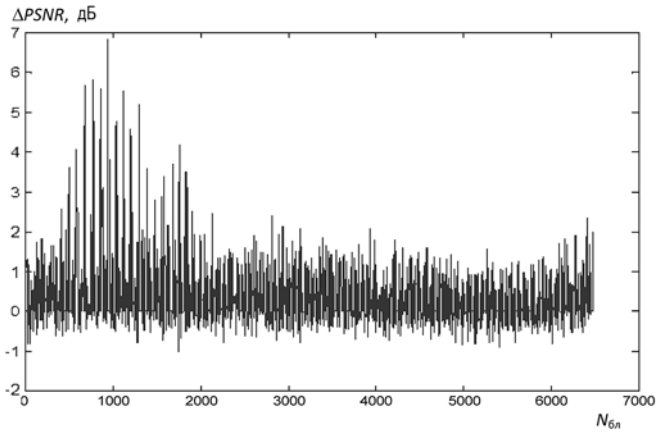


**Рис. 3. Увеличение *PSNR* за счет оптимального разбиения кодируемых фрагментов подвижного изображения**

на основе решения задачи (10). На рис. 1 видно, что максимальное уменьшение числа ненулевых коэффициентов для обрабатываемого куба за счет оптимального разбиения составило 130. Суммарное значение, на которое уменьшилось число значимых коэффициентов ДКП-3, составило 70 133, что соответствует уменьшению общего числа коэффициентов в 1,5 раза.

Уменьшение значимых коэффициентов в силу ортогональности ДКП-3 означает, что большая часть энергии изображения (под энергией здесь понимается сумма квадратов значений пикселей кодируемого фрагмента) сосредотачивается в меньшем числе коэффициентов. Это, в свою очередь, приводит к меньшим искажениям при квантовании коэффициентов ДКП-3. Следовательно, уменьшение значимых коэффициентов на основе оптимального разбиения приводит и к увеличению качества восстановленных изображений. Данное обстоятельство подтверждается рис. 3.

Как и в предыдущем случае, по оси абсцисс показаны номера кодируемых трехмерных кубов исходного подвижного изображения. По оси ординат показано значение увеличения *PSNR* восстановленных фрагментов при их оптимальном разбиении относительно случая, когда каждый куб кодируется как одна область. Максимальное возрастание качества на один фрагмент для данного примера составило почти 7 дБ. Суммарное повышение качества для всего обрабатываемого изображения размером 576 × 720 × 16 пикселей составило около 1,43 дБ.

Таким образом, компенсация движения подвижных изображений на основе оптимальных разбиений приводит не только к увеличению коэффициента сжатия, но и к улучшению качества восстановленных изображений.

* * *

В работе развит известный подход к устранению межкадровой избыточности подвижных изображений на основе оптимальных разбиений на непересекающиеся подмножества. Развитие подхода состоит в постановке задачи поиска оптимального разбиения. Сформулирована задача определения оптимального числа областей разбиения, приводящего к минимизации значимых коэффициентов ДКП-3. Предложен алгоритм решения поставленной оптимизационной задачи о разбиениях на основе метода случайного поиска с наилучшей пробой. Приведены результаты моделирования, которые подтверждают эффективность предложенных решений. Экспериментально показано, что оптимальное разбиение кодируемых фрагментов трехмерного изображения по временно́й оси приводит не только к увеличению сжатия, но и к повышению качества восстановленных изображений.

**Список литературы**


1. **Беляев Е. А., Сухов Т. М., Шостацкий Н. Н.** Сжатие видеоинформации на основе трехмерного дискретного псевдо-косинусного преобразования для энергоэффективных систем видеонаблюдения // Компьютерная оптика. 2010. Т. 34, № 2. С. 260—272.
2. **Sgouros N. P., Athineos S. S., Mardaki P. E.** etc. al. Use of an Adaptive 3D-DCT Scheme for Coding Multiview Stereo Images // IEEE Proceedings of ISSPIT. 2005. P. 180—185.
3. **Chan Y. L., Siu W. C.** Variable Temporal-length 3-D Discrete Cosine Transform Coding // IEEE Transactions on Image Processing. 1997. Vol. 6, N 5. P. 758—763.
4. **Koivusaari J. J., Takala J. H.** Simplified Three-Dimensional Discrete Cosine Transform Based Video Codec // SPIE Proc. of Multimedia on Mobile Devices. 2005. P. 11—21.
5. **Dugad R., Ahuja N.** A Fast Scheme for Image Size Change in the Compressed Domain // IEEE Transactions on Circuits and Systems for Video Technology. 2001. P. 461—474.
6. **Koivusaari J. J., Takala J. H., Gabbouj M.** Image Coding Using Adaptive Resizing in the Block-DCT Domain // SPIE Proc. of Multimedia on Mobile Devices II. 2006. P. 1—9.
7. **Takala J., Gabbouj M., Chen H.** Use of Adaptive Resizing in 3-D DCT Domain for Video Coding // Picture Coding Symp., Lisbon, Portugal, 2007, Nov. P. 7—9.
8. **Dai Q., Chen X., Lin C.** Fast Algorithms for Multidimensional DCT-to-DCT Computation between a Block and Its Associated Subblocks // IEEE Transactions on Circuits and Syst. for Video Tech., 2003. Vol. 13, N 7. P. 717—725.
9. **Ахмед Н., Рао К. Р.** Ортогональные преобразования при обработке цифровых сигналов. М.: Связь, 1980. 248 с.
10. **Ермольев Ю. М.** Методы стохастического программирования. М.: Наука, 1976. 237 с.

УДК 003.26.004.7.004.9

**Н. В. Чичварин,** канд. техн. наук, доц., e-mail: genrich.gertz@gmail.com,
МГТУ им. Н. Э. Баумана

# Выбор методов защиты проектной документации от несанкционированного доступа

*Основой публикации являются результаты проведенных исследований по выбору метода и средств защиты проектной документации, продуцируемой в САПР, от несанкционированного доступа. Рассмотрена модель угроз информационной безопасности. Приведены результаты анализа возможностей криптографии, компьютерной и цифровой стеганографии. Рассмотрены методы компьютерной стеганографии и предложено их применение для хранения архивных данных в информационной подсистеме САПР. При анализе также учтены существенные особенности структуры проектной документации, определяющей выбор, либо синтез того или иного стеганографического алгоритма.*

***Ключевые слова:*** *защита проектной документации, несанкционированный доступ, модель угроз информационной безопасности, компьютерная стеганография, САПР, стеганографический алгоритм*

**N. V. Chichvarin**

# The Choice of Methods of Protection Design Documents from Unauthorized Access

*The basis of publications are the results of research on the choice of the method and means to protect the design documentation from unauthorized access. A model of the information security threats. The results of the analysis of cryptographic features, computed and digital steganography. The methods and computer steganography suggested their use for archival storage of archival data in the information subsystem of CAD. In the analysis also takes into account the essential features of the structure of the project documentation, defining the selection or synthesis of a steganographic algorithm.*

***Keywords:*** *protection of the design documentation, unauthorized access, the model of information security threats, computer steganography, CAD, steganographic algorithm*

## Введение

Основой публикации являются результаты проведенных исследований по разработке методов и средств стеганорафического шифрования проектной документации, продуцируемой в системах автоматизированного проектирования (САПР). Материалы публикации изложены в цикле из двух статей. В настоящей статье на основе анализа характера документации, разрабатываемой на каждом иерархическом уровне САПР, а также форматов проектных данных для наиболее распространенных САПР разного назначения делается выбор наиболее эффективного метода защиты документации. Обоснован вывод о возможности разработки метода, основанного на сокрытии текстовых спецификаций в изображениях схем и чертежей, приведенных к единообразному формату.

В настоящее время вопросам безопасности информационных систем уделяется нарастающее внимание. Указ президента Путина В.В. "О концепции национальной безопасности РФ" от 10.01.2000 г. определяет необходимость дальнейших разработок в этом направлении. Если в области безопасности информационных систем уже накоплен большой теоретический и практический опыт, то в области информационной безопасности (ИБ) САПР специальных комплексных разработок не ведется. Проведенный анализ различных публикаций показывает, что такие информационные системы, как САПР, рассматриваются российскими и иностранными специалистами в области информационной безопасности без учета многих специфичных факторов.

В публикации под САПР понимаются системы, обозначаемые за рубежом аббревиатурами CAD/

CAM/CAE, CAD/CAM/CAPP, CAD/CAM/PDM, PLM. Большинство средств автоматизированного проектирования, применяемых в отечественных проектных организациях, являются зарубежными продуктами. Несмотря на существование эффективных структур сертификации (ФСТЭК и т. п.), обеспечивающих выявление недекларированных возможностей (НДВ) программных продуктов, защиту от программных и аппаратных закладок, а также пресечение несанкционированного доступа (НСД) к корпоративной информации, специфичные особенности ИБ САПР пока учитываются не в полной мере. Как показал анализ публикаций [1—5], проблемы ИБ САПР нарастают и за рубежом. Известно, что у американской аэрокосмической корпорации Lockheed Martin в 1997 г. была совершена кража электронных чертежей и информации о конструкции самолета-невидимки Stealth. Еще в 2000 г. был обнаружен первый вирус ACAD.Star для программного обеспечения CAD/CAM — AutoCAD. До настоящего времени ни в России, ни за рубежом не найдены методы и средства обеспечения ИБ проекта и объекта проектирования в период всего жизненного цикла объекта (т. е. в условиях применения CALS-технологий). Нет и фундаментальных теоретических исследований в этой области. Существующие системы защиты информации (СЗИ) от НСД (например, "Страж NT", "Secret Net", "Security Studio", "Панцирь-К" и т. п.) в полной мере специфику ИБ САПР не учитывают.

Задача защиты проектной документации от несанкционированного доступа условно может быть разделена на две: первая — защита документации, составляющей коммерческую тайну; вторая — защита документации, составляющей государственную тайну. Вторая задача зачастую решается путем использования вычислительных средств, оснащенных сертифицированными программно-аппаратными комплексами, прошедших аттестацию. Однако использование криптографических методов значительно усложняет процесс проектирования и не всегда допустимо с точки зрения действующего законодательства. Исследования методов сокрытия чертежей и изображений объекта проектирования становятся все более необходимыми. Это особенно важно для САПР, реализующих CALS-технологии.

Централизация всей проектной документации, большое число как внутренних, так и внешних пользователей систем CALS, особенно на заключительном этапе цикла — эксплуатационном, требуют особого внимания к задаче обеспечения ИБ.

В условиях замедления темпов экономического роста есть соблазн сократить расходы на ИБ в рамках общего сокращения издержек. Однако, согласно отчету KPMG [1], большинство организаций планируют увеличить бюджеты на ИБ в следующем году, что доказывает возросшее внимание за последние несколько лет к управлению информационными рисками.

Важная роль в решении задач ИБ отводится международному стандарту ISO17799. Российские предприятия руководствуются также "РД Гостехкомиссии России: Автоматизированные системы. Защита от НСД к информации. Классификация АС и требования по защите информации" и другими документами, рассмотренными, в частности, в работах [2, 3].

Результат внедрения стандарта ISO17799 — система менеджмента ИБ [4, 5]. Ее цель — сокращение потерь, связанных с нарушением ИБ. В некоторых случаях масштаб потерь может быть таким, что грозит предприятию банкротством. Примером невосполнимой потери может служить CALS-проект высокотехнологичного изделия с длительным циклом разработки и поддержки в эксплуатации.

Говоря о практическом применении стандарта ISO17799, следует иметь в виду три обстоятельства, затрудняющие его непосредственное использование:

- рекомендации стандарта в ряде случаев являются весьма общими;
- в организации, как правило, уже существует определенная система процессов, в которую необходимо интегрировать процесс управления безопасностью;
- реализации стандарта ISO17799 характеризуются определенной статичностью, что не позволяет им достаточно оперативно следовать постоянным изменениям в информационных технологиях (ИТ).

В последнее время значительное внимание уделяется развитию практики интеграции процесса управления безопасностью Security Management [3] в систему процессов службы ИТ. Такие процессы являются ITIL-процессами, т. е. отвечают рекомендациям стандарта IT Infrastructure Library, разработанного ССТА (Central Computer and Telecommunications Agency, UK). Применение концепции ITIL, как показано в работе [5], может служить основой стратегии развития высокотехнологичного предприятия. Переходя от стратегии на уровень процессов, необходимо отметить целесообразность реализации CALS-проектов на основе ITIL-процессов.

Гибкая и адаптируемая в действующих системах подсистема ИБ САПР будет наиболее полезна для обеспечения коммерческой тайны, сокрытия "ноу-хау" путем защиты от НСД проектной и эксплуатационной документации.

В работе проведен теоретико-экспериментальный анализ возможностей известных стеганографических алгоритмов, предназначенных для сокрытия данных в различных контейнерах применительно к защите проектной документации. Использованы программный инструментарий и результаты численных экспериментов, проведенных студентами МГТУ им. Н. Э. Баумана под руководством автора на-

стоящей публикации [6—16]. При этом учтены результаты, полученные в работах [17—19].

Как показывает проведенный в работах [7, 15, 16] анализ, специфика проектной документации заключается в следующем:

- документация всегда строго структурирована в соответствии с требованиями ГОСТ;
- основные компоненты структуры — это та или иная схема (чертеж) и спецификация;
- первая компонента — всегда графическая, вторая — текстовая, изложенная в соответствии с требованиями ГОСТ языком деловой прозы;
- каждый компонент идентифицируется своим децимальным номером, определяющим тип документа, его назначение и место хранения в архиве держателя проектной документации;
- для реализации CALS-технологий необходима передача проектной документации по каналам связи, включая открытые;
- содержание документов может составлять государственную либо коммерческую тайну.

### Анализ модели угроз ИБ САПР

Перечень специфических угроз, характерных для САПР, дополняется перечнем, характерным для любой информационной системы (ИС). Известная обобщенная схема алгоритма построения модели угроз ИБ в САПР, в которой выделены угрозы целостности и конфиденциальность информации, представлена на рис. 1 [2].

Модель, построенная в соответствии с обобщенной схемой алгоритма (рис. 1), учитывает защиту проектной документации во время жизненного цикла объекта проектирования, предусматривая анализ возможности атак на каналы передачи проектной документации.

ИБ внутренних каналов и аппаратных средств обеспечивается средствами защиты от НСД, таких как "Страж-NT", "Панцирь" и т. п. По внешним каналам происходит обмен проектной и эксплуатационной документациями при сопровождении объекта проектирования в период всего жизненного цикла. В настоящее время вопросам защиты проектной документации от НСД при передаче по различным каналам (особенно открытым) внимание практически не уделяется. Для решения этих задач возможно применение различных средств. Среди таких средств существенными можно признать шифрование программной документации. Для выбора возможных прототипов и разработки соответствующих средств необходимо проанализировать форматы данных, продуцируемых САПР.

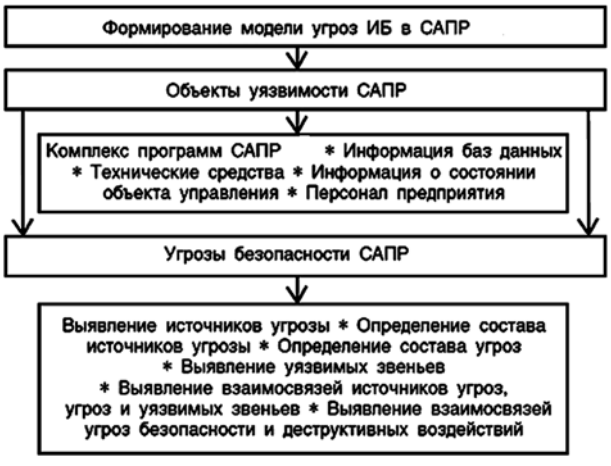


**Рис. 1. Алгоритм построения модели угроз информационной безопасности САПР**

### Анализ форматов данных, составляющих проектную документацию

В соответствии с блочно-иерархическим подходом введем описание структуры САПР обобщенного объекта проектирования, включающее пять уровней (рис. 2, 3): архитектурный, функционально-логический, системотехнический, схемотехнический, физический.

Для каждого из данных уровней необходимо рассматривать круг требующих решения задач, определять набор реализуемых на нем функций, проводить анализ надежности и полноты обеспечения поставленных целей.

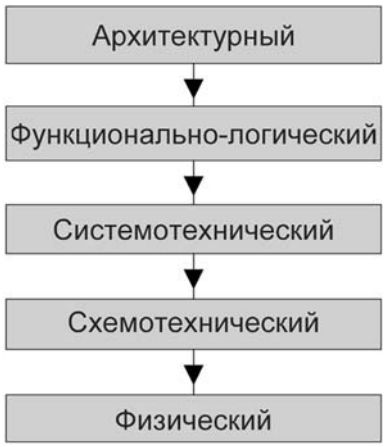


**Рис. 2. Иерархическая схема САПР обобщенного объекта проектирования**

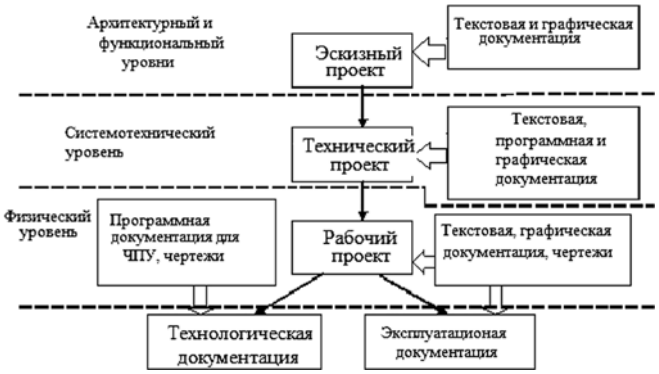


**Рис. 3. Виды проектной документации**

Функционал архитектурного уровня соответствует степени детализации объекта проектирования, не требующей учета:

- физической природы элементов объекта проектирования логического носителя сигнала;
- внутренней структуры подсистем.

На архитектурном уровне иерархической структуры рассматриваются модели топологий обобщенного объекта проектирования, правила и условия их построения. Для многих конкретных объектов этот уровень отсутствует.

Модели функционально-логического уровня строятся для решения задач согласования подсистем (функционально полных узлов), входящих в состав объекта проектирования.

Системотехнический уровень не учитывает физических свойств объекта и соответствует степени детализации в приближении моделей "черный ящик" или "серый ящик". Подсистемы данного уровня выполняют функции проектирования объекта в целом и согласования отдельных компонентов каждой подсистемы.

На схемотехническом уровне в модельном представлении объекта проектирования учитывается физическая природа компонентов объекта проектирования совместно с характером преобразования в отдельных моделях типа "черный ящик" (принципиальные схемы).

На физическом уровне иерархической структуры выполняются работы по рабочему проектированию и технологической подготовки производства.

Проведен анализ форматов данных основных программных средств (ПС) САПР, приведенных ниже. ПС САПР можно классифицировать на ПС сквозного проектирования, ПС инженерных расчетов и ПС схемотехнического проектирования.

*Средства сквозного (диакоптического) проектирования, сопровождающие проектные работы на всех уровнях:*

- MicroStation — универсальная САПР. Основа программных решений для: ГИС, геодезии, картографии, земельного кадастра, инженерных сетей, проектирования электроники, архитектуры, строительства мостов, автодорог, зданий и сооружений, проектирования промышленных предприятий и заводов машиностроения, дизайна интерьеров. Основные форматы данных: DGN и DWG.
- Pro/Engineer — универсальная САПР для промышленных компаний.
- CADDS5 — мощный машиностроительный пакет САПР.
- CATIA — универсальная САПР.
- SolidWorks — универсальная САПР.
- КОМПАС, КОМПАС-График — мощные машиностроительные пакеты САПР.
- AutoCAD — машиностроительный пакет САПР.

*Средства инженерных расчетов (архитектурный, функционально-логический уровень, системотехнический уровни):*

- LabVIEW — САПР для создания виртуальных приборов, может работать в составе систем промышленной автоматизации.
- ANSIS, NASTRAN — вычислительные САПР (CAE), программные комплексы в области конечно-элементного анализа (FEM).
- MathCAD — САПР для научно-технических расчетов, ориентированная на подготовку интерактивных документов с вычислениями и визуальным сопровождением.
- MATLAB — САПР для решения задач выполнения технических вычислений, имеет одноименный язык программирования.
- Mathematica — САПР для научно-технических исследований, символьных вычислений, визуализации данных, решения различных прикладных задач с множеством других возможностей.
- Euler Math Toolbox — САПР для инженерных и научных расчетов (система компьютерной математики).
- GNU Octave — САПР для инженерных и научных расчетов, система для математических вычислений, использующая совместимый с MATLAB язык высокого уровня.
- R — свободная программная среда вычислений с открытым исходным кодом в рамках проекта GNU.
- Maxima — открытая система компьютерной алгебры.
- Euler Math Toolbox — численный пакет с открытым исходным кодом, использует матричный язык, подобный MATLAB.
- Maple — единственная система аналитических вычислений.
- SALOME — программно-методический комплекс, поддерживающий взаимодействие между САПР CAD моделирования и вычислительной САПР CAE программным обеспечением, объединяет множество CAE решателей.

*Средства схемотехнического проектирования:*

- Варианты САПР типа Spice: ICAP/4Window (Intusoft), Saber Mixed-technology Simulator (фирмы Analogy), Viewanalog (Viewlogic Systems), Continuum (Mentor Graphics), AVOCAD, ПА9 (МГТУ им. Н. Э. Баумана).
- Программные комплексы семейства Omega-PLUS, Microwave Office, Omega PLUS, P-CAD, P-CAD Версия Accel EDA 15.0, Protel, OrCAD, SPECCTR (с поддержкой текстовых форматов DXF и PDIF), PSpice.

В табл. 1 приведены форматы данных в основных средствах проектирования. Учтены наиболее распространенные средства, которые применяются в зарубежных и отечественных разработках на раз-

Таблица 1

**Форматы данных, используемые в системах CAD/CAM**

| | |
|---|---|
| ADM | Файлы внутреннего формата САПР ADEM |
| ASM | Файлы сборок Pro/ENGINEER и SolidEdge |
| APTSOURC | Данные, генерируемые в CATIA для программ создания постпроцессоров для станков с ЧПУ (к примеру GPost) |
| CAD | Формат CAD/CAM системы Tebis |
| CADAM | Файлы принадлежат Dassault Systems. Чертежные файлы CADAM имеют расширение CDD |
| CADDS 5 | Файлы принадлежат Parametric Technology Corporation (PTC). Чертежные файлы имеют расширение CADDS |
| CADIF | Формат данных, разработанный фирмой Zuken для своей САПР электронных компонентов Visula на основании формата EDIF |
| CATIA | Файлы данных имеют следующие расширения: CATIA, CATMATERIAL, CATPART, CATPROCESS, CATPRODUCT, CATSHAPE, CATSWL, CATSYSTEM, CATDATA |
| CFG | Расширение, применяемое для файлов с конфигурационным настройками программ (к примеру, в P-CAD 8.x используется для загрузки схем проектов, располагаемых на нескольких листах) |
| DGM | Диаграммы Pro/Engineer |
| DFT | Файл чертежа САПР SolidEdge |
| DGM | Диаграммы Pro/EngineerDGN — чертежный формат САПР Bentley MicroStationDITA |
| DWF | Формат публикации чертежных данных AutoCAD и других программ Autodesk, оптимизированный для WEB |
| DWG | Внутренний формат чертежных данных AutoCAD и других программ Autodesk |
| DXB | Бинарный формат обмена чертежными данными AutoCAD и других программ Autodesk |
| DXF | Формат обмена чертежными данными AutoCAD и других программ Autodesk |
| ECAD (IDF) | Общее название для формата IDF, предназначенного для обмена данными между различными системами проектирования электронных компонентов (ECAD) |
| IDF (2.0, 3.0 и 4.0). IDF 2.0 и 3.0 | Используются в паре. В первом хранится информация о разводке печатной платы, а во втором — об электронных компонентах.IDF 4.0 содержит дополнительную информацию, помимо доступной в IDF 2.0 и 3.0 |
| EDIF | Универсальный формат для обмена схематическими данными электронных проектов ELT — Cimatron EEMF — графический растровый 32-битный формат. Развитие формата WMF |
| EMP | Файл описания печатной платы в IDF-формате |
| EMN | Файл описания детали в IDF-формате |
| EPF | Файлы деталей учебной версии EdgeCAM. Файлы деталей полнофункциональной версии имеют расширение PPF |
| EPS | Формат для отображения графических файлов, основанный на языке PostScript |
| EXP | Файл экспорта CATIA.EXPRESS — язык моделирования данных в STEP. F |
| FRG | Файлы фрагментов чертежей в САПР T-Flex |
| FRM | Форматки Pro/Engineer |

*Продолжение табл. 1*

| | |
|---|---|
| FRW | Файлы фрагментов чертежей в САПР КОМПАС |
| FVT | Файл в формате FlowVision, мощного пакета моделирования аэро- и гидродинамических процессов |
| I-DEAS | САПР, разработанная корпорацией Structural Dynamics Research, в настоящее время продан компании Unigrafics и является частью предлагаемого ей пакета NX. Файлы формата I-DEAS имеют расширение UNV |
| IDW | Файлы чертежей Autodesk Inventor |
| MDB | Файлы баз данных MS-Access. Также расширение файлов с информацией о моделировании в Pro/MECHANICA |
| MESH | Файлы графического движка OGRE, также термин применяется для обозначения каркасных моделей в различных САПР (Solidworks, Solid Edge, NX |
| MFG | Файлы, описывающие процесс производства детали (к примеру, на станке с ЧПУ) в САПР Pro/ENGINEER |
| MOD | Файл модели САПР CATIA и CADdy++ Mechanical |
| MPR | Файлы Mechanical Desktop с результатами расчета характеристик детали или сборки в соответствии с назначенными на ее составляющие материалами |
| MRK | Файл заметки для объекта Pro/EngineerMTS |
| RPCIII | Файлы данных программ анализа усталостных разрушений, долговечности и ресурса, разработанных компанией MTS Systems, файлы имеют расширение RSP |
| NC | Расширение файлов с программным кодом для станков с ЧПУ |
| NSH | Используется в областях CAD/CAM/CAE и является частью множества индустриальных стандартов, таких как IGES, STEP и ACIS |
| PAR | Файлы деталей в САПР SolidEdge.Parasolid — ядро системы геометрического моделирования, в настоящее время используемое в таких САПР, как Unigraphics, SolidWorks, SolidEdge, Powershape, MasterCAM и др. |
| PLA | Формат архивных проектов архитектурной САПР Archicad (содержащих все используемые в проекте данные) |
| PPF | Внутренний формат EdgeCAM, программы для автоматизированного создания управляющих программ для станков с ЧПУ |
| PRN | Модуль визуализации моделей в Pro/ENGINEERProductView — просмотровщик различных форматов файлов от PTC |
| PRT | Расширение файлов деталей и компонентов в CADkey, PCAD, Unigraphics, Pro/Engineer |
| PSM | Файлы чертежей листовых деталей в САПР SolidEdge |
| SCH | Файлы схем в P-CAD, PSpice, OrCAD, EAGLE Layout Editor |
| SEC | Эскизы Pro/Engineer SET — установочные шаблоны листов в MasterCAM (ПО для станков с ЧПУ) |
| SLDASM | Расширение файлов сборок Solidworks.SLDPRT — расширение файлов деталей Solidworks |
| SRF | Файлы в формате Surfer, программы для построения поверхностей по трехмерным матрицам |

*Окончание табл. 1*

| | |
|---|---|
| SSL | Файлы библиотек символов Corel Flowchart |
| STD | Файл карты слоев печатной платы в PCAD |
| PRO -TSH | Формат обмена данными между Pro/ENGINEER и STHENO |
| STL | Формат стереолитографической САПР, созданной 3D System. В настоящее время поддерживается большинством ПО быстрого прототипирования и производства. Формат описывает только поверхность трехмерных объектов, не учитывая цвет, текстуру и другие общие для CAD-моделей параметры. Спецификация STL предусматривает как текстовое, так и бинарное представление |
| STP | Применяется в CADKEY для экспорта данных в Pro/Engineer |
| TAP | Файлы редактора Geopath — ПО для создания управляющих программ для станков с ЧПУ |
| UNV | Расширение файлов системы I-DEAS |
| PRX | Файлы проектов в Primavera Project Planner — ПО для планирования, управления и контроля, также расширение скомпилированных программ в Foxpro |
| VCD | Формат чертежей VisualCADDVDA — расширение файлов формата VDA-FS (интерфейс для технических характеристик элементов и точек, установленных союзом немецких автомобилестроителей) |
| VIS | Файлы определения слоев в E3.Series (САПР для электротехники и АСУ ТП) |

личных иерархических уровнях. В табл. 2 приведены форматы данных, применяемых в ГИС.

Проведенный анализ форматов данных в САПР позволяет сделать следующие промежуточные выводы:

- на верхних иерархических уровнях проектная документация представляется в текстовых и растровых форматах;
- на схемотехническом уровне проектная документация представляется в векторных форматах (принципиальные схемы) и текстах: описание, данные о конструктивных параметрах, микропрограммы для "прошивки" ПЛИС, микроконтроллеров, однокристальных ЭВМ;
- на физическом уровне проектная документация представляется в виде чертежей (векторные форматы) и спецификаций (текстовая документация).

## Особенности форматов файлов компьютерной графики с учетом задачи защиты проектной документации от несанкционированного доступа

**Растровая графика.** Основным (наименьшим) элементом растрового изображения является точка. Если изображение экранное, то эта точка называется пикселем. Каждый пиксель растрового изображения имеет свойства: размещение и цвет. Недостаток растровых изображений связан с невозможностью их увеличения для рассмотрения деталей. Файлы с проектной документацией в растровых

Таблица 2

**Форматы данных, применяемых в ГИС**

| | |
|---|---|
| TDEF | Формат обмена данными между геодезическим ПО TIF, TIFF — формат для хранения изображений, таких как фотографии и штриховые иллюстрации. Поддерживается большей частью ПО для работы с графикой, которое доступно в настоящее время. Среди отличительных особенностей формата — поддержка нескольких изображений в одном файле и возможность использования LZW-компрессии |
| TIFF | Формат для хранения изображений, таких как фотографии и штриховые иллюстрации. Поддержка нескольких изображений в одном файле и возможность использования LZW-компрессии |
| 3D | Общее наименование для файлов трехмерных векторных данных различных форматов |
| 3DM | Расширение файлов моделей, созданных в Rhino 3D-пакете трехмерного моделирования с помощью NURBS (Non-Uniform Rational B-Splines) |
| 3DS | Формат данных программы трехмерной визуализации и анимации Autodesk 3D Studio MAX/VIZ.3TA5 — файлы данных, создаваемых геодезическим инструментом УОМ3 |
| GEO | Файлы GEOSTAR (файлы модуля пакета COSMOS/M, включающий графический 3D-построитель, пред- и постпроцессор для CAE-анализа) |
| GeoTiff | Стандарт для хранения географических ссылочных графических данных. |
| IT | Файл топографических данных программы ГИС VisIT |
| JOB | Файлы в формате, создаваемом геодезическим инструментом Geodimeter |
| JSM | Файлы программы визуализации Jig ShapeMap |
| R4, R5 | Формат данных, создаваемый геодезическим инструментом Zeiss/Trimble |
| RGS | ПО для решения геодезических задач. Последние версии носят название GeoniCS Изыскания |
| SDAI | В настоящее время существуют привязки SDAI к C/C++, Java.SDE (Spatial Database Engine). Применяется в ГИС Arc/Info |
| SDF | Spatial Data File — файлы пространственных данных, внутренний формат Autodesk MapGuide |
| SDR20, SDR33 | Формат данных, создаваемый геодезическим инструментом Sokkia |
| TOP | Расширение чертежей Topocad, ПО для землеустройства, картографии и ГИС |
| XYZ | Текстовый формат, представляющий собой разделенный запятыми список X-, Y- и Z-координат точек в пространстве. DBG — расширение файлов с сообщениями отладки в ArcView |
| DEM | Расширение файлов цифровых моделей высот в ArcView |
| DRW | Формат чертежных файлов САПР Micrografx. Развитием Micrografx в настоящее время является система Corel DESIGNER. Чертежи САПР Pro/Engineer |
| DTM | Расширение файлов цифровых моделей рельефа |
| GenCA | ASCII-формат, общий для индустрии электронных компонентов. Файлы в формате GenCAD имеют расширение CAD |
| Gerber ART | Файлы в формате Gerber могут иметь расширения GBR, GBX, PHD, SPL или Geosoft Grid-файлы |

форматах продуцируются и используются в ГИС, геодезическо-архитектурных САПР. Для остальных САПР файлы растрового формата применяются при передаче по коммуникационным каналам.

**Векторная графика.** Так же как в растровой графике основным элементом изображения является точка, в векторной графике основным элементом изображения является линия (при этом не важно, прямая это линия или кривая). Как известно, в растровой графике тоже существуют линии, но там они рассматриваются как комбинации точек. Для каждой точки линии в растровой графике отводится одна или несколько ячеек памяти (чем больше цветов могут иметь точки, тем больше ячеек им выделяется). В векторной графике объем памяти для хранения линии не зависит от размеров линии, поскольку линия представляется в виде формулы, а точнее говоря, в виде уравнения, содержащего несколько параметров. При изменении этой линии меняются только ее параметры, хранящиеся в ячейках памяти. Число же ячеек остается неизменным для любой линии. Линия рассматривается как элементарный объект векторной графики. Простейшие объекты объединяются в более сложные, например, объект "четырехугольник" можно рассматривать как четыре связанные линии, а объект "куб" еще более сложен — его можно рассматривать либо как двенадцать связанных линий, либо как шесть связанных четырехугольников. Поэтому векторную графику часто называют объектно-ориентированной графикой. Объекты векторной графики хранятся в памяти в виде набора параметров, но на экран и принтер все изображения выводятся в виде точек. Перед выводом на экран и принтер программа проводит вычисления координат экранных точек в изображении объекта, поэтому векторную графику иногда называют вычисляемой графикой.

**Фрактальная графика.** Как известно, фрактал — это рисунок, который состоит из подобных между собой элементов. Существует большое число графических изображений, которые являются фракталами: треугольник Серпинского, снежинка Коха, "дракон" Хартера—Хейтуея, множество Мандельброта. Построение фрактального рисунка осуществляется по определенному алгоритму путем автоматической генерации изображений с помощью вычислений по конкретным формулам. Изменения значений в алгоритмах или коэффициентов в формулах приводит к модификации этих изображений. Главным преимуществом фрактальной графики является то, что в файле фрактального изображения сохраняются только алгоритмы и формулы.

**Трехмерная графика.** Трехмерная графика (3D-графика), как известно, изучает приемы и методы создания объемных моделей объектов, которые максимально соответствуют реальным. Для создания объемных изображений используют разные графические фигуры и гладкие поверхности. Для двигающихся объектов указывают траекторию движения, скорость. Как показывает анализ публикаций, существует множество профессиональных программных средств для конвертации растрового формата в формат векторный. Ниже перечисляются наиболее распространенные средства:

- система GRASS предлагает два модуля для автоматической конвертации растровых линейных данных в векторный формат;
- система Vextractor v.2.70 Algolab Raster to Vector Conversion Toolkit 2.96 ScanPro fo Windows 5.0 Raster Desk 7 (SpotlightPro 7) обеспечивает конвертирование растра в вектор;
- система Vextractor 3.94 — это профессиональная программа для конвертации растровых изображений в векторные (vectorizer). Vextractor конвертирует рисунки, карты и другие изображения, включая логотипы и черно-белые иллюстрации из растрового формата в векторный;
- векторизатор Algolab Raster позволяет конвертировать отсканированные архитектурные и технические рисунки, схемы, карты, иллюстрации из растрового в векторный формат для дальнейшего использования в CAD или графическом редакторе.

Существует также множество профессиональных программных средств для конвертации векторного формата в формат растровый. Редактор SVG поддерживает как собственно векторный формат, так и различные растровые форматы. Можно "вставлять" в графический проект растровые изображения или конвертировать векторные изображения в растровые. Отсюда следует утверждение, что возможно сокрытие спецификаций и описаний схем в изображениях чертежей и схем. Это позволяет обеспечивать хранение защищенных проектных данных в базах данных информационных подсистем САПР и обеспечение безопасной передачи проектной документации по внутренним и внешним каналам данных. Для эффективной передачи больших объемов текстовых данных по внешним открытым линиям связи наиболее приемлемы аудиоконтейнеры.

## Заключение

Проектирование средств защиты проектной документации от несанкционированного доступа можно осуществлять двумя способами:

- разработкой метода шифрования проектных данных только на системотехническом или схемотехническом и физическом уровнях путем раздельного шифрования схем и чертежей и текстовой документации в виде описаний и спецификаций;
- использованием фрактальных форматов для защиты документации на архитектурном и функциональном уровнях.

**Список литературы**

1. **KPMG.** Global Information Security Survey 2002. URL: http://www.cnews.ru/consulting/kpmg.shtml (дата обращения 02.03.13).
2. **Точилов Л. С., Крылов Е. С.** Стратегия развития инфраструктуры сервисов IT-подразделений высокотехнологичного предприятия // Тр. 2-й Междунар. конф. CAD/CAM/PDM:. (Москва, 2002). М.: Институт проблем управления РАН. С. 270—274.
3. **Точилов Л. С.** Информационная безопасность и корпоративные сети. М.: Корпорация "Галактика", 1999. 36 с.
4. **Точилов Л. С.** Управление безопасностью на примере Федеральной резервной системы США. 2003. URL: http://lstochilov.narod.ru (дата обращения 02.03.13).
5. **OGC.** IT Infrastructure Library. Best Practice for Security Management. London: TSO. 2002. 124 с.
6. **Хузина Э. И.** Экспериментальные исследования алгоритма стеганографического сокрытия данных методом катера // Сб. тр. Третьей всероссийской научно-техн. конф. "Безопасные информационные технологии". (Москва, 2012). М.: МГТУ им. Н. Э. Баумана. С. 169-172.
7. **Чичварин Н. В.** Сопоставительный анализ областей применения и граничных возможностей характерных стеганографических алгоритмов // Сб. тр. Третьей всероссийской научно-техн. конф. "Безопасные информационные технологии". (Москва, 2012). М.: МГТУ им. Н. Э. Баумана. С. 174—179.
8. **Ларионцева Е. Л., Стельмашук Н. Н.** Экспериментальные исследования эффективности стеганографического алгоритма, реализующего метод lsb // Сб. тр. Третьей всероссийской научно-техн. конф. "Безопасные информационные технологии". (Москва, 2012). М.: МГТУ им. Н. Э. Баумана. С. 94—96.
9. **Логинов К. Е.** Экспериментальные исследования устойчивости алгоритма стеганографического сокрытия данных методом langelaar при воздействиях на стегоконтейнер // Сб. тр. Третьей всероссийской научно-техн. конф. "Безопасные информационные технологии". (Москва, 2012). М.: МГТУ им. Н. Э. Баумана. С. 99—101.
10. **Сиволапов А. С.** Исследование влияния контейнера на качество сокрытия сообщений методом Langelaar // Сб. тр. Третьей всероссийской научно-техн. конф. "Безопасные информационные технологии". (Москва, 2012). М.: МГТУ им. Н. Э. Баумана. С. 153—155.
11. **Круглая Е. И., Пилипенко А. В.** Защита данных в САПР: анализ стеганографических алгоритмов коча (koch) и бенхама (benham) // Сб. тр. Третьей всероссийской научно-техн. конф. "Безопасные информационные технологии". (Москва, 2012). М.: МГТУ им. Н. Э. Баумана. С. 87—90.
12. **Максимов Р. Л.** Экспериментальное исследование эффективности стеганографического алгоритма, реализующего метод брайндонкса (bruyndonckx) // Сб. тр. Третьей всероссийской научно-техн. конф. "Безопасные информационные технологии". (Москва, 2012). М.: МГТУ им. Н. Э. Баумана. С. 101—105.
13. **Гончаров И. О., Заикин М. А.** Экспериментальные исследования стеганографического метода эхо-кодирования // Сб. тр. Третьей всероссийской научно-техн. конф. "Безопасные информационные технологии". (Москва, 2012). М.: МГТУ им. Н. Э. Баумана. С. 45—48.
14. **Иванова Е. Ю.** Обзор атак на стегоалгоритм patchwork и методов противодействия // Сб. тр. Третьей всероссийской научно-техн. конф. "Безопасные информационные технологии". (Москва, 2012). М.: МГТУ им. Н. Э. Баумана. С. 66—69.
15. **Волосатова Т. М., Денисов А. В., Чичварин Н. В.** Комбинированные методы защиты данных в САПР // Информационные технологии. Приложение. 2012. № 5. 32 с.
16. **Волосатова Т. М., Денисов А. В., Чичварин Н. В.** Защита проектной документации от несанкционированного доступа // Тр. 9-й Междунар. конф. "Эффективные методы автоматизации подготовки и планирования производства". (Москва, 2012). М.: МГТУ им. Н. Э. Баумана. С. 141—144.
17. **Real-time** Watermarking Techniques for Compressed Video Data // Langelaar, Gerrit Cornelis — Thesis Delft University of Technology.(Veenendaal, 2000). V.: Universal Press. 136 с.
18. **Bruyndonckx O., Quisquater J.-J., Macq B.** Spatial method of copyright labeling of digital images // IEEE Workshop on Nonlinear Images/Signal Processing, Thessal. 1995. June. P. 19—27.
19. **Bender B., Morimoto N., Lu.** Methods of hiding data // IBM System Journal. 1996. July. P. 25—33.

УДК 004.457

**А. Ю. Долгопятов,** ст. преподаватель, e-mail: bosik_99@mail.ru
Азовский технологический институт Донского государственного технического университета, Азов

# Восстановление удаленных данных

*Рассмотрена возможность восстановления удаленных данных. Условно отказы жесткого диска можно разбить на несколько частей: отказ механической части, отказ электроники, дефекты поверхности, сбой файловой системы. В отличие от многих программ только некоторые могут восстанавливать сбойные секторы самого жесткого диска по всему заданному разделу или в заданном диапазоне. Некоторые протоколы, применяя технологию перемагничивания поверхности, обнуляют сбойные секторы винчестера.*

***Ключевые слова:*** *винчестер, случайное искажение, загрузочная запись, повреждение файловой системы, сервометки*

**A. Yu. Dolgopyatov**

# Recovery of Remote Data

*In article possibility of recovery of remote data is considered. Conditionally refusals of a hard disk can be broken into some parts: refusal of mechanical part, failure of electronics, defects of a surface, failure of file system. Unlike many programs, only some can restore faulty sectors of the most hard disk according to all set section or in the set range. Some protocols, knowing technology of magnetic reversal of a surface, only nullify faulty sectors.*

***Keywords:*** *winchester, casual distortion, loading record, damage of file system, servometka*

## Введение

Известно, что одним из наиболее распространенных способов несанкционированного доступа к информации является так называемый просмотр мусорных корзин. Для случая компьютерной информации данный метод заключается в попытках восстановления удаленных с помощью штатных средств файлов, просмотра сохранившихся после некорректного завершения программ временных файлов, кэш-областей браузеров и т. п. В наиболее распространенных операционных системах (ОС) семейства Microsoft и Nowell Netware при удалении файла стирается только его заголовок в FAT, а сам файл физически остается на носителе, и его можно элементарно "вытащить из корзины". Альтернативный подход к данной проблеме демонстрируют UNIX-подобные ОС, у которых удаление файла происходит на физическом уровне.

Различные сбойные ситуации, случающиеся с магнитными носителями, а также выход их из строя не являются серьезными препятствиями для восстановления данных. При уничтожении загрузочного сектора конкретные значения блока параметров BIOS (размер кластера, число кластеров в томе, число элементов FAT и т. п.) могут быть получены расчетным путем. Уничтожение таблиц FAT (например, при форматировании диска) значительно усложняет задачу восстановления данных, так как именно они определяют схему расположения файлов. Автоматическое восстановление данных с помощью утилит при этом событии не гарантирует полного восстановления, а вероятность падает с ростом степени фрагментации файлов.

В ряде случаев злоумышленники применяют принцип имитации выхода из строя накопителей на жестких магнитных дисках (НЖМД) после определенного периода функционирования и накопления информации. Поскольку в большинстве случаев невозможно провести ремонт и обслуживание вышедшего из строя НЖМД на месте, его заменяют на новый. При этом вся информация остается на подлежащем замене НЖМД.

Совершенствуются также методы восстановления информации. Так, разработаны методы, основанные на принципах магнитной силовой микроскопии (MFM). MFM базируется на сканирующей зондовой микроскопии, при которой магнитный наконечник зонда движется над поверхностью пластины на расстоянии порядка 1...10 нм. В зависимости от силы магнитного взаимодействия между пластиной и наконечником расстояние между ними изменяется. Эти колебания расстояния детектируются оптическим интерферометром. Полученное изображение представляет собой образ распределения намагниченности. Этими методами можно измерить магнитный рельеф поверхности диска и, следовательно, восстановить информацию. Вследствие очень высокой плотности записи в современных НЖМД привод магнитной головки не в состоянии точно следовать по требуемой траектории. Поэтому при записи новых данных поверх информации они всегда будут записаны с некоторым смещением относительно записанных ранее. Разрешающая способность магнитной силовой микроскопии достаточна для раздельного считывания нескольких последовательных записей информации.

В настоящее время для хранения компьютерной информации наиболее широко используются:

- магнитные носители (на основе ферромагнетиков);
- оптические носители;
- магнито-оптические носители.

Рассмотрим более подробно самые распространенные — магнитные носители.

Все магнитные носители выполняются на основе материалов, представляющих по своим свойствам ферромагнетики. Отличительной особенностью ферромагнетиков является наличие макроскопических объемов вещества — доменов, в которых магнитные моменты атомов (ионов) ориентированы одинаково. Домены обладают самопроизвольной намагниченностью (магнитными моментами) даже при отсутствии внешнего магнитного поля.

В ферромагнетике, не подвергавшемся воздействию внешних магнитных полей, магнитные моменты различных доменов обычно взаимно скомпенсированы, и их результирующее магнитное поле близко к нулю. Для ферромагнетиков характерен гистерезис при перемагничивании внешним магнитным полем, т. е. запаздывание изменений магнитной индукции вещества ($B$) от изменений магнитного поля ($H$). Под воздействием внешнего магнитного поля происходит ориентация элементарных магнитных полей, создаваемых круговым движением электронов в атомах и молекулах ферромагнетика. В результате увеличиваются размеры магнитных доменов, ориентированных по направлению внешнего поля. После прекращения внешнего воздействия изменения, происшедшие в размерах и ориентации магнитных доменов, частично сохраняются.

## Причины потери данных

В электрике могут возникнуть всего две неисправности: потеря контакта там, где он должен быть, или появление контакта там, где его быть не должно. На диске истинных причин потери данных тоже всего две: случайное изменение содержимого ячейки и повреждение этой ячейки. Случаев изменения содержимого очень много. Они будут рассмотрены в порядке "от поверхности диска до операционной системы".

***Неисправности платы электроники*** ведут к недоступности диска. При этом все данные могут быть в сохранности либо поврежденной может быть лишь часть информации. После замены или ремонта платы диск может использоваться снова.

***Поломки механической части*** внутри гермоблока (отрывы и сколы головок, разрушение подшипников и двигателей) очень часто сопровождаются и повреждением пластин. В таком случае винчестер чаще всего не запускается и не определяется BIOS компьютера. Бывает, что блок головок "застревает" в парковочной зоне. Возможна ситуация, когда жесткий диск стартует, но из-за повреждения одной головки недоступна группа секторов. Восстановление начинается с ремонта диска в "чистой комнате", а результат зависит от степени повреждения поверхности пластин.

***Повреждение пластин*** в общем виде проявляется как возникновение на диске BAD-блоков, т. е. недоступных участков. Появление этой проблемы говорит о том, что возможности скрытого переназначения секторов уже исчерпаны. Если на поврежденный сектор приходится информация о структуре, то исчезает соответствующий уровень логической структуры диска, если файл — файл становится нечитаемым.

***Случайное искажение*** содержимого сектора — довольно редкая ситуация. В цифровом мире случайностей почти не бывает. Это или проявление начинающегося "железного" дефекта, или результат работы вредоносной программы и в очень редких ситуациях — действительно случайная запись в момент перепадов напряжения или из-за пролетевшей космической частицы. Последнее — не шутка: воздействие космического излучения на оперативную память и жесткие диски давно изучено и статистически обосновано. Проявление дефектов или искажений содержимого секторов зависит от того, что в этих секторах записано.

***Повреждение содержимого главной загрузочной записи (MBR)*** ведет к тому, что разделы либо не могут быть найдены операционной системой, либо их параметры определяются неверно. Самый легкий случай — повреждение сигнатуры. Операционная система решает, что на месте MBR находится случайная информация, а сам диск вообще не разбит на разделы и никакой полезной информации не несет. Для восстановления структуры диска достаточно всего лишь исправить сигнатуру любым дисковым или HEX-редактором.

При искажении или разрушении кода загрузчика попытка загрузки операционной системы с такого диска заканчивается, как правило, "зависанием" компьютера. При этом если содержимое таблицы разделов не повреждено, вся логическая структура диска сохраняется и потери данных не происходит. Достаточно загрузить компьютер с другого диска, и вся информация на накопителе становится доступной. Для исправления кода загрузчика проще всего загрузить компьютер со стандартной загрузочной дискеты MS-DOS, а затем запустить находящуюся на ней утилиту fdisk с ключом /mbr: fdisk/mbr. Более тяжелый случай повреждения содержимого MBR — разрушение самого содержимого таблицы разделов. При этом теряется доступ к хранящейся в разделах информации. Таблица может быть повреждена полностью или частично. Бывают случаи, когда таблица разделов MBR цела, а разрушена запись о разделах в одном из звеньев цепи Extended Partition. Методика восстановления во всех случаях одинакова — ручное исправление таблиц разделов. Альтернатива — извлечение файлов с диска с помощью программ восстановления.

***Повреждение файловой системы*** очень похоже на повреждение содержимого MBR. Различие состоит лишь в том, что раздел на диске виден, но операционная система сообщает, что он не отформатирован. Соответственно, недоступно и все содержимое раздела. В файловых системах NTFS и FAT возможно восстановление основных записей за счет дубликатов, это делается благодаря встроенным в операционную систему средствам проверки и исправления дисков. Тем не менее, исправление ошибок файловой системы часто не решает проблему — часть файлов начинает рассматриваться как потерянные цепочки. Для восстановления данных правильнее пользоваться специальными программами восстановления.

***Корректное удаление файлов и папок*** средствами операционной системы — самая простая ситуация. Именно с ней чаще всего сталкиваются пользователи, и она же очень их пугает. В операционной системе Windows прежде всего надо искать файл в Корзине. При удалении файла или папки в таблице файловой системы сначала лишь меняется одно из полей: в FAT первый байт имени файла изменяется на 0xE5, а в MFT атрибут по смещению 14h изменяется с 1 на 0. Кроме того, в NTFS изменяется запись о свободном месте на диске в файле BitMaP. Существует множество программ, специально предназначенных для восстановления удаленных объектов.

### Задача восстановления данных

Эта задача является частным случаем исправления логических ошибок диска. Вся особенность состоит в том, что пользователь обычно знает, что, откуда и когда удалено, а диск, как правило, совершенно исправен и физически, и логически. Кроме того, прогноз восстановления полностью определяется тем, как быстро владелец компьютера вспомнил об утрате.

При физических дефектах или серьезных логических ошибках диска работа сразу прекращается, и винчестер экстренно начинают восстанавливать. После случайного удаления документов пользователь обычно работает до тех пор, пока эти документы вдруг не понадобятся. Если на место удаленных данных что-то было записано, вероятность восстановления уменьшается, а если записано было много, она стремится к нулю.

Извлекать данные после удаления файлов, форматирования или удаления разделов умеют практически все программы восстановления данных. Существуют и утилиты, специально предназначенные для восстановления случайно удаленной информации.

В действительности данные на пластинах винчестера организованы довольно сложно. Об истинном расположении данных на пластинах винчестера "знают" только его контроллер и микропрограмма. Пока все работает, через интерфейс винчестер видится как стандартная матрица блоков или секторов. Если же выходят из строя головки, разрушаются некоторые области пластин, поэтому прочитать данные можно лишь штатными средствами такого жесткого диска. Сами специалисты фирм-изготовителей признают, что все рассуждения на тему сканирования извлеченных из корпуса пластин, считывания остаточной намагниченности оказываются бесполезными. Даже теория хранения данных на винчестере оставляет место для неопределенности.

Достаточно жестко прописана на поверхности пластин лишь сервоинформация. Это магнитные метки и коды, которые указывают положение дорожек и секторов. Благодаря им головки позиционируются относительно пластин и находят нужные дорожки и сектора. Сервометки записываются на почти готовый винчестер в процессе изготовления на специальном оборудовании, после этого их невозможно ни стереть, ни изменить.

Полный объем каждого сектора составляет 571 байт. Из них 512 байт предназначены для записи данных (data), а 95 байт содержат служебные сведения о внутреннем номере сектора, контрольные суммы и т. д. Эта информация записывается при низкоуровневом форматировании диска еще на заводе, и доступ к ней через интерфейс предельно ограничен.

Если потеряны данные на жестком диске, начать стоит с простейшей диагностики проблемы. При этом важно решить: связана ли эта потеря с аппаратной аварией либо все ограничивается случайным искажением записей таблицы разделов или файловой системы.

В первом случае необходим срочный ремонт винчестера, после чего данные будут доступны. Во втором случае восстановление информации проводится программными методами.

Программное восстановление должно быть неразрушающим, т. е. не связанным с записью данных на проблемный диск. Сначала нужно создать полную посекторную копию диска на другом носителе, а затем всю дальнейшую работу проводить уже с этим образом.

## Заключение

Для извлечения информации более целесообразно использовать специальные программные пакеты, хотя возможно и ручное восстановление с помощью дисковых редакторов. Возможность извлечения данных отчасти зависит от числа попыток: разные программы по-разному справляются с восстановлением в конкретных ситуациях. И если требуемый результат не получен с первой попытки, стоит использовать другое средство.

### Список литературы


1. **Ташков П. А.** Восстанавливаем данные. СПб.: Питер, 2010.
2. **Смирнов Ю. К.** Секреты восстановления жестких дисков ПК. СПб.: БХВ — Петербург, 2011.
3. **Бобков С. Г.** Методика проектирования микросхем для компьютерной серии "Багет" // Информационные технологии. 2008. № 3. С. 2—7.
4. **Аряшев С. И., Барских М. Е.** Методы повышения производительности суперскалярного RISC-процессора "ИППМ РАН". М.: ИПУ РАН. 2005.

УДК 621.3

**В. М. Артюшенко,** д-р техн. наук, проф., e-mail: artuschenko@mail.ru,
**Т. С. Аббасова,** канд. техн. наук, доц., e-mail: abbasova_univer@mail.ru,
ФГБОУ ВПО "Финансово-технологическая академия", г. Королев

# Эффективность защиты от внешних помех электропроводных каналов структурированных кабельных систем для передачи высокоскоростных информационных приложений

*Осуществлен анализ защиты электротехнических кабелей "витая пара" для передачи данных в диапазоне частот 100...500 МГц от внешних электромагнитных воздействий. Получены графические зависимости, позволяющие обеспечивать электромагнитную совместимость при построении и эксплуатации структурированных кабельных систем для передачи данных высокоскоростных протоколов.*

***Ключевые слова:*** *внешнее электромагнитное воздействие, электромагнитная совместимость, структурированные кабельные сети*

**V. M. Artuschenko, T. S. Abbasova**

# Effective Protection from External Interference Conductivity Channel Structured Cabling Transmission High-Speed Data Applications

*The analysis of protection of the electrotechnical cables "twisted couple" for data transmission in the range of frequencies of 100...500 MHz, from external electromagnetic influences is carried out; the graphic dependences, allowing to provide electromagnetic compatibility at construction and operation of the structured cable systems for data transmission of high-speed protocols are received.*

***Keywords:*** *external electromagnetic effects; electromagnetic compatibility; structured cabling systems*

## Введение

Фундаментальные исследования в области теории передачи информации для описания формирования процессов и функционирования симметричных электропроводных кабельных линий структурированных кабельных систем (СКС) и мультисервисных кабельных систем (МКС), передающих по кабельным сетям видео-, аудио- и мультимедиа информацию, проведены следующими отечественными учеными: Андрушко Л. М., Семеновым Н. А., Ватутиным В. М., Шавриным С. С. Современное электрооборудование СКС и МКС достаточно надежно, однако эволюция технологий в сторону высоких частот делает актуальной проблему электромагнитной совместимости (ЭМС) для постоянно растущего числа электротехнических и электронных устройств СКС и МКС. Увеличение рабочей частоты кабельных систем до 500 МГц приводит как к увеличению уровня собственных излучений кабельных каналов, так и их большей уязвимости к внешним электромагнитным воздействиям [1—4].

## Постановка задачи

В условиях интенсивного внедрения новых высокочастотных технологий передачи и обработки данных необходимо ужесточение требований к обеспечению электромагнитной совместимости электрооборудования кабельных систем. Электрооборудование кабельных систем должно обеспечивать не только неискаженный прием сигналов в условиях воздействия внешних электромагнитных помех, но и оптимальные параметры передачи в диапазоне частот, необходимые для передачи управляющей информации в режиме реального времени, а также требуемую защищенность между цепями внутри кабеля. Одной из важнейших проблем при построении и эксплуатации СКС является проблема обеспечения электромагнитной совместимости (ЭМС) [5—13]. Особую актуальность эта проблема приобретает при передаче по СКС сигналов высокоскоростных протоколов и их прокладки в непосредственной близости от телекоммуникационных и силовых кабелей [14—15].

### Анализ эффективности защиты горизонтальной подсистемы СКС на основе кабеля "витая пара" от внешних помех

СКС делится на три подсистемы (рис. 1):

- магистральная подсистема комплекса;
- магистральная подсистема здания;
- горизонтальная подсистема.

Распределительные пункты (РП), телекоммуникационные разъемы (ТР), точки перехода (ТП) обеспечивают возможность создания топологии каналов типа "шина", "звезда" или "кольцо".

В качестве среды передачи сигналов для горизонтальной кабельной подсистемы обычно используется электрический кабель "витая пара", для магистральных подсистем — оптоволоконный кабель. Для передачи данных высокоскоростных протоколов по кабелям горизонтальной подсистемы необходима скорость 10 Гбит/с (приложение 10GBaseT). Группой IEEE 802.3 подготовлены спецификации интерфейсов для передачи данных по кабелям из витых пар со скоростью 10 Гбит/с, для достижения которой по каждой из четырех пар горизонтального кабельного тракта (100 м, четыре точки коммутации) в двух направлениях должны передаваться сигналы со скоростью 2,5 Гбит/с. Скорость 2,5 Гбит/с позволяет организовывать стык между локальной сетью и глобальными сетями ATM (от англ. Asynchronous Transfer Mode — асинхронный режим переноса информации), одна из иерархических ступеней которых имеет скорость передачи около 2,5 Гбит/с. Типичная дальность передачи для горизонтальных каналов СКС составляет 50...100 м, поэтому нет необходимости использовать оптические варианты сетевого интерфейса длиной от сотен метров до нескольких километров для достижения заданной пропускной способности.

Распространены следующие конструкции кабеля "витая пара" для приложений 10GBaseT:

- неэкранированный кабель UTP (от. англ. Unshielded Twisted Pair — неэкранированная витая пара) категории 6а (диаметр ~7,9 мм);
- экранированный кабель FTP (от. англ. Foiled Twisted Pair — экранированная витая пара) категории 6 (диаметр ~ 6,3 мм).

Наряду с перечисленными существуют новые предложения: экранированный кабель категории 7 (диаметр ~ 7,8 мм); дважды экранированные кабели категории 8 с медными проводниками (диаметр ~ 6,0 мм).

В СКС "эконом-класс" в качестве среды передачи для горизонтальной подсистемы используют неэкранированные кабели с сопротивлением 100 Ом (ANSI/TIA/EIA-568-B.2) категорий 6, 6а (ANSI/TIA/EIA-568-B.2-1). Для горизонтальных электрических кабелей длина не должна превышать 90 м (длина горизонтального тракта с коммутационными и абонентскими кабелями не должна превышать 100 м). Кроме кабелей категорий 6, 6а в горизонтальной подсистеме возможна прокладка оптоволоконных кабелей, однако для систем "эконом-класс" достаточно электрических кабелей, но при этом должна быть обеспечена их электромагнитная совместимость.

Проведем анализ эффективности защиты горизонтальной подсистемы СКС на основе кабеля "витая пара" от внешних помех с точки зрения обеспечения электромагнитной совместимости.

Важным аспектом проблемы ЭМС является взаимное влияние кабелей, проложенных в одном и том же кабелепроводе. Согласно теории, если источником помех, воздействующих на неэкранированный кабель СКС, является проложенный вместе с ним кабель, излучающий электромагнитные помехи, то для приближенной оценки электромагнитных помех можно воспользоваться выражением [5, 7]

$$A_{\text{п}} \approx A_{\text{пп}} - 20\lg(50L_{\text{п}}/rR_{\text{к}}),\ \text{дБ}, \qquad (1)$$

где $L_{\text{п}}$ — длина кабеля, подверженного воздействию электромагнитной помехи, м; $r$ — расстояние между "мешающим" кабелем и кабелем, подверженным воздействию помехи, м; $R_{\text{к}}$ — сопротивление кабеля, Ом; $A_{\text{пп}} = f(F_{\text{п}})$ — ослабление электромагнитных помех, для двух параллельно проложенных кабелей в зависимости от рабочей частоты "мешающего" кабеля $F_{\text{п}}$ (рис. 2).

Если один из кабелей экранирован, то для определения ослабления $A_{\text{пп}}$ можно воспользоваться гра-


**Рис. 1. Подсистемы СКС**

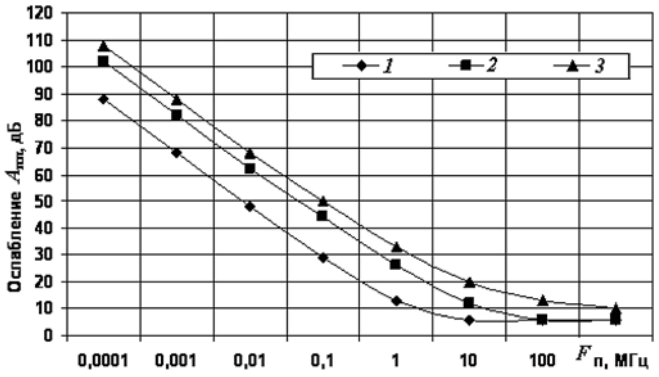


**Рис. 2. Ослабление электромагнитных помех, при параллельной прокладке двух неэкранированных кабелей, один из которых является источником помех:**
*1* — $L_{\text{п}}$ = 10 м, $r$ = 0,01 м; *2* — $L_{\text{п}}$ = 2 м, $r$ = 0,01 м; *3* — $L_{\text{п}}$ = 1 м, $r$ = 0,01 м

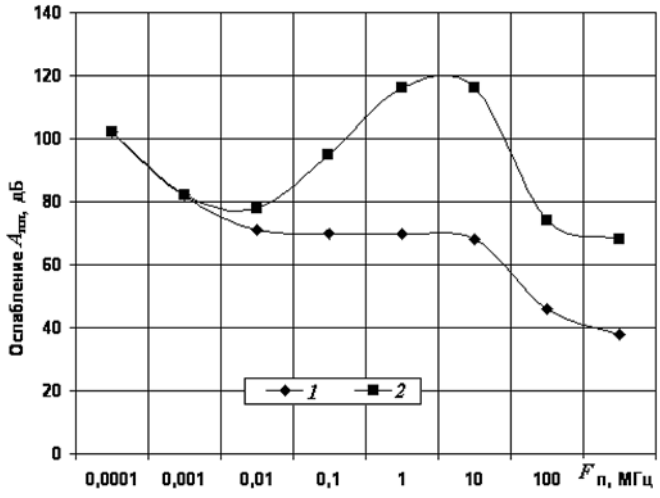


**Рис. 3. Ослабление электромагнитных помех, при параллельной прокладке двух кабелей (проводов), один (или оба) из которых защищены экраном:**
*1* — один кабель экранирован; *2* — оба кабеля экранированы

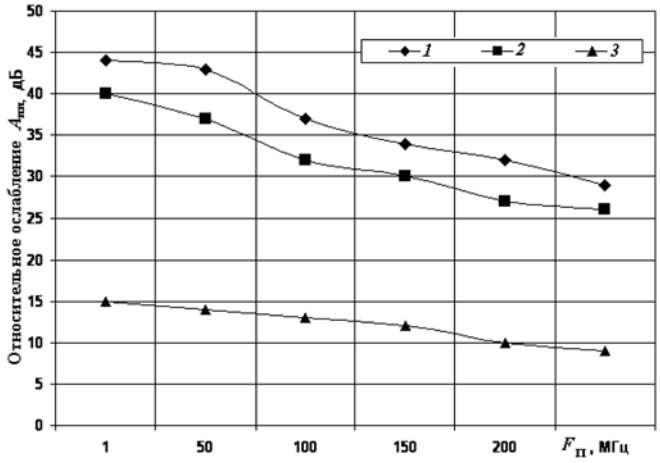


**Рис. 4. Зависимость относительного ослабления от частоты**

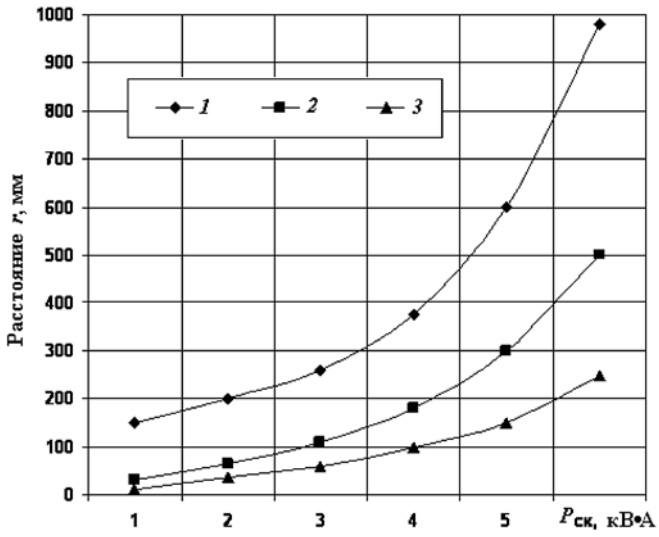


**Рис. 5. Предельно допустимые расстояния между силовыми и информационными кабелями без металлического разделителя:**
*1* — оба кабеля неэкранированы; *2* — один из кабелей экранирован; *3* — оба кабеля экранированы

фиками, представленными на рис. 3, когда $L_п = 2$ м, $r = 0{,}01$ м.

Анализируя выражение (1) и приведенные на рис. 2 и 3 зависимости, можно сделать следующие выводы. Для уменьшения влияния электромагнитных помех в случае прокладки двух кабелей, один из которых является источником помех, необходимо:

- максимально разнести их друг от друга;
- обеспечить их защиту простейшим защитным экраном;
- максимально уменьшить длину кабеля, подверженного воздействию электромагнитной помехи.

На рис. 4 представлены зависимости ослабления электромагнитной помехи для трех случаев защиты информационного кабеля: *1* — экранирование кабеля; *2* — использование алюминиевой вставки; *3* — использование металлического напыления короба из ПВХ [5].

Наилучшую защиту обеспечил экранированный кабель. Применение алюминиевой вставки показало себя на 5 дБ хуже во всем диапазоне частот. Металлическое напыление оказалось на 20...30 дБ или примерно в 100...1000 раз менее эффективным, чем использование экранированного кабеля [5, 8].

Воспользовавшись методикой, изложенной в стандартах EIA/TIA 569 и EN50174-2, были получены зависимости предельно допустимых расстояний между информационными и силовыми кабелями, проложенными с использованием различных разделителей (рис. 5—10).

Из представленных зависимостей видно, что использование разделителя между неэкранированным силовым и информационным кабелем позволяет значительно уменьшить предельно допустимое расстояние. Так, при передаче по силовому кабелю мощности $P_{ск} = 3$ кВ · А алюминиевый разделитель уменьшает его почти в 1,5 раза, стальной — в 2,6 раз. При 5 кВ · А, алюминиевый разделитель уменьшает в 1,2 раза, стальной — почти в 2 раза [8].

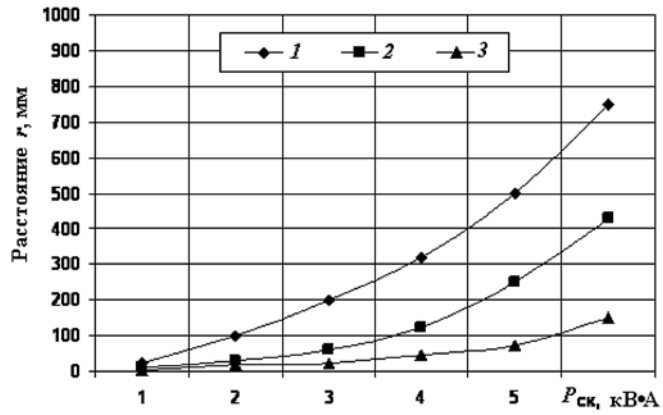


**Рис. 6. Предельно допустимые расстояния между силовыми и информационными кабелями с алюминиевым разделителеме:**
*1* — оба кабеля неэкранированы; *2* — один из кабелей экранирован; *3* — оба кабеля экранированы

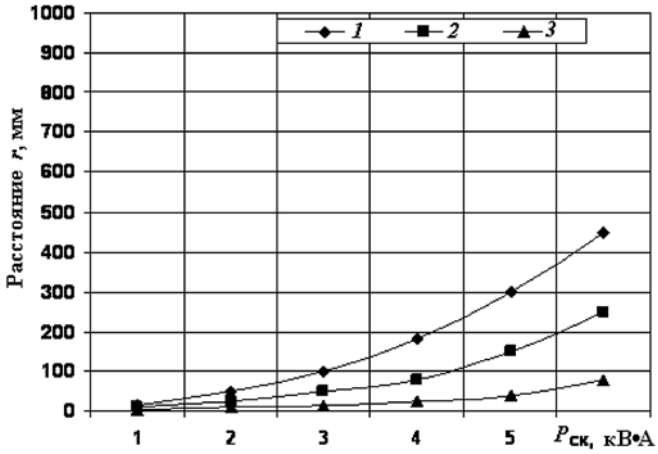


**Рис. 7. Предельно допустимые расстояния между силовыми и информационными кабелями со стальным разделителем:**
*1* — оба кабеля неэкранированы; *2* — один из кабелей экранирован; *3* — оба кабеля экранированы

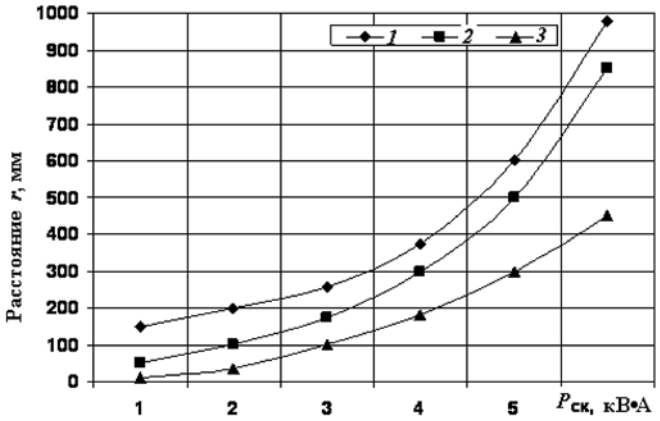


**Рис. 8. Предельно допустимые расстояния между неэкранированными силовыми и информационными кабелями:**
*1* — без металлического разделения; *2* — с алюминиевым разделителем; *3* — со стальным разделителем

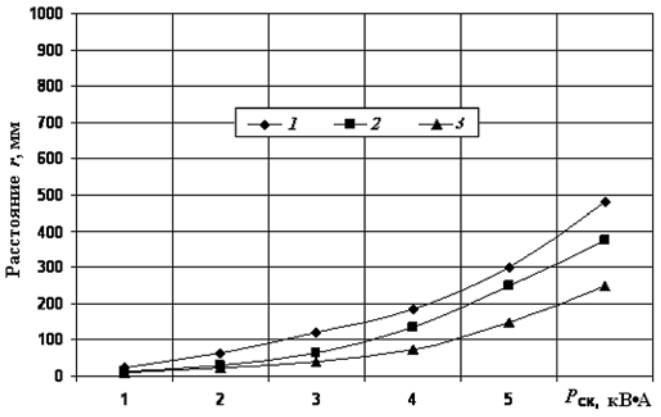


**Рис. 9. Предельно допустимые расстояния между экранированными силовыми и неэкранированными информационными кабелями:**
*1* — без металлического разделения; *2* — с алюминиевым разделителем; 3 — со стальным разделителем

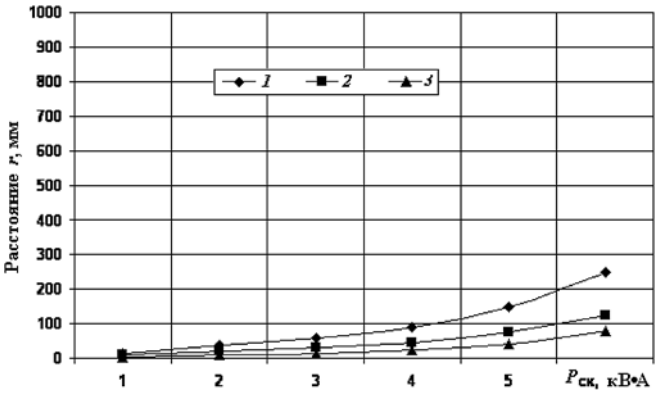


**Рис. 10. Предельно допустимые расстояния между экранированными силовыми и информационными кабелями:**
*1* — без металлического разделения; *2* — с алюминиевым разделителем; *3* — со стальным разделителем

## Результаты компьютерных исследований помехозащищенности электрических кабелей СКС

Для исследования устойчивости СКС к внешним электромагнитным воздействиям была разработана специальная компьютерная программа в MATLAB, позволяющая имитировать работу сети, использующую стандартный высокоскоростной протокол передачи данных ATM 155, в условиях воздействия радиоизлучений. Программа основывалась на методике измерений, разработанной лабораторией NAMAS, и рекомендациях стандарта EN61000-4-3: 1996.

Неэкранированная система категории 6 не обеспечила защиту от электромагнитных помех для напряженности поля в 3 В/м. Системы, использующие экранированные кабели, устойчиво работали до уровня наводок в 15 В/м [10, 12].

На рис. 11 представлены уровни наведенных напряженностей поля $U_1$ в витых парах неэкраниро-

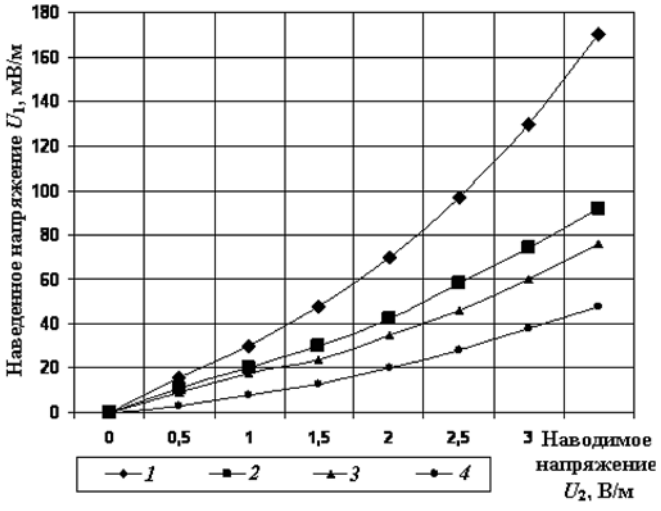


**Рис. 11. Уровни наведенных напряженностей поля в витых парах неэкранированной системы:**
*1* — 3-я пара; *2* — 1-я пара; *3* — 2-я пара; *4* — 4-я пара

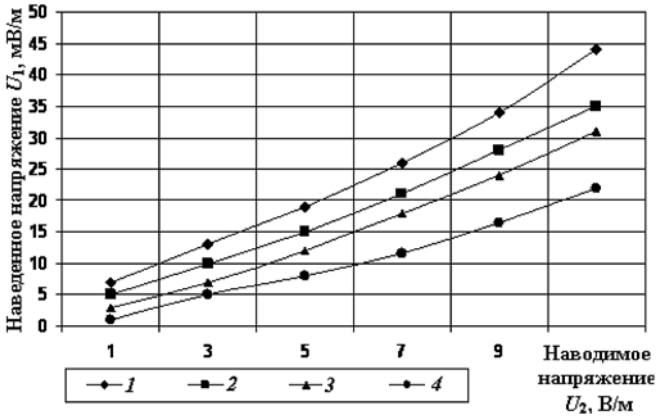


**Рис. 12. Уровни наведенного напряжения в витых парах экранированной системы:**
*1* — 3-я пара; *2* — 1-я пара; *3* — 2-я пара; *4* — 4-я пара

ванной системы в зависимости от наводимого напряжения $U_2$.

На рис. 12 приведены результаты тестирования, характеризующие относительную устойчивость экранированной системы к воздействию внешних наводок [12].

Полученные результаты хорошо совпадают с результатами компании ITT NS & S, организовавшей многочисленные измерения систем разных производителей, проведенные различными лабораториями в США и Англии.

## Заключение

Таким образом, сравнительный анализ защиты от взаимного влияния электрических кабелей СКС, проложенных в одном и том же кабелепроводе, показал, что в диапазоне до 250 МГц наилучшую защиту обеспечивает экранированный кабель. Алюминиевая вставка защищает информационный кабель на 5 дБ хуже во всем диапазоне частот. Металлическое напыление оказалось примерно в 100...1000 раз менее эффективным, чем применение экранированного кабеля.

Использование разделителей между неэкранированным силовым и информационным кабелем позволяет значительно уменьшить предельно допустимые расстояния. Так, при 3 кВА алюминиевый разделитель уменьшает его почти в 1,5 раза, стальной — в 2,6 раз.

Для обеспечения защиты от электромагнитных помех при напряженности поля выше 3 В/м необходимо использовать экранированные СКС категории 6, так как уровень наводок в неэкранированной системе оказался в 5...10 раз выше, чем в экранированной.

### Список литературы

1. **Семенов А. Б.** Эволюция и направления развития систем интерактивного управления СКС // Вестник связи. 2005. № 10. С. 37—43.

2. **Семенов А. Б.** Последние новинки СКС // Вестник связи. 2006. № 6. С. 42—48.

3. **Шаврин С. С.** Механизм и информационная модель субъективного восприятия эхосигналов // Электросвязь. 2008. № 7. С. 16—20.

4. **Шаврин С. С.** Оценка мешающего воздействия электрического эха на абонентов // Электросвязь. 2008. № 9. С. 53—55.

5. **Артюшенко В. М.** Защита структурированных кабельных систем от внешних электромагнитных воздействий // Промышленный сервис. 2005. № 3. С. 20—27.

6. **Артюшенко В. М., Гуреев А. К., Абраменков В. В., Енютин К. А.** Мультимедийные гибридные сети. М.: МГУС, 2007. 94 с.

7. **Артюшенко В. М., Аббасова Т. С.** Электромагнитная совместимость электропроводных кабелей и коммутационного оборудования высокоскоростных структурированных кабельных систем // Электротехнические и информационные комплексы и системы. 2008. Т. 4, № 4. С. 22—29.

8. **Артюшенко В. М., Маленкин А. В.** Количественная оценка электромагнитного влияния однопроводных линий электротехнического оборудования // Электротехнические и информационные комплексы и системы. 2008. Т. 4, № 1, 2. С. 29—32.

9. **Артюшенко В. М., Корчагин В. А.** Проблемы электромагнитной совместимости цифрового электротехнического оборудования на промышленных и бытовых объектах // Вестник ассоциации вузов туризма и сервиса. 2009. № 4 (11). С. 95—98.

10. **Артюшенко В. М., Аббасова Т. С.** Проектирование мультисервисных систем в условиях воздействия внешних электромагнитных помех / Под ред. В. М. Артюшенко. М.: ФГОУ ВПО РГУТиС, 2011. 110 с.

11. **Артюшенко В. М.** Оценка электромагнитных наводок в информационных экранированных кабельных линиях // Информационные технологии. Радиоэлектроника. Телекоммуникации (ITRT-2012). Сб. ст. II Междунар. заочной научно-технической конф. Ч. 1. Тольятти: Изд-во ПВГУС, 2012. С. 54—71.

12. **Артюшенко В. М., Аббасова Т. С.** Расчет и проектирование структурированных мультисервисных кабельных систем в условиях мешающих электромагнитных воздействий: учеб. пособие / Под ред. В. М. Артюшенко. Королев МО: ФТА, 2012. 264 с.

13. **Аббасова Т. С., Артюшенко В. М.** Электромагнитная совместимость электропроводных кабелей и коммутационного оборудования высокоскоростных структурированных кабельных систем // Электротехнические и информационные комплексы и системы. 2008. Т. 4, № 4. С. 22—29.

14. **Артюшенко В. М., Аббасова Т. С.** Особенности резервирования источников бесперебойного питания компьютерного и телекоммуникационного оборудования // Электротехнические и информационные комплексы и системы. 2007. Т. 3, № 3. 20 с.

15. **Артюшенко В. М., Аббасова Т. С.** Сервис информационных систем в электротехнических комплексах / Под науч. ред. В. М. Артюшенко. М.: ФГОУ ВПО РГУТиС, 2010. 98 с.

ЖУРНАЛ В ЖУРНАЛЕ

# НЕЙРОСЕТЕВЫЕ ТЕХНОЛОГИИ

№ 5
МАЙ
2014

УДК 004.855.5

**Э. Д. Аведьян,** д-р техн. наук, гл. науч. сотр., e-mail: avedian@mail.ru,
**В. Э. Луганский,** зам. директора, e-mail: lugansky@inevm.ru,
ФГАНУ "Центр информационных технологий и систем органов исполнительной власти"
(ФГАНУ ЦИТиС), г. Москва

# Способы повышения точности заполнения числовых пропусков в таблицах, основанные на модифицированных нейронных сетях CMAC

*Настоящая работа, в которой задача восстановления пропущенных данных в таблицах решается с помощью модифицированных нейронных сетей CMAC (НС CMAC), развивает результаты, полученные в работе [1]. В роли модифицированных НС CMAC выступают двухслойная НС CMAC и НС CMAC на основе метода наименьших квадратов, которые решают задачу восстановления пропущенных данных с более высокой точностью, чем это делает классическая НС CMAC. Показано, что из двух модифицированных сетей предпочтение следует отдать НС CMAC на основе метода наименьших квадратов, точность решения которой выше, чем у двухслойной нейронной сети.*

***Ключевые слова:*** *заполнение числовых пропусков в таблицах, двухслойная нейронная сеть CMAC, нейронная сеть CMAC на основе метода наименьших квадратов, компьютерное моделирование*

**E. D. Aved'yan, V. E. Lugansky**

# Methods for Improving the Accuracy of Filling Gaps in Tables Based on the Modified CMAC Neural Networks

*The present work, in which the problem of filling gaps in tables is solved using modified CMAC neural networks (CMAC NN), develop the results obtained in [1]. In the role of the modified CMAC NN act two layer CMAC NN and CMAC NN based on the least squares method. Modified NN solve the problem of reconstructing missing data with higher accuracy than it does the classical CMAC NN. It is shown that the preference should be given to the CMAC NN based on the least squares method for which the accuracy solution is higher than for two-layer CMAC NN.*

***Keywords:*** *filling gaps in table, CMAC NN based on the least squares method, two layer CMAC NN, computer simulation*

## Введение

В работе [1] отмечалось, что проблема обнаружения и восстановления пропущенных числовых данных сопутствует многим практическим задачам, а нейросетевые методы наиболее перспективны в решении данной проблемы. Приведенные в работе [1] результаты компьютерного моделирования показывают, что классическая нейронная сеть CMAC (НС CMAC) [2, 3] достаточно хорошо приспособлена для решения задачи восстановления пропущенных числовых данных в таблицах, причем точность восстановления пропущенных данных зависит от свойств восстанавливаемых данных. Настоящая работа, в которой задача восстановления пропущенных данных в таблицах решается с помощью модифицированных НС CMAC, развивает результаты, полученные в работе [1]. В роли модифицированных нейронных сетей CMAC выступают двухслойная НС CMAC [4] и НС CMAC на основе метода наименьших квадратов [5, 6], которые, как будет показано далее, решают задачу восстановления пропущенных данных с более высокой точностью, чем это делает классическая НС CMAC.

В разд. 1 дается краткое описание этой нейронной сети, как основа для понимания функционирования модифицированных НС CMAC. В разд. 2 и 3 описываются структуры двухслойной НС CMAC и НС CMAC на основе метода наименьших квадратов. Результаты сравнительного компьютерного моделирования рассматриваемой задачи приведены в разд. 4, где показано, что НС CMAC на основе метода наименьших квадратов является наиболее эффективным инструментом решения задачи восстановления пропущенных числовых данных в таблицах.

## 1. Структура и алгоритм функционирования нейронной сети CMAC в задаче восстановления пропущенных числовых данных в таблице

Впервые НС CMAC была представлена в работах [2, 3], подробное ее описание можно найти в русскоязычных статьях [4, 7]. Наиболее существенным отличием НС CMAC от других нейронных сетей является следующее:

- аргументы $x$ запоминаемой функции $y(x)$ и воспроизводимой функции $\tilde{y}(x)$ принимают только дискретные значения;

- нелинейное преобразование аргументов функции выполняется с помощью алгоритма вычисления номеров ячеек ассоциативной памяти, в которых хранятся числа, определяющие значение функции.

Областью определения запоминаемой функции $N$-переменных $y(x)$ является целочисленная $N$-мерная сетка

$$X = \{x^{(1)} = \overline{1, x_{\max}^{(1)}}; x^{(2)} = \overline{1, x_{\max}^{(2)}}, ...; x^{(N)} = \overline{1, x_{\max}^{(N)}}\}. \quad (1)$$

В НС СМАС каждый входной $N$-мерный сигнал $x$ (аргумент функции) возбуждает или делает активными ровно ρ* ячеек памяти, суммарное содержимое которых равно значению запоминаемой функции. Каждому входному $N$-мерному вектору $x$ однозначно соответствует ρ*-мерный вектор активных номеров ячеек памяти $m$. Параметр ρ* (обобщающий параметр) определяет разрешающую способность НС СМАС и требуемый объем памяти. Алгоритм вычисления номеров активных ячеек памяти можно найти в работе [7], его вывод приведен в работе [8].

Введем следующие переменные: $w[n]$ — $M$-мерный вектор памяти сети, вычисленный на $n$-м шаге обучения, каждая $j$-я компонента которого $w_j[n]$, $j = \overline{1, M}$, соответствует содержимому $j$-й ячейки памяти НС СМАС; $a(x)$ — $M$-мерный ассоциативный вектор, однозначно связанный с вектором аргументов $x$ посредством вектора $m$ номеров активных ячеек памяти по следующему правилу: элементы вектора $a_j(x)$, $j = \overline{1, M}$, номера которых совпадают с номерами активных ячеек памяти, равны единице, все остальные элементы вектора $a(x)$ равны нулю. Тогда в соответствии с правилом функционирования нейронной сети НС СМАС ее выход $\tilde{y}(x[n])$ при заданном входном векторе $x[n]$ равен сумме содержимого активных ячеек памяти, т. е. скалярному произведению векторов $a(x[n])$ и $w[n]$:

$$\tilde{y}(x[n]) = a^{\mathrm{T}}(x[n])w[n]. \quad (2)$$

Алгоритм обучения нейронной сети НС СМАС функционирует следующим образом.

Пусть после $(n-1)$-го измерения значения запоминаемой функции $y(x)$ и соответствующего значения вектора аргументов $x$, $(y(x[i]), x[i])$, $i = \overline{1, n-1}$, вычислен вектор памяти $w[n-1]$. Тогда на следующем $n$-м шаге после измерения значения функции $y[n] \equiv y(x[n])$ при известном значении аргумента $x[n]$ сначала с помощью алгоритма нелинейного преобразования аргументов вычисляются номера активных ячеек памяти, далее вычисляется предсказываемое нейронной сетью значение функции $\tilde{y}[n] \equiv \tilde{y}(x[n])$, равное сумме содержимого активных ячеек памяти. Вычисляются ошибка предсказания $\varepsilon[n] = y[n] - \tilde{y}[n]$ и значение коррекции $\Delta w[k] = \gamma\varepsilon[n]/\rho^*$, $0 < \gamma < 2$, которая прибавляется к содержимому активных ячеек памяти. Неактивные ячейки коррекции не подвергаются. Аналитическая форма алгоритма обучения имеет вид:

$$w[n] = w[n-1] + \gamma \frac{y[n] - a^{\mathrm{T}}(x[n])w[n-1]}{a^{\mathrm{T}}(x[n])a(x[n])} a(x[n]), \quad (3)$$

$$w[0] = w_0,\ n = 1, 2, \ldots .$$

Обученная нейронная сеть НС СМАС способна предсказывать значения функции, которые ей не были предъявлены при обучении. Для этого необходимо на вход обученной сети подать вектор аргументов запрашиваемого значения функции, нейронная сеть по данному вектору вычислит адреса активных ячеек памяти, сумма содержимого которых и будет значением предсказываемой функции.

Рассмотрим кратко, как в работе [1] НС СМАС решает задачу заполнения числовых пропусков в таблице. Пусть имеется таблица, содержащая $N_1$ строк и $N_2$ столбцов. Числовые данные таблицы будем рассматривать как значения функции двух дискретных переменных $n_r$ и $n_c$, где $n_r$ — номера строк, а $n_c$ — номера столбцов таблицы, причем $n_r = \overline{1, N_1}$, $n_c = \overline{1, N_2}$. Предположим, что выполнен анализ таблицы в целях автоматического выделения координат пропущенных и непропущенных данных, в результате чего сформированы два множества: 1) $\{X_{exst}, Y_{exst}\}$ — множество, состоящее из координат непропущенных данных $X_{exst}$ и соответствующих им значений таблицы $Y_{exst}$; 2) множество координат пропущенных данных $X_{gap}$. Процедура автоматического формирования этих множеств описана в работе [1].

Задается область определения (1) запоминаемой таблицы в виде квадрата:

$$X = \{x^{(1)} = \overline{1, x_{\max}^{(1)}}; x^{(2)} = \overline{1, x_{\max}^{(2)}}\}, \quad (4)$$

где $x_{\max}^{(1)} = x_{\max}^{(2)} = x_{\max}$, причем сторона квадрата вычисляется по формуле

$$x_{\max} = 2^L + 1,$$

где

$$L = \begin{cases} Trunc(\log_2(d-1)) + 1, \\ \text{если } (\log_2(d-1) - Trunc(\log_2(d-1)) > 0, \\ \log_2(d-1), \text{ в противном случае,} \end{cases} \quad (5)$$

а $d = \max(N_1, N_2)$ — максимальное из двух чисел $N_1$ и $N_2$. Функция $Trunc(a)$ — целая часть числа $a$. При таком задании параметров (4), (5) область определения НС СМАС полностью включает в себя восстанавливаемую таблицу. Параметр ρ* может принимать следующие значения: $\rho^* = 2^k$, где $k = \overline{1, L}$.

Для таблицы, которая будет предметом исследований в настоящей работе и которая исследовалась в работе [1], число строк $N_1 = 14$ и число столбцов $N_2 = 12$. Для этой таблицы согласно формулам (4), (5) сторона квадрата $x_{\max} = 17$, а обобщающий параметр ρ* может принять четыре значения: 2, 4, 8 и 16.

Из множества $\{X_{exst}, Y_{exst}\}$ формируется обучающая выборка длиной $n$ случайным извлечением элементов из этого множества, эта выборка используется для оценивания оптимального значения обобщающего параметра ρ* и последующего обучения НС CMAC при оптимальном значении ρ*.

Качество заполнения числовых пропусков оценивают по двум критериям. Первый критерий $E_{indirect}$ [1] применяют в случаях, когда необходимо восстановить пропущенные числовые данные, а информация о пропущенных данных отсутствует. Его используют для нахождения оптимальных параметров НС CMAC. Второй критерий $E_{direct}$ применяют в случае, когда информация о пропущенных числовых данных имеется, например, при исследовании того или иного алгоритма восстановления пропущенных данных.

Критерий $E_{indirect}$ строят следующим образом. Выполняется обучение нейронной сети при заданных значениях параметров сети с помощью части множества непропущенных данных $\{X_{exst}, Y_{exst}\}$, в котором поочередно удаляются точки из непропущенных данных. Вычисляют оценки удаленных значений с помощью обученной нейронной сети, что и позволяет вычислить косвенную ошибку заполнения пропусков в таблице (в процентах):

$$E_{indirect} = 100 \sum_{i=1}^{N_{exist}} \frac{|y_{exst}[i] - \tilde{y}_{exst}[i]|}{|y_{exst}[i]|} \frac{1}{N_{exst}}, \qquad (6)$$

где $y_{exst}[i]$ — $i$-я удаленная точка из непропущенных данных; $\tilde{y}_{exst}[i]$ — оценка $i$-й удаленной точки; $|a|$ — модуль $a$.

Процедура косвенного оценивания ошибки заполнения пропусков в таблице не зависит от того, какую нейронную сеть используют для восстановления пропусков в таблице.

Обученная при оптимальных значениях параметров НС CMAC позволяет оценить значения пропущенных данных таблицы. С этой целью на нее последовательно подают координаты пропущенных данных из множества $X_{gap}$, а выход НС CMAC определяет оценку пропущенного значения таблицы для соответствующей координаты.

При построении критерия $E_{direct}$ предполагается, что пропущенные числовые данные таблицы априори известны. В этом случае после обучения сети при выбранных оптимальных значениях параметров, полученных с помощью критерия $E_{indirect}$, можно вычислить критерий $E_{direct}$ как истинную ошибку между известными значениями пропущенных данных $y_{gap}[i]$ и их оценками $\tilde{y}_{gap}[i]$, $i = 1, N_{gap}$:

$$E_{direct} = 100 \sum_{i=1}^{N_{exist}} \frac{|y_{gap}[i] - \tilde{y}_{gap}[i]|}{|y_{gap}[i]|} \frac{1}{N_{gap}}, \%, \qquad (7)$$

где $N_{gap}$ — число пропущенных данных таблицы.

Характер поведения истинной ошибки обучения $E_{direct}$ на примере решения задачи заполнения пропусков в табл. 1 с помощью классической нейронной сети CMAC отражает табл. 2. Обе таблицы заимствованы из работы [1]. Из табл. 2 следует, что ошибка обучения $E_{direct}$ довольно быстро входит в некоторую область и колеблется в ней в среднем около значения 8 %. Увеличение длины обучающей последовательности не влияет на случайное колебательное поведение этой ошибки.

Поясним причину такого поведения ошибки обучения $E_{direct}$. Алгоритм обучения НС CMAC (3) является рекуррентной процедурой решения следующей системы линейных уравнений:

$$a^{\mathrm{T}}(x_{exist}[i])w = y_{exist}[i], \quad i = \overline{1, N_{exist}}, \qquad (8)$$

где $N_{exist}$ — число непропущенных числовых данных таблицы; $w$ — вектор памяти нейронной сети размерности $M$, которая зависит

Таблица 1

**Таблица 14 × 12, содержащая 30 пропусков (около 18 % всех элементов таблицы)**

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 3,320 | 3,749 | | | 4,811 | 5,116 | 5,404 | 5,677 | 5,938 | 6,188 | 6,428 | |
| 4,124 | 4,708 | 5,265 | | 6,310 | 6,803 | 7,277 | 7,732 | 8,168 | 8,587 | 8,986 | 9,367 |
| | 5,563 | 6,303 | 7,009 | 7,671 | 8,284 | 8,840 | 9,333 | 9,758 | | 10,391 | 10,596 |
| 5,382 | 6,346 | | 8,102 | 8,834 | 9,435 | 9,889 | 10,184 | 10,317 | 10,292 | 10,122 | |
| 5,912 | 7,076 | 8,150 | 9,060 | 9,746 | 10,169 | | 10,178 | 9,802 | | 8,540 | 7,801 |
| 6,397 | 7,763 | 8,963 | | | 10,451 | 10,112 | | 8,535 | 7,568 | 6,693 | 6,056 |
| | 8,412 | 9,702 | 10,506 | 10,705 | 10,301 | 9,413 | 8,253 | 7,090 | 6,189 | | 5,935 |
| 3,320 | 3,749 | | | 4,811 | 5,116 | 5,404 | 5,677 | 5,938 | 6,188 | 6,428 | |
| 4,124 | 4,708 | 5,265 | | 6,310 | 6,803 | 7,277 | 7,732 | 8,168 | 8,587 | 8,986 | 9,367 |
| | 5,563 | 6,303 | 7,009 | 7,671 | 8,284 | 8,840 | 9,333 | 9,758 | | 10,391 | 10,596 |
| 5,382 | 6,346 | | 8,102 | 8,834 | 9,435 | 9,889 | 10,184 | 10,317 | 10,292 | 10,122 | |
| 5,912 | 7,076 | 8,150 | 9,060 | 9,746 | 10,169 | | 10,178 | 9,802 | | 8,540 | 7,801 |
| 6,397 | 7,763 | 8,963 | | | 10,451 | 10,112 | | 8,535 | 7,568 | 6,693 | 6,056 |
| | 8,412 | 9,702 | 10,506 | 10,705 | 10,301 | 9,413 | 8,253 | 7,090 | 6,189 | | 5,935 |

Таблица 2

**Зависимость ошибки заполнения пропусков табл. 1 от числа циклов и длины обучающей последовательности, % ($E_{indirect}$ для двухслойной НС CMAC, $E_{direct}$ для классической НС CMAC)**

| $N_{cycle}$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 50 | 100 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $N_{learn}$ | 168 | 336 | 504 | 672 | 840 | 1008 | 1176 | 1334 | 1512 | 1680 | 8400 | 16800 |
| $E_{indirect}$ | 11,023 | 10,265 | 8,842 | 8,285 | 8,076 | 7,547 | 7,104 | 7,268 | 7,041 | 6,888 | 5,970 | 5,868 |
| $E_{direct}$ | 13,568 | 10,578 | 11,167 | 8,021 | 10,432 | 9,829 | 10,766 | 8,096 | 9,541 | 10,530 | 7,572 | 9,705 |

от значения обобщающего параметра ρ*. Для рассматриваемой таблицы в соответствии с результатами, приведенными в работе [7], $M(\rho^* = 2) = 162$, $M(\rho^* = 4) = 100$, $M(\rho^* = 8) = 72$ и $M(\rho^* = 16) = 64$. Число уравнений системы $N_{exist} = 138$. Число пропущенных данных равно $N_{gap} = 30$. Несмотря на то, что при $\rho^* = 2$ размерность вектора памяти нейронной сети $M = 162 > N_{exist} = 138$, система (8) является переопределенной, поскольку в силу структуры НС CMAC ее 66 коэффициентов вектора памяти зависят только от начальных условий и не изменяются в процессе обучения. Фактически размерность вектора памяти при $\rho^* = 2$ равна $M = 162 - 66 = 96 < N_{exist}$. Следовательно, для всех возможных значений ρ* размерность $M$ вектора памяти $w$ меньше числа уравнений $N_{exist} = 138$ системы (8), т. е. эта система является переопределенной, и единственное решение возможно лишь в частном случае. В общем случае процедура обучения (3) приводит к тому, что точки $w[n]$, последовательно проектируясь на гиперплоскости (8), входят в зону их пересечения и совершают в ней хаотические движения в силу случайного характера выбора уравнения из системы (8). Результатом этого и является хаотический характер ошибки обучения $E_{direct}$, представленный в табл. 2. Отметим, что практически монотонный характер поведения косвенной ошибки $E_{indirect}$ (табл. 2) связан с тем, что при его вычислении использовалась двухслойная НС CMAC, описанная далее в разд. 2.

Обойти характер хаотического движения вектора памяти НС CMAC, и следовательно, и хаотическое движение ошибки заполнения пропусков в таблице можно нахождением центра зоны, в которой вектор $w[n]$ совершает хаотические движения. Далее будут представлены два способа решения этой задачи.

## 2. Двухслойная НС CMAC в задаче повышения точности восстановления пропущенных числовых данных в таблице

В работе [4] было показано, что помехи измерений могут быть подавлены с помощью введения второго слоя в НС CMAC, в котором оптимальный вектор памяти $w^*$ вычисляется осреднением по времени оценок $w[n]$, получаемых с помощью классического алгоритма обучения (3). Этот же подход можно распространить и на решение задачи заполнения пропусков в таблице с помощью НС CMAC, если рассматривать несовместную систему уравнений (8) как систему, в которой имеются помехи измерений.

С этой целью к существующей НС CMAC добавляется второй слой, предназначенный для вычисления и хранения осредненных по времени компонент вектора памяти $\tilde{w}[n]$. При этом в силу свойств нейронной сети CMAC на каждом шаге измерений процедуре осреднения подвергаются только ρ* активных компонент вектора памяти первого слоя $w[n]$. Процесс осреднения начинается с некоторого ненулевого шага, когда заканчивается переходной процесс, и оценки $w[n]$ входят в зону решения. В этом случае каждая составляющая вектора памяти второго слоя $\tilde{w}[n]$ стремится к константе, следовательно, и весь вектор $\tilde{w}[n]$ также стремится к постоянному значению $\tilde{w}^*$, близкому к центру области решения.

Восстановление пропущенных данных теперь осуществляется с помощью двухслойной НС CMAC при оптимальном значении обобщающего параметра ρ* и соответствующего вектора памяти сети $\tilde{w}^*$, при значениях которых косвенная ошибка заполнения пропусков таблицы $E_{indirect}(\rho^*)$ достигает минимального значения.

Эффект введения второго слоя в нейронную сеть CMAC иллюстрируют результаты компьютерного моделирования, приведенные в разд. 4.

## 3. НС CMAC на основе метода наименьших квадратов для повышения точности восстановления пропущенных данных в таблице

Другим способом нахождения центра зоны решения является метод наименьших квадратов, в соответствии с которым находится решение несовместной системы уравнений (8). Такой подход был предложен в работах [4, 5] и достаточно подробно описан в работе [6]. В соответствии с этим методом оценку вектора памяти НС CMAC можно найти из условия минимума функционала по вектору памяти $w$ при фиксированном значении обобщающего параметра ρ*:

$$J(w, \rho^*) = \sum_{i=1}^{N_{exist}} (y_{exist}[i] - a^{\mathrm{T}}(x_{exist}[i])w), \qquad (9)$$

из которого следует уравнение для нахождения оптимального значения вектора памяти НС CMAC

$$\left( \sum_{i=1}^{N_{exist}} a(x_{exist}[i])a^{\mathrm{T}}(x_{exist}[i]) \right) w = \sum_{i=1}^{N_{exist}} y_{exist}[i]a(x_{exist}[i]). \qquad (10)$$

К сожалению, в силу структуры НС CMAC, часть компонент векторов $a(x_{exist}[i])$, $i = 1, N_{exist}$ всегда равна нулю, поэтому матрица

$$D_{exist} = \sum_{i=1}^{N_{exist}} a(x_{exist}[i])a^{\mathrm{T}}(x_{exist}[i]) \qquad (11)$$

не является положительно определенной (часть ее диагональных элементов равна нулю), а решение система (10) относится к классу некорректных задач. Такие задачи решают либо введением параметра регуляризации [9], либо с помощью псевдообращения матрицы $D(N_{exist})$ [10]. В данной работе для решения задачи используется параметр регуляризации $pr$, который позволяет не только решить систему (10), но и оптимизировать решение по значению параметра регуляризации. Введение параметра ре-

гуляризации $pr$ приводит к тому, что вместо системы (10) ищется решение $w^*$ системы (12):

$$D_{exist} w = b_{exist}, \qquad (12)$$

где

$$D_{exist} = \sum_{i=1}^{N_{exist}} a(x_{exist}[i]) a^{\mathrm{T}}(x_{exist}[i]) + pr \cdot I,$$

$$b_{exist} = \sum_{i=1}^{N_{exist}} y_{exist}[i] a(x_{exist}[i]), \qquad (13)$$

$I$ — единичная матрица.

Программная реализация решения системы (12) основана на методе квадратного корня [11], который заключается в факторизации симметричной квадратной матрицы $D_{exist}$ (13) в виде произведения двух транспонированных друг к другу треугольных матриц $S_{exist}$, $S^{\mathrm{T}}_{exist}$: $D_{exist} = S_{exist} \cdot S^{\mathrm{T}}_{exist}$. Отметим, что размерность квадратной матрицы $D_{exist}$ равна размерности $M$ вектора памяти НС CMAC, верхняя оценка которой не превышает $x^2_{\max}$ (4).

Существенными особенностями матрицы $D_{exist}$ являются ее большая размерность при больших размерах таблицы, сильная разреженность (всего порядка $M \cdot \rho^*$ ненулевых элементов) и ленточная структура. Эти свойства порождены алгоритмом вычисления адресов активных ячеек памяти [7]. Учет этих свойств позволяет на порядок уменьшить время решения задачи (12).

При машинной реализации выражений (13) эффективно использовать рекуррентные уравнения относительно матрицы $D_{exist}$ и вектора $b_{exist}$:

$$D_{exist}[n] = D_{exist}[n-1] + a(x[n]) \cdot a^{\mathrm{T}}(x[n]),$$

$$b_{exist}[n] = b_{exist}[n-1] + a(x[n]) \cdot y[n],$$

$$n = 0, 1, 2, \ldots, N_{exist}, \qquad (14)$$

где $D_{exist}[0] = pr \cdot I$, $b_{exist}[0] = 0$.

Результаты применения метода наименьших квадратов продемонстрированы в разд. 4.

Обобщающий параметр $\rho^*$ принимает небольшое число целочисленных значений (в рассматриваемом случае это число равно 4), а параметр регуляризации $pr$ — величина положительная, $pr > 0$. Косвенная ошибка заполнения пропусков таблицы теперь является функцией двух переменных $E_{indirect}(\rho^*, pr)$. Минимизация функции $E_{indirect}(pr, \rho^*)$ по переменной $\rho^*$ осуществляется перебором ее значений, а по переменной $pr$ — методом последовательных приближений.

## 4. Результаты экспериментальных исследований

Экспериментальные исследования алгоритмов восстановления пропусков в таблице проведены в среде программирования Delphi на языке Object Pascal. В качестве объекта исследования здесь выбрана табл. 1 с пропусками, которая была также одним из объектов исследования, выполненного в работе [1]. Табл. 1 получена случайным удалением элементов из табл. 3, не содержащей пропуски. В этой таблице серым цветом выделены элементы таблицы, отсутствующие в табл. 1.

***Двухслойная НС CMAC, режим on-line.*** В разд. 1 было показано, что для восстановления пропущенных числовых данных таблицы необходимо вычислить косвенную ошибку $E_{indirect}(n, \rho^*)$ заполнения пропусков при различной длине обучающей последовательности $n$ и различных значениях обобщающего параметра $\rho^*$. Оптимальным значением параметра $n$ считается такое число, при котором процесс обучения закончился и значение $E_{indirect}(n, \rho^*)$ практически не меняется с увеличением длины обучающей последовательности $n$. При оптимальном значении обобщающего параметра $\rho^*$ ошибка $E_{indirect}(n, \rho^*)$ после завершения процесса обучения минимальна по $\rho^*$.

Таблица 3

**Исходная таблица 14 × 12 без пропусков**

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 3,320 | 3,749 | 4,133 | 4,485 | 4,811 | 5,116 | 5,404 | 5,677 | 5,938 | 6,188 | 6,428 | 6,660 |
| 4,124 | 4,708 | 5,265 | 5,798 | 6,310 | 6,803 | 7,277 | 7,732 | 8,168 | 8,587 | 8,986 | 9,367 |
| 4,794 | 5,563 | 6,303 | 7,009 | 7,671 | 8,284 | 8,840 | 9,333 | 9,758 | 10,111 | 10,391 | 10,596 |
| 5,382 | 6,346 | 7,263 | 8,102 | 8,834 | 9,435 | 9,889 | 10,184 | 10,317 | 10,292 | 10,122 | 9,826 |
| 5,912 | 7,076 | 8,150 | 9,060 | 9,746 | 10,169 | 10,311 | 10,178 | 9,802 | 9,233 | 8,540 | 7,801 |
| 6,397 | 7,763 | 8,963 | 9,865 | 10,375 | 10,451 | 10,112 | 9,433 | 8,535 | 7,568 | 6,693 | 6,056 |
| 6,849 | 8,412 | 9,702 | 10,506 | 10,705 | 10,301 | 9,413 | 8,253 | 7,090 | 6,189 | 5,765 | 5,935 |
| 3,320 | 3,749 | 4,133 | 4,485 | 4,811 | 5,116 | 5,404 | 5,677 | 5,938 | 6,188 | 6,428 | 6,660 |
| 4,124 | 4,708 | 5,265 | 5,798 | 6,310 | 6,803 | 7,277 | 7,732 | 8,168 | 8,587 | 8,986 | 9,367 |
| 4,794 | 5,563 | 6,303 | 7,009 | 7,671 | 8,284 | 8,840 | 9,333 | 9,758 | 10,111 | 10,391 | 10,596 |
| 5,382 | 6,346 | 7,263 | 8,102 | 8,834 | 9,435 | 9,889 | 10,184 | 10,317 | 10,292 | 10,122 | 9,826 |
| 5,912 | 7,076 | 8,150 | 9,060 | 9,746 | 10,169 | 10,311 | 10,178 | 9,802 | 9,233 | 8,540 | 7,801 |
| 6,397 | 7,763 | 8,963 | 9,865 | 10,375 | 10,451 | 10,112 | 9,433 | 8,535 | 7,568 | 6,693 | 6,056 |
| 6,849 | 8,412 | 9,702 | 10,506 | 10,705 | 10,301 | 9,413 | 8,253 | 7,090 | 6,189 | 5,765 | 5,935 |

На рис. 1 показана зависимость косвенной ошибки заполнения пропусков от длины обучающей последовательности при четырех возможных значениях обобщающего параметра $\rho^*$. Из рис. 1 следует, что для двухслойной НС CMAC ошибка $E_{indirect}(n, \rho^*)$ убывает монотонно, процесс обучения заканчивается при длине обучающей последовательности приблизительно $n = 2^{14} = 16\,384$ и с увеличением ее длины практически не изменяется. Значения ошибки $E_{indirect}(n, \rho^*)$ в конце про-

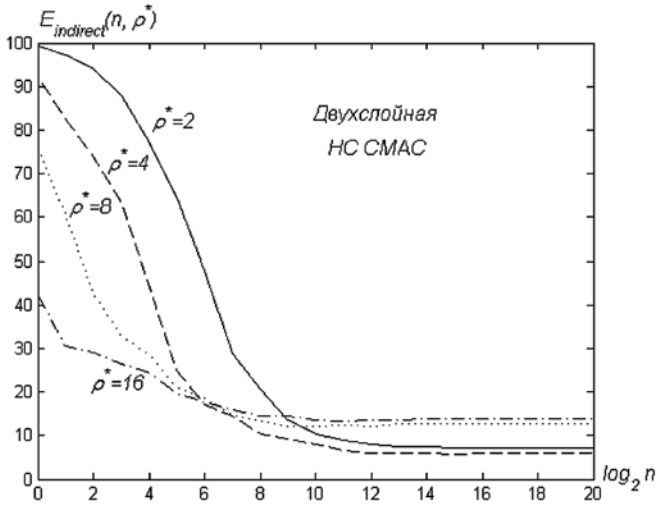


**Рис. 1. Зависимости косвенной ошибки $E_{indirect}(n, \rho^*)$ (%) заполнения представленных в табл. 1 пропусков в функции длины обучающей последовательности $n$ при различных значениях обобщающего параметра ρ* для двухслойной НС СМАС**

цесса обучения при $n = 2^{20}$ для различных значений обобщающего параметра ρ* равны

$$E_{indirect}(2^{20}, 2) = 7{,}218,$$
$$E_{indirect}(2^{20}, 4) = 5{,}878,$$
$$E_{indirect}(2^{20}, 8) = 12{,}779,$$
$$E_{indirect}(2^{20}, 16) = 13{,}771.$$

Следовательно, оптимальными значениями параметров двухслойной НС СМАС для восстановления пропущенных числовых данных табл. 1 являются ρ* = 4 и $n > 16\,384$.

Наличие информации о числовых значениях пропущенных данных, представленных в табл. 3, позволяет вычислить истинные ошибки $E_{direct}(n, \rho^*)$ при различных значениях ρ* в конце обучения при $n = 2^{20}$. Этими значениями являются:

$$E_{direct}(2^{20}, 2) = 6{,}794,$$
$$E_{direct}(2^{20}, 4) = 7{,}310,$$
$$E_{direct}(2^{20}, 8) = 13{,}381,$$
$$E_{direct}(2^{20}, 16) = 13{,}879.$$

Поскольку минимальное значение косвенной ошибки $E_{indirect}(2^{20}, 4) = 5{,}878$ имело место при ρ* = 4, то и восстановление пропусков выполняется при этом значении обобщающего параметра. Следовательно, истинная ошибка восстановления пропущенных числовых данных таблицы для двухслойной сети оказывается равной $E_{direct}(2^{20}, 4) = 7{,}310$, что несколько больше истинной ошибки $E_{direct}(2^{20}, 2) = 6{,}794$ при ρ* = 2.

Из сравнения полученных результатов с результатами, приведенными в табл. 2, следует, что двухслойная НС СМАС решает задачу восстановления пропусков стабильнее и точнее, чем классическая НС СМАС. Восстановление пропущенных данных с помощью двухслойной НС СМАС отражает табл. 4.

**НС СМАС на основе метода наименьших квадратов, режим of-line.** Как и в случае двухслойной НС СМАС, здесь также необходимо вычислить косвенную ошибку заполнения числовых пропусков, которая теперь является функцией параметра регуляризации $pr$ и обобщающего параметра ρ*, $E_{indirect} = E_{indirect}(pr, \rho^*)$. Минимальное значение функции $E_{indirect}(pr, \rho^*)$ определит те значения параметров $pr$ и ρ*, при которых будет выполнено обучение НС СМАС по методу наименьших квадратов и заполнение пропусков в таблице.

На рис. 2 показана зависимость косвенной ошибки заполнения пропусков от значения параметра регуляризации при четырех возможных значениях обобщающего параметра ρ*. Функции $E_{indirect}(pr, \rho^*)$ при фиксированном значении ρ* являются одноэкстремальными, поэтому определить значение параметра регуляризации при фиксированном значении ρ* можно методом последовательных приближений, начиная поиск от точек, близких к нулю, например $pr[0] = 10^{-6}$. Минимальные значения косвенной ошибки $E_{indirect}(pr, \rho^*)$ при фиксированных параметрах ρ* равны

$$E_{indirect}(0{,}000346, 2) = 4{,}419, (\log_{10}0{,}000346 = -3{,}461),$$
$$E_{indirect}(0{,}0392, 4) = 5{,}840, (\log_{10}0{,}0392 = -1{,}407),$$
$$E_{indirect}(1{,}246, 8) = 11{,}776, (\log_{10}1{,}246 = 0{,}096), \quad (15)$$
$$E_{indirect}(1{,}572, 16) = 13{,}505, (\log_{10}1{,}572 = 0{,}196).$$

Таблица 4

**Восстановленная табл. 1: двухслойная НС СМАС, длина обучающей последовательности $n$ = 1048576, значение обобщающего параметра ρ* = 4, истинная ошибка восстановления $E_{direct}$(1048576,4) = 7,310 %**

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 3,320 | 3,749 | 4,094 | 4,451 | 4,811 | 5,116 | 5,404 | 5,677 | 5,938 | 6,188 | 6,428 | 6,902 |
| 4,124 | 4,708 | 5,265 | 5,659 | 6,310 | 6,803 | 7,277 | 7,732 | 8,168 | 8,587 | 8,986 | 9,367 |
| 4,290 | 5,563 | 6,303 | 7,009 | 7,671 | 8,284 | 8,840 | 9,333 | 9,758 | 10,622 | 10,391 | 10,596 |
| 5,382 | 6,346 | 7,555 | 8,102 | 8,834 | 9,435 | 9,889 | 10,184 | 10,317 | 10,292 | 10,122 | 8,438 |
| 5,912 | 7,076 | 8,150 | 9,060 | 9,746 | 10,169 | 11,335 | 10,178 | 9,802 | 9,589 | 8,540 | 7,801 |
| 6,397 | 7,763 | 8,963 | 9,705 | 9,787 | 10,451 | 10,112 | 9,125 | 8,535 | 7,568 | 6,693 | 6,056 |
| 7,599 | 8,412 | 9,702 | 10,506 | 10,705 | 10,301 | 9,413 | 8,253 | 7,090 | 6,189 | 7,971 | 5,935 |
| 3,320 | 3,749 | 4,692 | 5,409 | 4,811 | 5,116 | 5,404 | 5,677 | 5,938 | 6,188 | 6,428 | 6,615 |
| 4,124 | 4,708 | 5,2265 | 6,240 | 6,310 | 6,803 | 7,277 | 7,732 | 8,168 | 8,587 | 8,986 | 9,367 |
| 4,425 | 5,563 | 6,303 | 7,009 | 7,671 | 8,284 | 8,840 | 9,333 | 9,758 | 9,257 | 10,391 | 10,596 |
| 5,382 | 6,346 | 7,429 | 8,102 | 8,834 | 9,435 | 9,889 | 10,184 | 10,317 | 10,292 | 10,122 | 10,213 |
| 5,912 | 7,076 | 8,150 | 9,060 | 9,746 | 10,169 | 10,147 | 10,178 | 9,802 | 9,090 | 8,540 | 7,801 |
| 6,397 | 7,763 | 8,963 | 9,714 | 9,877 | 10,451 | 10,112 | 9,513 | 8,535 | 7,568 | 6,693 | 6,056 |
| 5,324 | 8,412 | 9,702 | 10,506 | 10,705 | 10,301 | 9,413 | 8,253 | 7,090 | 6,189 | 6,163 | 5,935 |

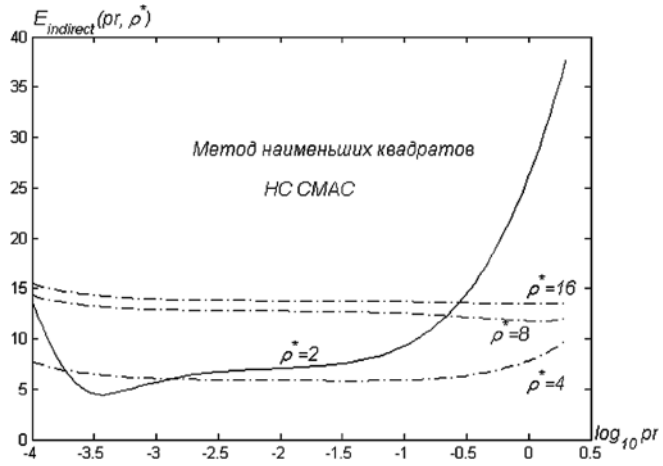


**Рис. 2. Зависимости косвенной ошибки $E_{indirect}(pr, \rho^*)$ (%) заполнения представленных в табл. 1 пропусков в функции параметра регуляризации *pr* и обобщающего параметра ρ* для НС СМАС по методу наименьших квадратов**

Из выражений (15) следует, что минимальное значение косвенная ошибка $E_{indirect}(pr, \rho^*)$ принимает при значениях параметров $pr = 0{,}000346$, $\rho^* = 2$, которые являются оптимальными для НС СМАС на основе метода наименьших квадратов при восстановлении пропущенных числовых данных табл. 1.

Как и в случае двухслойной НС СМАС, наличие информации о числовых значениях пропущенных данных, представленных табл. 3, позволяет вычислить истинные ошибки $E_{direct}(pr, \rho^*)$ при различных значениях ρ* и соответствующих оптимальных значениях параметров регуляризации *pr*:

$$\begin{aligned} &E_{direct}(0{,}000346,\ 2) = 5{,}142,\\ &E_{direct}(0{,}0392,\ 4) = 7{,}274,\\ &E_{direct}(1{,}246,\ 8) = 12{,}816,\\ &E_{direct}(1{,}572,\ 16) = 13{,}762. \end{aligned} \qquad (16)$$

Сравнивая значения косвенных (15) и истинных (16) ошибок для рассматриваемого метода, можно сделать вывод о том, что косвенные ошибки достаточно хорошо характеризуют истинные ошибки, и, следовательно, косвенные ошибки являются адекватным инструментом для оптимального заполнения пропусков в таблицах. Учитывая, что для косвенной ошибки $E_{indirect}$ оптимальными параметрами являются $pr = 0{,}000346$, $\rho^* = 2$, истинная ошибка (15) при этих значениях параметров равна $E_{direct}(0{,}000346, 2) = 5{,}142$. Эта ошибка меньше оптимальной ошибки $E_{direct}(2^{20}, 4) = 7{,}310$ двухслойной НС СМАС, и конечно меньше ошибки классической НС СМАС (см. табл. 2).

Точность восстановления пропущенных данных с помощью НС СМАС на основе метода наименьших квадратов характеризует табл. 5. Восстановление выполнено при оптимальных значениях параметров сети.

## Заключение

В работе показано, что двухслойная НС СМАС и НС СМАС на основе метода наименьших квадратов решают задачу заполнения пропусков в таблицах с точностями, которые выше, чем у классической НС СМАС. Из двух модификаций НС СМАС предпочтение следует отдать сети на основе метода наименьших квадратов, поскольку она решает исследуемую задачу с меньшей ошибкой восстановления пропущенных числовых данных.

Отметим также, что при восстановлении пропущенных данных в таблицах большой размерности вычисление косвенной ошибки $E_{indirect}$ следует выполнять не для всех данных таблицы, а только для их части. Выбор размера и способа формирования этой части выборки представляет отдельную задачу типа определения размера репрезентативной выборки.

Таблица 5

**Восстановленная табл. 1, НС СМАС на основе метода наименьших квадратов, значение параметра регуляризации *pr* = 0,000346, значение обобщающего параметра ρ* = 2, истинная ошибка восстановления $E_{direct}(0{,}000346,\ 2) = 5{,}142$ %**

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 3,320 | 3,749 | 4,306 | 4,269 | 4,811 | 5,116 | 5,404 | 5,677 | 5,938 | 6,188 | 6,428 | 5,966 |
| 4,124 | 4,708 | 5,265 | 5,800 | 6,310 | 6,803 | 7,277 | 7,732 | 8,168 | 8,587 | 8,986 | 9,367 |
| 4,897 | 5,563 | 6,303 | 7,009 | 7,671 | 8,284 | 8,840 | 9,333 | 9,758 | 10,317 | 10,391 | 10,596 |
| 5,382 | 6,346 | 7,334 | 8,102 | 8,834 | 9,435 | 9,889 | 10,184 | 10,317 | 10,292 | 10,122 | 9,194 |
| 5,912 | 7,076 | 8,150 | 9,060 | 9,746 | 10,169 | 9,981 | 10,178 | 9,802 | 9,582 | 8,540 | 7,801 |
| 6,397 | 7,763 | 8,963 | 10,225 | 11,030 | 10,451 | 10,112 | 8,576 | 8,535 | 7,568 | 6,693 | 6,056 |
| 6,929 | 8,412 | 9,702 | 10,506 | 10,705 | 10,301 | 9,413 | 8,253 | 7,090 | 6,189 | 6,665 | 5,935 |
| 3,320 | 3,749 | 4,912 | 5,109 | 4,811 | 5,116 | 5,404 | 5,677 | 5,938 | 6,188 | 6,428 | 6,625 |
| 4,124 | 4,708 | 5,265 | 5,491 | 6,310 | 6,803 | 7,277 | 7,732 | 8,168 | 8,587 | 8,986 | 9,367 |
| 4,646 | 5,563 | 6,303 | 7,009 | 7,671 | 8,284 | 8,840 | 9,333 | 9,758 | 9,935 | 10,391 | 10,596 |
| 5,382 | 6,346 | 7,352 | 8,102 | 8,834 | 9,435 | 9,889 | 10,184 | 10,317 | 10,292 | 10,122 | 10,387 |
| 5,912 | 7,076 | 8,150 | 9,060 | 9,746 | 10,169 | 10,424 | 10,178 | 9,802 | 8,856 | 8,540 | 7,801 |
| 6,397 | 7,763 | 8,963 | 9,918 | 10,150 | 10,451 | 10,112 | 9,670 | 8,535 | 7,568 | 6,693 | 6,056 |
| 7,732 | 8,412 | 9,702 | 10,506 | 10,705 | 10,301 | 9,413 | 8,253 | 7,090 | 6,189 | 5,375 | 5,935 |

**Список литературы**


1. **Аведьян Э. Д., Луганский В. Э.** Подход к задаче заполнения числовых пропусков в таблицах и строках, основанный на многослойной нейронной сети и нейронной сети CMAC // Информационные технологии. 2014. № 1. С. 58—66.

2. **Albus J. S.** A new approach to manipulator control: the cerebellar model articulation controller // ASME Trans., J. Dynamic Systems, Measurement and Control. 1975. V. 97, N 3. P. 220—227.

3. **Albus J. S.** Data storage in the cerebellar model articulation controller (CMAC) // ASME Trans., J. Dynamic Systems, Measurement and Control. 1975. V. 97, N 3. P. 228—233.

4. **Аведьян Э. Д.** Ассоциативная нейронная сеть CMAC. Часть II. Процессы обучения, ускоренное обучение, влияние помех, устранение влияния помех в двухслойной сети // Информационные технологии. 1997. № 6. С. 16—27.

5. **Horváth G.** Kernel CMAC: an Efficient Neural Network for Classification and Regression // Acta Polytechnica Hungarica. 2006. Vol. 3, N 1. P. 5—20.

6. **Аведьян Э. Д., Пантюхин Д. В.** Методы подавления помех в нейронной сети CMAC // Информатизация и связь. 2011. № 6. С. 6—12.

7. **Аведьян Э. Д.** Ассоциативная нейронная сеть CMAC. Часть I. Структура, объем памяти, обучение и базисные функции // Информационные технологии. 1997. № 5. С. 6—14.

8. **Аведьян Э. Д., Пантюхин Д. В.** Алгоритм нелинейного преобразования аргументов в нейронной сети CMAC // Информационные технологии. 2011. № 1. С. 64—72.

9. **Тихонов А. Н., Арсенин В. Я.** Методы решения некорректных задач. М.: Наука, 1986. 287 с.

10. **Гантмахер Ф. Р.** Теория матриц. М.: Наука, 1967. 576 с.

11. **Фаддеев Д. К., Фаддеева В. Н.** Вычислительные методы линейной алгебры. ФМ. М.-Л.: 1963. 734 с.

УДК 004.414.23:004.272.44

**А. В. Мышев,** канд. физ.-мат. наук, доц.,
Национальный исследовательский ядерный университет МИФИ —
Обнинский институт атомной энергетики, e-mail: mishev@iate.obninsk.ru

# Архитектура виртуальной потоковой вычислительной системы на основе информационной модели нейросети

*В работе раскрывается архитектура (структура) виртуальной потоковой вычислительной системы нового поколения, которая позволяет реализовать новую парадигму вычислительного интеллекта и осуществить доступ к этим новым интеллектуальным вычислительным возможностям на основе нетрадиционных нейросетевых технологий. Математические и логические основы парадигмы позволяют реализовать новые преимущества обозначенных вычислительных систем на основе информационных моделей нейросети и нетрадиционных принципов разработки и реализации различных форм компьютинга, которые дают возможность осуществить синтез влияния механизмов параллелизма, виртуализации и интеллектуализации на информационную динамику объектов среды вычислений.*

***Ключевые слова:*** *виртуальные потоковые вычислительные системы, компьютинг, среда вычислений, информационная модель нейросети, многослойная модульная схема*

**A. V. Myshev**

# Architecture of Virtual Flow Computing System Streaming Based on Information Model of Neural Networks

*In the work the architecture (structure) of a virtual flow computing system of new generation computer system, which allows for a new paradigm of computational intelligence and make access to these new intelligent computing power through non-conventional neural networks is considered. Mathematical and logical foundations of the paradigm allow to realize the benefits of new computer systems identified on the basis of information models of neural networks and innovative design principles and implementation of various forms of computing that allow the synthesis of the impact of the mechanisms of parallelism, virtualization and intellectualization of information on the dynamics of the objects of the computing environment.*

***Keywords:*** *virtual computer systems streaming, computing, computing environment, information model of neural networks, multi-layer modular design*

## Введение

Известные разновидности потоковых вычислительных систем (ВС) основаны на модели "чистого потока данных", это статистические ВС и динамические ВС [1]. Общие принципы организации вычислительных процессов в таких ВС направлены на достижение высокой степени параллелизма и обеспечение высокого коэффициента загрузки аппаратных средств, которые являются основными факторами, определяющими эффективность ВС как по ряду технических и алгоритмических характеристик, так и в целом.

Разработка и реализация вычислительных процессов для технологий имитационного моделирования различных задач и обработка информации в

таких системах на основе моделей последовательных и параллельных алгоритмов осуществляется на вычислениях, которые выполняются посредством строгой логической схемы последовательности дискретных шагов, включая точные данные (точность определяется числом значащих цифр или символов). Даже при таких строгих допущениях ошибка всего лишь в единственном бите в последовательности операндов или в данных часто делает результат вычисления бесполезным или размытым. Мощные усилия, направленные на устранение или минимизацию таких ошибок средствами аппаратного и программного обеспечения, в основном связаны с решением дилеммы: точность вычислительной системы и успешность предотвращения неустойчивого исполнения, обусловленного такой точностью, которые являются ключевыми величинами в традиционной вычислительной индустрии.

Точность и надежность в традиционных последовательных и параллельных вычислительных системах очень далеки от надежной организации вычислительных процессов, синтеза и обработки информации в интеллектуальных живых системах. Остается открытым вопрос о выборе критериев оценки эффективности и надежности алгоритмов, а также и многие другие.

Традиционные потоковые вычислительные системы и вычислительные системы для решения потоковых задач функционально являются системами, имеющими высокую вычислительную эффективность как по техническим характеристикам, так и по алгоритмическому обеспечению. Эффективность достигается в этом случае за счет высокой степени параллелизма аппаратного взаимодействия ее элементов и методологии синтеза параллельных схем и алгоритмов вычислительных процессов. Интеллектуальные возможности и интеллектуальная эффективность таких вычислительных систем как интеллектуального инструмента в технологиях организации вычислительных процессов, имитационного моделирования и обработки информации в целях достижения точных и устойчивых результатов, когда диффузия информации локализована в пространственно-временных масштабах, не рассматривали и не исследовали. Также открыты такие вопросы, которые связаны с разработкой информационных моделей потоковых вычислений в условиях модельной и алгоритмической замкнутости, обмена информацией, ограничениями среды вычислений и информационной неопределенности.

В основе причин, порождающих ошибки, обусловленных ключевыми величинами (точность и неустойчивость) вычислительных процессов, лежит тот факт, что традиционно практикуемые технологии последовательных и потоковых вычислений весьма неустойчивы. Эта неустойчивость во многом обусловлена тем, что технологии вычислительных процессов потоковых и скалярных вычислений реализуются без учета информационной динамики объектов среды вычислений, которая определяет событийность пространственно—временных и информационных условий контролируемости вычислительных процессов. В настоящее время достоверно неизвестно, как представляются, хранятся и обрабатываются данные в живых интеллектуальных системах обработки информации, как в них организованы вычислительные механизмы и многие другие вычислительные аспекты. Работы по изучению обозначенных вопросов-секретов ранее и в настоящее время в основном сфокусированы на исследовании биохимических, нейрофизиологических и структурных аспектов информационных процессов в живых нейросистемах.

Результаты проводимых исследований позволили идентифицировать лишь некоторые ограниченные вычислительные механизмы, но нет математического описания логических структур данных для представления информации в таких нейросистемах, не построена общая теория оперирования данными и вычисления в живых нейросистемах. Также не создана, что является наиболее важным, приемлемая общая теория, которая описывала бы вычислительные возможности интеллектуальных живых систем на основе информационных моделей передачи и обработки информации в нейросистемах, которые можно адаптировать для разработки и реализации интеллектуальных вычислительных систем, способных проявлять подобные характеристики обработки информации. Но тем не менее можно с высокой степенью достоверности утверждать, что живые интеллектуальные системы обработки информации реализуют свои функции, в том числе и вычислительные, в условиях замкнутости, ограничений, обмена (энергией и информацией) и неопределенности.

Обозначенные условия функционирования живых систем характерны и для потоковых вычислительных систем, которые в настоящее время не реализованы и не описаны ни в одной модели существующих вычислительных систем (реальных и абстрактных). Поэтому проблема расширения функциональных и интеллектуальных возможностей вычислительных систем нового поколения, повышения надежности их информационной работоспособности в условиях замкнутости, ограничений, обмена и неопределенности, обеспечения возможности создания сверхсложных интеллектуальных информационных виртуальных нейросетей, возможности реализации динамически расширяемой архитектуры виртуальной вычислительной системы, повышения информационной достоверности результатов вычислений и обработки данных является не только фундаментальной и актуальной в обозначенной области, но и требует решения новых научных и технических задач.

### Нейросетевая модель архитектуры виртуальной потоковой вычислительной системы

Информационная модель нейросети (рис. 1) как математический и логический прототип интеллектуальной вычислительной системы имеет важные вычислительные и информационные свойства механизмов вычислительных процессов и позволяет их определить и описать с учетом математических, динамических, информационных и метрологических аспектов, которые не являются очевидными и поддерживают новую парадигму вычислений и осуществления нетрадиционных форм компьютинга с учетом информационной динамики объектов среды вычислений [2, 3]. Эта парадигма имеет новые вычислительные преимущества на основе нетрадиционных принципов разработки и реализации различных форм компьютинга, которые позволяют синтезировать влияние механизмов параллелизма, виртуализации и интеллектуализации на информационную динамику объектов среды вычислений. А математический и логический аппарат парадигмы раскрывает архитектуру (структуру) виртуальной потоковой вычислительной системы, которая позволяет реализовать эту новую парадигму вычислительного интеллекта и осуществить доступ к новым интеллектуальным вычислительным возможностям. Обозначенные цели достигаются следующим способом.

Во-первых, на основе способа построения виртуальной потоковой вычислительной системы, согласно которому процессорные модули и модули памяти такой системы логически, функционально и информационно взаимодействуют между собой по тем же принципам, что и нейроны в сетевой информационной модели нейросети. Этот способ определяет организацию и реализацию этой виртуальной системы в виде многослойной модульной системы потоковой параллельной обработки данных, где имеется несколько слоев: слой процессорных элементов, слой многоуровневой виртуальной оперативной памяти для каждого модуля процессорного слоя и слой общей активной виртуальной памяти для всего процессорного слоя. Процессорный слой разбивается на виртуальные процессорные модули, а слой общей активной виртуальной памяти — на виртуальные модули памяти. Процессорные модули имеют сетевую модель организации информационных связей и выполняют следующие функции: интерфейса как между элементами слоев, так и с внешней средой, функции инициализации вычислительной системы, маршрутизации информации в системе, конфигурации архитектуры виртуальной потоковой вычислительной системы под структуру информационного графа решаемой задачи, восстановления результатов вычислительного процесса при отказе всей системы, отдельных слоев, модулей или элементов. Модули слоя виртуальной оперативной памяти выполняют функции хранения атрибутов модулей, команд, локальной системы координат информационной привязки и поверки результатов (промежуточных и конечных) вычислительных процессов в среде вычислений. Модули общей активной виртуальной памяти функционально и логически предназначены: 1) для хранения глобальной информационной системы координат проверки и поверки и отражения на ней результатов вычислительного процесса на каждом его шаге; 2) выполняют буферные функции в операциях тайлинга между модулями слоев, функции виртуальных портов и информационных шлюзов с внешним миром.

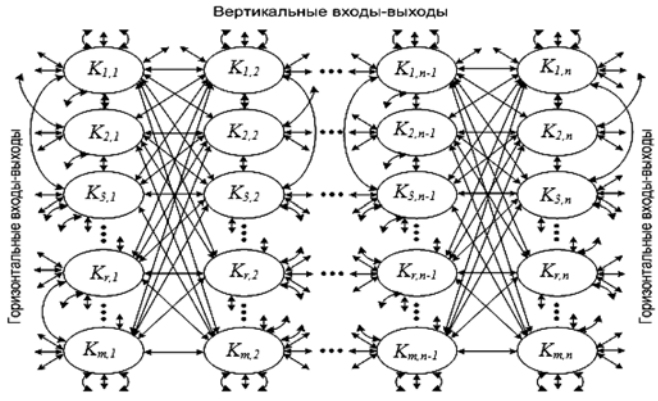


**Рис. 1. Информационная модель нейросети как математический и логический прототип архитектуры виртуальной потоковой вычислительной системы**

Согласно обозначенному способу процессорные модули логически и функционально объединяются и конфигурируются в виртуальную вычислительную структуру с геометрией и топологией связей сетевой информационной модели нейросети. Каждый процессорный модуль функционирует в соответствующей ему области информационного пространства координатной системы привязки и поверки результатов. Это обеспечивает высокую надежность работы всей вычислительной системы и достоверность получаемых результатов. Базовыми и образующими элементами модулей виртуальной оперативной памяти и общей виртуальной активной памяти являются виртуальные ячейки двух типов: активные и пассивные. Выполнение логических и арифметических операций в процессорных модулях реализуется по следующим принципам компьютинга: взаимодействия операндов среды вычислений с информационной виртуальной средой; виртуальной перспективы в операторе проектирования; связанности результатов вычислений — фрактальной и информационной.

Во-вторых, обозначенную цель достигают с помощью структуры виртуальной потоковой вычислительной системы, реализующей новую парадигму вычислительного интеллекта на основе сетевой информационной модели нейросети. Сетевая информационная модель нейросети, с одной стороны,

описывает структуру (архитектуру) виртуальной потоковой вычислительной системы в виде глобальной виртуальной информационной динамической сети, в которой характер взаимодействия между элементами на уровнях управления и обмена информацией определяется как процессы с локальным взаимодействием. А с другой стороны, элемент такой сети, формальный нейрон, определяется как локальная виртуальная вычислительная система, которая объединяет ресурсы процессорных элементов и памяти реальной физической вычислительной системы с жесткой или реконфигурируемой архитектурой, логическая и физическая функции которой заключаются в следующем. Во-первых, это локальная виртуальная вычислительная сеть, которая определяет и задает в физической среде реальной потоковой ВС вычислительный сервер-домен, т. е. пространство виртуальных адресов-идентификаторов для ее процессорных элементов и соответствующих им сегментов активной памяти. Во-вторых, виртуальный вычислительный сервер-домен задает не только область информационного определения в пространстве виртуальных адресов его активной памяти — его домен, но и определяет в информационных границах адресного пространства домена памяти общую числовую шкалу для вычислительного процесса и цену деления шкалы (числовая точность) — его диапазон. Элементами активной памяти являются виртуальные ячейки разных типов, отличающихся по функциональному признаку, логической организации и информационному наполнению.

Нейрон в обозначенной архитектуре (структуре) глобальной виртуальной потоковой вычислительной системы (ВПВС) определяется как образующий ее компонент, предназначение и функция которого состоят в следующем: 1) реализация для потоковой параллельной задачи ее выполнения в заданной области возможных значений и фиксированного кванта физического времени; 2) реализация статистической синхронизации вычислительного процесса в нейросети в пределах выделенного кванта реального времени. Нейроны объединяются в виртуальную вычислительную структуру, которая поддерживает вычислительный процесс с динамически изменяемой популяцией активных нейронов в фиксированный и конкретный квант физического времени. Активные нейроны могут реализовывать различные вычислительные процессы в пределах конкретного кванта физического времени, если взаимодействие между ними параметризовано алгоритмом задачи и средой вычислений.

В области активной памяти нейрона задается локальный сегмент (только в режиме чтения), в котором хранится система координат информационной привязки и поверки для результатов вычислений (конечных и промежуточных), отягощенных различного рода ошибками и информационной диффузией в вычислительных технологиях. Геометрическая интерпретация такой системы состоит в том, что она является базовой информационной координатной решеткой, на которой отражается динамика информационных процессов среды вычислений. Узлы этой решетки в широком смысле являются образным и символическим отражением геометрических и информационных свойств динамики процессов среды вычислений. Информационная сущность геометрии узлов на решетке определяет ее как сегмент памяти виртуальных пассивных ячеек фиксированной длины и с неизменяемой в них информацией, т. е. это сегмент констант в виде символьных цепочек. Такой сегмент задает область определения отображения результатов вычислительного процесса в информационном пространстве активной виртуальной памяти среды вычислений — их домен и область значений — диапазон. Введение описанной логической схемы организации памяти и вычислительного процесса определяет вычислительные системы с такой архитектурой как виртуальные потоковые вычислительные системы с переменными доменами-диапазонами. Если результаты вычислительного процесса в среде вычислений нейрона выходят за границы диапазона, то происходит взаимодействие с соседним нейроном, в информационный диапазон которого они попадают.

Вычислительные процессы в обозначенных системах квантуются по физическому времени. Длительность квантов может быть произвольной и определяется условиями статистической синхронизации и вычислительного процесса. В каждый квант физического времени число активных нейронов определяется результатом вычислений в предыдущий квант и условиями задачи.

Виртуальные ячейки активной памяти нейрона, которые не принадлежат системе координат информационной привязки и поверки, являются активными ячейками переменной длины с динамически изменяемой информацией в них, т. е. это рабочие виртуальные ячейки. Процесс взаимодействия символьных цепочек, хранящихся в пассивных и активных виртуальных ячейках, определяется как информационный процесс с локальным взаимодействием. Любая операция между ячейками разных типов задается в виде двух операндов — взаимодействия и проектирования. Первым шагом при выполнении операции является реализация процедуры информационного выравнивания символьных цепочек разной длины, которое определяется как взаимодействие со средой. Далее реализуется операция взаимодействия между символьными цепочками и формируется результат, который проектируется в узел базовой координатной решетки. Смысл операции проектирования состоит в следующем. Если информационное расстояние между символьной цепочкой результата операции и узлом

решетки меньше информационной окрестности узла, то результат находится в этом узле.

Структура виртуальной потоковой вычислительной системы, основанной на информационных моделях формальной нейросети и нейрона, в качестве физической платформы может иметь потоковую вычислительную систему (например, транспьютер и др.) или мультиконвейерную вычислительную систему (с жесткой или реконфигурируемой архитектурой) [4], состоит из процессорных элементов и блока многоуровневой виртуальной активной памяти. Такие виртуальные вычислительные системы поддерживают три фазы виртуализации: виртуальное пространство; изображение; среда вычислений.

Практические следствия использования сетевой информационной модели нейросети и нейрона в технологиях вычислительного интеллекта и новых формах осуществления компьютинга означают, что многие, долгое время недостижимые и неизвестные, свойства и когнитивные возможности интеллектуальных вычислительных систем обработки и анализа информации в условиях замкнутости, ограничений, обмена и неопределенности становятся доступными уже в готовом виде. Эти свойства и когнитивные возможности обозначенных систем включают в себя способность, подобную живым информационным системам, создания новых форм интеллектуальных потоковых виртуальных параллельных вычислений, новых форм надежности их работоспособности в условиях замкнутости, ограничений, обмена и неопределенности, нового класса компьютерных вычислительных систем.

### Многослойная модульная схема реализации виртуальной потоковой вычислительной системы

Физическая и виртуальная реализация архитектуры виртуальной потоковой ВС, основанной на сетевой информационной модели нейросети и нейрона, осуществляется по принципу многослойной модульной вычислительной системы. Организация такой системы включает несколько слоев (рис. 2): слой процессорных элементов (ПЭ) (1-й слой $n1$); слой активной виртуальной памяти для каждого ПЭ или модуля ПЭ (2-й слой $n2$); слой общей активной виртуальной памяти, доступной для всех ПЭ (3-й слой $n3$). Такая схема организации многослойной модульной вычислительной системы может быть реализована как на физическом и логическом уровне, так и на виртуальном. Число слоев такой системы может быть большим (но не менее трех), в зависимости от сложности решаемой задачи, требований надежности результата, условий организации вычислительного процесса и др. Каждый слой содержит элементы (физические и виртуальные) по функциональному и логическому назначению одного типа для каждого слоя (1-го, 2-го, 3-го). Интерфейс между элементами слоев организуется на логическом и физическом уровнях, с использованием входов—выходов: 1-й слой — $i|o1$; 2-й слой — $i|o2$; 3-й слой — $i|o3$. Процессорные элементы выполняют функции интерфейса как между элементами слоев, так и с внешней средой, инициализации вычислительной системы, маршрутизации информации в системе, конфигурации топологии нейросети как логического образа архитектуры виртуальной потоковой вычислительной системы под структуру информационного графа решаемой задачи, восстановления результатов вычислительного процесса при отказе всей системы, отдельных слоев или элементов. Процессорный слой $n1$ состоит из $m$ ПЭ с произвольной топологией связей и необязательно унифицированным логическим пулингом взаимодействия между ними. В этом случае ПЭ является функциональным модулем слоя. Каналы связи между ПЭ и пропускная способность также необязательно должны быть однотипными. Каждый ПЭ имеет двухуровневую оперативную память (ОП).

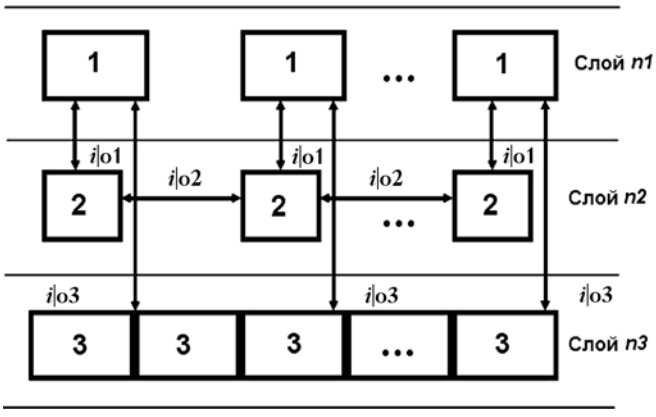


**Рис. 2. Геометрическая иллюстрация многослойной модульной схемы реализации виртуальной потоковой вычислительной системы**

Множество блоков ОП для каждого ПЭ образуют слой $n2$, т. е. блоки ОП являются слоями памяти.

**Первый уровень** модуля ОП доступен только конкретному ПЭ или модулю ПЭ. Модуль памяти этого уровня разбивается на следующие поля.

***1-е поле*** — это поле атрибутов ПЭ, в качестве которых выступают следующие характеристики: 1) логический номер ПЭ, являющийся информационным атрибутом, посредством которого идентифицируется номер деления в квантовой шкале измерения значений вычисляемой величины; 2) признак и уровень информационной активности ПЭ, посредством которых в течение каждого выделенного кванта физического времени потоковой обработки могут включаться новые ПЭ и оцениваться уровень их активности; 3) атрибуты связи с памятью более низких уровней;

***2-е поле*** — это поле памяти команд;

***3-е поле*** — поле активной базовой виртуальной памяти, которое разбито на пассивные виртуаль-

ные ячейки произвольной фиксированной длины, неизменяемой в течение вычислительного процесса, и информация в них также не изменяется. Это поле является той областью памяти (недоступной для записи), в которой хранится локальная система координат информационной привязки и поверки результатов (промежуточных и конечных) вычислительных процессов в среде вычислений ПЭ. Информационная система координат, в свою очередь, является глобальным информационным атрибутом, посредством которого задается область определения отображения состояний динамической эволюции объектов вычислительных процессов в информационном пространстве активной памяти — их домен — и область их значений — диапазон.

**Второй уровень** памяти по ее организации (логической, виртуальной и информационной) с приоритетом доступа к полям этого уровня различными ПЭ имеет следующие особенности.

Во-первых, поля этого уровня образуют динамически активную виртуальную память, элементами логической организации которой являются виртуальные ячейки двух типов: активные ячейки, размерность которых и информация в них изменяются; активно-пассивные ячейки, где изменяется их размерность, но не смысловая информация.

Во-вторых, этот уровень памяти логически разделен: на поля, доступные конкретному ПЭ, и на поля, доступные другим ПЭ виртуальной потоковой вычислительной системы.

Поля, доступные конкретному ПЭ, представляют собой подмножества виртуальных адресов для виртуальных ячеек различных типов в реальной физической памяти. Логическая структура организации этих полей в вычислительных процессах является динамически изменяемой. Механизм реализации динамической изменяемой логической структуры таких полей в адресном пространстве виртуальных ячеек различных типов в ограниченной области реального физического пространства памяти ВС достаточно прост для осуществления различных форм компьютинга и не привязан ни к типам данных, ни к системам адресации. В этом случае содержимое виртуальной ячейки интерпретируется как бинарное поле (множество), а адреса виртуальных ячеек (логическая структура адресного пространства) могут изменяться после каждой операции или их совокупности. Эти поля образуют также пространство виртуальных адресов для рабочих ячеек и хранения промежуточных результатов конкретного ПЭ.

Слой $n3$ образует пространство общей активной виртуальной памяти в адресном и информационном пространстве реальной физической памяти ВС. Этот слой логически разделен на модули памяти, которые используются для реализации следующих целей.

Во-первых, на выделенном подмножестве модулей формируется логическая структура виртуальной памяти, на которой отражается координатная решетка информационной системы координат проверки и поверки результатов вычислительного процесса, узлам решетки соответствуют пассивные виртуальные ячейки. Результаты вычислений (промежуточных и конечных) в последовательные кванты времени, на которые разбивается вычислительный процесс, хранятся в пассивных ячейках памяти координатной решетки и образуют логическую растровую структуру отражения результатов в виде символьных цепочек для значений дискретной размытой случайной функции. Логическая структура результатов на узлах решетки является информационным отображением конечной топологии изображения решения моделируемой задачи или вычислительного процесса по слоям.

Во-вторых, модули слоя $n3$ выполняют буферные функции в операциях тайлинга между модулями слоев, функции виртуальных портов и информационных шлюзов с внешним миром.

### Организация вычислительного процесса в среде вычислений виртуальной потоковой вычислительной системы

Виртуальные потоковые вычислительные системы на основе информационной модели нейросети и нейрона являются универсальными вычислительными системами нового поколения с новыми возможностями. В данном разделе подтверждается это утверждение и описывается логическая схема конфигурации архитектуры виртуальной потоковой вычислительной системы согласно геометрии и топологии нейросети под структуру информационного графа решаемой задачи.

Фундаментальным блоком виртуальной потоковой вычислительной системы является нейрон (см. рис. 1). Физическим прототипом нейрона в процессорном слое $n1$ является объединение определенного числа ПЭ (слоя 1), а в слое $n2$ — это объединение соответствующих модулей памяти (слоя 2). Число ПЭ в нейроне определяется по функциональному назначению и логической организации вычислительного процесса согласно структуре информационного графа решаемой задачи. Организация информационных входных и выходных связей для различных типов шин между модулями слоев $n1$, $n2$, $n3$ в границах информационного пространства как нейросети, так и нейрона осуществляется по входам—выходам $i|o1$, $i|o2$, $i|o3$ многослойной модульной схемы вычислительной системы. ПЭ выполняют арифметические и логические операции над символьными цепочками переменной и фиксированной длины, которые хранятся в активной виртуальной памяти модуля 2. Так как вычислительные процессы в виртуальной среде вычислительной системы реализуются и протекают в усло-

виях модельной замкнутости, ограничений, обмена и неопределенности, то модели алгоритмов и процедур в среде вычислений строят по стохастическим схемам типа Монте-Карло. Алгоритмы вычислений, построенных по таким схемам, относятся к классу алгоритмов на нечетких подмножествах, а результаты вычислений на основе таких алгоритмов являются размытыми стохастическими величинами, значения которых (символьные цепочки фиксированной длины) определены на узлах решетки информационной системы координат, прообразом которой является множество пассивных виртуальных ячеек активной виртуальной памяти.

Логическую схему построения вычислений типа итерационных процессов, точность вычислений и эффективность которых нельзя проверить прямой подстановкой, на основе размытых алгоритмов операций и процедур можно описать следующим образом. Точность информационного представления исходных данных, операндов операций и получаемых результатов в среде вычислений любой вычислительной системы в виде символьной цепочки задается числом значащих символов. Для того чтобы обеспечить требуемую точность, задаваемую числом значащих символов, на всем интервале такого итерационного вычислительного процесса необходимо, чтобы длина символьных цепочек операндов логических или арифметических операций была намного больше требуемой точности. В этом случае в операциях участвуют цепочки разной длины. Поэтому вначале выполнения основной операции, как было описано выше, реализуется операция взаимодействия с информационной средой, а далее выполняется конкретная логическая или арифметическая операция [5].

Следует особо отметить, что в среде вычислений потоковых виртуальных вычислительных систем строго определено и обозначено различие между понятиями вычисляемая величина и вычисленная величина, которое заключается в том, что вычисляемая величина является детерминированной величиной, а вычисленная величина — это размытая величина, информационный образ которой (символьная цепочка фиксированной длины) определяется на узлах решетки информационной системы координат.

После выполнения операции взаимодействия между операндами выполняется операция проектирования результата предыдущей операции в узел решетки информационной системы координат. Начальные условия обозначенного итерационного процесса определяются и задаются в информационной окрестности одного или нескольких узлов решетки информационной системы координат. Решетка информационной системы координат логически разбивается на слои возможных значений (область определения) результата вычислений, связанных между собой по времени или параметрической переменной. Интервал времени (физического или информационного) реализации вычислительного процесса кратен физическим квантам времени фиксированной длительности. В выделенный квант физического времени взаимодействуют только информационные объекты соседних слоев. Также следует отметить, что решетка информационной системы координат имеет геометрию, аналогичную нейросети (см. рис. 1). В каждый квант физического времени в среде вычислений число активных нейронов фиксировано и определено, но после завершения вычислений в конкретный фиксированный квант времени их число на следующем шаге-кванте может измениться количественно и качественно, если к вычислениям подключатся новые нейроны, Это обусловлено тем, что при информационном взаимодействии результатов вычислений с узлами решетки информационной системы координат они могут попадать в информационную зону неактивных нейронов, и тогда такой нейрон на следующем шаге-кванте активизируется.

При организации вычислительного процесса в среде вычислений вычислительной системы, имеющей архитектуру информационной модели нейросети, координатная решетка информационной системы привязки и поверки результатов вычислений логически и информационно разбивается на части и каждый нейрон функционирует в соответствующей ему области информационного пространства координатной решетки.

После завершения вычислительного процесса на узлах решетки информационной системы координат получаем размытое изображение образа результата вычисления в виде топологического комплекса на множестве активных узлов, значения которых отражаются на множестве пассивных виртуальных ячеек виртуальной активной памяти.

Далее реализуется процедура выделения из размытого изображения результата вычисления в виде детерминированного образа результата. Для реализации этой процедуры осуществляется операция топологической инкарнации размытого изображения результата, которая учитывает критерии связанности результатов вычислений: фрактальная связанность и информационная связанность. Адекватность результата определяется по степени близости оценок значений результата, полученных по указанным критериям.

То в чем виртуальные потоковые вычислительные системы, основанные на информационной модели нейросети и нейрона, являются действительно эффективными и новыми, это не эмулирование или синтез вычислительных технологий в среде вычислений традиционных компьютеров, а эмулирование и синтез технологий интеллектуальной обработки информации в системах восприятия живых систем. Например, глаз живой системы детектирует и обрабатывает информацию на основе опе-

раций взаимодействия и проектирования в узлы решетки глазного дна. Ухо обрабатывает информацию по такому же принципу.

Уникальная и устойчивая природа информационной модели нейросети и ее элементов определяет и отражает ее аналог в архитектуре соответствующей виртуальной потоковой вычислительной системы, что создает широкие возможности разработки систем искусственного и вычислительного интеллекта, широкого спектра логических и программных конструкторов, относящихся к сфере реализации решений в области искусственного интеллекта и распознавания образов, посредством того, что позволяют реализовать новые принципы осуществления компьютинга.

Дополнительным преимуществом виртуальной потоковой вычислительной системы является уникальная форма параллельной обработки информации и организации параллельных вычислительных процессов. Поскольку все нейроны нейросети (см. рис. 1) образуют локальные динамические вычислительные сети, которые в пределах кванта-шага вычислений являются автономными, даже для процессов, объединенных одним алгоритмом, позволяют использовать нейросеть с ее геометрией и топологией связей как шаблон-среду, в которой нейроны параллельно могут принимать участие при решении различных задач и реализации широкого спектра логических схем.

## Заключение

Изложенный выше подход построения архитектур виртуальных потоковых виртуальных вычислительных систем на основе информационной модели нейросети и нейрона на уровне сетевых моделей и алгоритмов позволяет реализовать архитектуры вычислительных систем нового поколения, в которых среда вычислений и информационные процессы логически и функционально организованы, исходя из следующих сформулированных принципов компьютинга: принципа взаимодействия объектов (символьные цепочки) среды вычислений с информационной виртуальной средой; принципа виртуальной перспективы в операторе проектирования; принципа связанности результатов вычислений.

Методология теории построения информационных моделей нейросети и теория динамики информационного взаимодействия в ней на логическом и математическом уровне описывает и раскрывает практическую сущность обозначенных потоковых виртуальных вычислительных систем как систем нового поколения [2, 3, 5]. Введение новых сущностей-объектов — информационной модели нейросети и нейрона, модели активной виртуальной памяти, пассивных и активных виртуальных ячеек памяти, многослойной модульной схемы реализации виртуальной потоковой вычислительной системы, процессов с локальным информационным взаимодействием, операторов взаимодействия и проектирования, информационной системы координат среды вычислений и др. значительно расширяет онтологию предметной области вычислительных систем. Каждая из этих сущностей-объектов является архитектурой или ее элементом для системы, которая может быть определена в широком смысле как виртуальная потоковая вычислительная система, основанная на информационной модели нейросети и нейрона, и является объектом настоящего и будущих исследований.

### Список литературы


1. **Барский А. Б., Шилов В. В.** Потоковая вычислительная система: программирование и оценка эффективности // Информационные технологии. 2003. № 7. Приложение. 24 с.
2. **Мышев А. В.** Информационная модель нейросети в технологиях вычислительного интеллекта и формах реализации компьютинга // Информационные технологии. 2012. № 1. С. 63—70.
3. **Мышев А. В.** Динамика информационных процессов в вычислительных технологиях компьютерного моделирования // Труды ИСА РАН. Т. 58. М.: КРАСАНД, 2010. С. 137—148.
4. **Каляев И. А., Левин И. И.** Реконфигурируемые мультиконвейерные вычислительные системы для решения потоковых задач // Информационные технологии и вычислительные системы. 2010. № 2. С. 12—22.
5. **Мышев А. В.** Метод виртуальной перспективы в моделировании размытых задач // Информационные технологии и вычислительные системы. 2011. № 3. С. 65—78.



**Адрес редакции:**

107076, Москва, Стромынский пер., 4

Телефон редакции журнала **(499) 269-5510**

E-mail: it@novtex.ru

Дизайнер *Т.Н. Погорелова*. Технический редактор *Е. В. Конова*.

Корректор *Е.В. Комиссарова*.

Сдано в набор 06.03.2014. Подписано в печать 21.04.2014. Формат 60×88 1/8. Бумага офсетная.

Усл. печ. л. 8,86. Заказ IT514. Цена договорная.

Журнал зарегистрирован в Министерстве Российской Федерации по делам печати, телерадиовещания и средств массовых коммуникаций.

Свидетельство о регистрации ПИ № 77-15565 от 02 июня 2003 г.

Оригинал-макет ООО "Адвансед солюшнз". Отпечатано в ООО "Адвансед солюшнз".

119071, г. Москва, Ленинский пр-т, д. 19, стр. 1.