

# ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

10(218)  
2014

ТЕОРЕТИЧЕСКИЙ И ПРИКЛАДНОЙ НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ

Издается с ноября 1995 г.

УЧРЕДИТЕЛЬ  
Издательство "Новые технологии"

## СОДЕРЖАНИЕ

### ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

- Зак Ю. А.** Модифицированные генетические алгоритмы решения многоэкстремальных задач в условиях ограничений . . . . . 3  
**Воронов Р. В., Мошевикин А. П.** Применение условной энтропии при формировании рекомендаций по размещению базовых станций в локальных системах позиционирования . . . . . 11

### ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ И СЕТИ

- Соловьев А. Н., Стемпковский А. Л.** Методы повышения отказоустойчивости работы устройства управления микросистемы за счет введения структурной избыточности . . . . . 17  
**Амербаев В. М., Соловьев Р. А., Тельпухов Д. В.** Метод вычисления циклической свертки на базе БПФ с использованием чисел Прота . . . . . 22

### МОДЕЛИРОВАНИЕ И ОПТИМИЗАЦИЯ

- Перегуда А. И.** Математическая модель надежности информационной системы, состоящей из объекта защиты и  $N$  систем безопасности . . . . . 28

### СИСТЕМЫ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ

- Гурарий М. М., Жаров М. М., Русаков С. Г., Ульянов С. Л.** Методы моделирования в частотной области периодических режимов автогенераторных схем . . . . . 36

### ОБРАБОТКА ИЗОБРАЖЕНИЙ

- Маркелов К. С., Нечаев В. В.** Алгоритм повышения качества цифровых изображений дактилоскопической информации . . . . . 44

### ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В БИМЕДИЦИНСКИХ СИСТЕМАХ

- Садыков С. С., Буланова Ю. А., Канунова Е. Е., Захарова Е. А.** Разработка концепций построения информационной системы диагностики новообразований на маммограммах . . . . . 51  
**Абдуллаев Н. Т., Дышин О. А., Абдуллаева Г. Э.** Интервальные байесовские механизмы вывода при дифференциальной диагностике нервно-мышечных заболеваний . . . . . 56

### ПРИКЛАДНЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

- Марченко А. М., Попов Е. А., Савченко В. Ю.** Трассировка соединений в нанометровых стандартных ячейках с помощью шаблонов . . . . . 65  
**Михайлов В. Ю., Гридин В. Н., Мазепа Р. Б.** Безопасное информационное взаимодействие. Проблемы и решения . . . . . 72  
**Добрынин А. С., Койнов Р. С.** Об одном алгоритме построения расписаний в условиях временных ограничений . . . . . 77

Главный редактор:  
СТЕМПКОВСКИЙ А. Л.,  
акад. РАН, д. т. н., проф.

Зам. главного редактора:  
ИВАННИКОВ А. Д., д. т. н., проф.  
ФИЛИМОНОВ Н. Б., д. т. н., с.н.с.

Редакционный совет:  
БЫЧКОВ И. В., акад. РАН, д. т. н.  
ЖУРАВЛЕВ Ю. И.,  
акад. РАН, д. ф.-м. н., проф.  
КУЛЕШОВ А. П.,  
акад. РАН, д. т. н., проф.  
ПОПКОВ Ю. С.,  
чл.-корр. РАН, д. т. н., проф.  
РУСАКОВ С. Г.,  
чл.-корр. РАН, д. т. н., проф.  
СОЙФЕР В. А.,  
чл.-корр. РАН, д. т. н., проф.  
СОКОЛОВ И. А., акад.  
РАН, д. т. н., проф.  
СУЕТИН Н. В., д. ф.-м. н., проф.  
ЧАПЛЫГИН Ю. А.,  
чл.-корр. РАН, д. т. н., проф.  
ШАХНОВ В. А.,  
чл.-корр. РАН, д. т. н., проф.  
ШОКИН Ю. И.,  
акад. РАН, д. т. н., проф.  
ЮСУПОВ Р. М.,  
чл.-корр. РАН, д. т. н., проф.

Редакционная коллегия:  
АВДОШИН С. М., к. т. н., доц.  
АНТОНОВ Б. И.  
БАРСКИЙ А. Б., д. т. н., проф.  
ВАСЕНИН В. А., д. ф.-м. н., проф.  
ГАЛУШКИН А. И., д. т. н., проф.  
ДИМИТРИЕНКО Ю. И., д. ф.-м. н., проф.  
ДОМРАЧЕВ В. Г., д. т. н., проф.  
ЗАГИДУЛЛИН Р. Ш., к. т. н., доц.  
ЗАРУБИН В. С., д. т. н., проф.  
ИСАЕНКО Р. О., к. т. н., с.н.с.  
КАРПЕНКО А. П., д. ф.-м. н., проф.  
КОЛИН К. К., д. т. н., проф.  
КУЛАГИН В. П., д. т. н., проф.  
КУРЕЙЧИК В. М., д. т. н., проф.  
КУХАРЕНКО Б. Г., к. ф.-м. н., доц.  
ЛЬВОВИЧ Я. Е., д. т. н., проф.  
МИХАЙЛОВ Б. М., д. т. н., проф.  
НЕЧАЕВ В. В., к. т. н., проф.  
РЯБОВ Г. Г., чл.-корр. РАН, д. т. н., проф.  
СОКОЛОВ Б. В., д. т. н., проф.  
УСКОВ В. Л., к. т. н. (США)  
ФОМИЧЕВ В. А., д. т. н., проф.  
ЧЕРМОШЕНЦЕВ С. Ф., д. т. н., проф.  
ШИЛОВ В. В., к. т. н., доц.

Редакция:  
БЕЗМЕНОВА М. Ю.  
ГРИГОРИН-РЯБОВА Е. В.  
ЛЫСЕНКО А. В.  
ЧУГУНОВА А. В.

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

**CONTENTS**

**INTELLIGENT SYSTEMS AND TECHNOLOGIES**

**Zack Yu. A.** Modified Genetic Algorithm for Solving Multiextremal Problems within the Constraints . . . . . 3  
**Voronov R. V., Moschevikin A. P.** Use of Conditional Entropy for Optimal Disposition of Base Stations in Local Positioning Systems . . . . . 11

**COMPUTING SYSTEMS AND NETWORKS**

**Soloviev A. N., Stempkovskij A. L.** Methods to Improve Noise Tolerance of the Control Equipment Microsystems by Structural Redundancy Introduction . . . . . 17  
**Amerbaev V. M., Solovyev R. A., Telpukhov D. V.** Cyclic Convolution Calculation Method Based on FFT Using Proth Numbers . . . . . 22

**MODELING AND OPTIMIZATION**

**Pereguda A. I.** Mathematical Model of Reliability of Information System, Consisting of Object Protection and N Security Systems . . . . . 28

**CAD-SYSTEMS**

**Gourary M. M., Zharov M. M., Rusakov S. G., Ulyanov S. L.** Frequency Domain Simulation Methods of Periodic States of Oscillators . . . . . 36

**IMAGE PROCESSING**

**Markelov K. S., Nechaev V. V.** Algorithm to Improve the Quality of Digital Images of Fingerprint Information . . . . . 44

**INFORMATION TECHNOLOGIES IN BIOMEDICAL SYSTEMS**

**Sadykov S. S., Bulanova Y. A., Kanunova E. E., Zaharova E.A.** Development of Concepts for Building Information Systems Diagnosis of Tumors on Mammograms . . . 51  
**Abdullaev N. T., Dyshin O. A., Abdullaeva G. E.** Interval Bayesian Inference in the Differential Diagnosis of Neuromuscular Diseases . . . . . 56

**APPLIED INFORMATION TECHNOLOGIES**

**Marchenko A. M., Popov E. A., Savchenko V. Yu.** Pattern Based Standard Cell Routing . . . . . 65  
**Dobrynin A. S., Koynov R. S.** About an Algorithm for Construction of a Time-Schedule in Conditions of Timig Constraints . . . . . 72  
**Mikhaylov V. Yu., Gridin V. N., Mazepa R. B.** Secure Information Interaction. Problems and Solutions . . . . . 77

**Editor-in-Chief:**

Stempkovsky A. L., Member of RAS,  
Dr. Sci. (Tech.), Prof.

**Deputy Editor-in-Chief:**

Ivannikov A. D., Dr. Sci. (Tech.), Prof.  
Filimonov N. B., Dr. Sci. (Tech.), Prof.

**Chairman:**

Bychkov I. V., Member of RAS,  
Dr. Sci. (Tech.), Prof.  
Zhuravljov Yu. I., Member of RAS,  
Dr. Sci. (Phys.-Math.), Prof.  
Kuleshov A. P., Member of RAS,  
Dr. Sci. (Tech.), Prof.  
Popkov Yu. S., Corresp. Member of RAS,  
Dr. Sci. (Tech.), Prof.  
Rusakov S. G., Corresp. Member of RAS,  
Dr. Sci. (Tech.), Prof.  
Soifer V. A., Corresp. Member of RAS,  
Dr. Sci. (Tech.), Prof.  
Sokolov I. A., Member of RAS,  
Dr. Sci. (Phys.-Math.), Prof.  
Suetin N. V.,  
Dr. Sci. (Phys.-Math.), Prof.  
Chaplygin Yu. A., Corresp. Member of RAS,  
Dr. Sci. (Tech.), Prof.  
Shakhnov V. A., Corresp. Member of RAS,  
Dr. Sci. (Tech.), Prof.  
Shokin Yu. I., Member of RAS,  
Dr. Sci. (Tech.), Prof.  
Yusupov R. M., Corresp. Member of RAS,  
Dr. Sci. (Tech.), Prof.

**Editorial Board Members:**

Avdoshin S. M., Cand. Sci. (Tech.), Ass. Prof.  
Antonov B. I.  
Barsky A. B., Dr. Sci. (Tech.), Prof.  
Vasenin V. A., Dr. Sci. (Phys.-Math.), Prof.  
Galushkin A. I., Dr. Sci. (Tech.), Prof.  
Dimitrienko Yu. I., Dr. Sci. (Phys.-Math.), Prof.  
Domrachev V. G., Dr. Sci. (Tech.), Prof.  
Zagidullin R. Sh., Cand. Sci. (Tech.), Ass. Prof.  
Zarubin V. S., Dr. Sci. (Tech.), Prof.  
Isaenko R. O., Cand. Sci. (Tech.)  
Karpenko A. P., Dr. Sci. (Phys.-Math.), Prof.  
Kolin K. K., Dr. Sci. (Tech.)  
Kulagin V. P., Dr. Sci. (Tech.), Prof.  
Kurechik V. M., Dr. Sci. (Tech.), Prof.  
Kukhareenko B. G., Cand. Sci. (Phys.-Math.)  
Ljvovich Ya. E., Dr. Sci. (Tech.), Prof.  
Mikhailov B. M., Dr. Sci. (Tech.), Prof.  
Nechaev V. V., Cand. Sci. (Tech.), Ass. Prof.  
Ryabov G. G., Corresp. Member of RAS,  
Dr. Sci. (Tech.), Prof.  
Sokolov B. V., Dr. Sci. (Tech.)  
Uskov V. L. (USA), Dr. Sci. (Tech.)  
Fomichev V. A., Dr. Sci. (Tech.), Prof.  
Chermoshentsev S. F., Dr. Sci. (Tech.), Prof.  
Shilov V. V., Cand. Sci. (Tech.), Ass. Prof.

**Editors:**

Bezmenova M. Yu.  
Grigorin-Ryabova E. V.  
Lysenko A. V.  
Chugunova A. V.

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

УДК 004.421

Ю. А. Зак, д-р техн. наук, научный эксперт и консультант,  
г. Аахен, Германия, e-mail: yuriy\_zack@hotmail.com

## Модифицированные генетические алгоритмы решения многоэкстремальных задач в условиях ограничений

*Предложены модификации генетических алгоритмов для решения экстремальных задач в условиях ограничений, для которых по простым формулам могут быть вычислены оценки оптимального решения и граничные значения левых частей функций ограничений задачи. Допустимая область поиска решений разбивается на несколько подобластей существенно меньшей размерности, в каждой из которых реализуется эволюционная стратегия. Использование оценок оптимального решения, исключение областей, не содержащих допустимых и оптимальных решений, использование операторов проектирования на область ограничений задачи позволяют повысить эффективность работы генетических алгоритмов для рассматриваемых классов функций.*

**Ключевые слова:** генетические алгоритмы, сепарабельно квазимонотонные функции, области, не содержащие допустимых и оптимальных планов, оценки оптимального решения

### Введение

Генетический алгоритм (англ. *genetic algorithm*) (GA) — это эвристический алгоритм решения задач оптимизации, представляющий собой последовательный подбор, комбинирование и вариацию искоемых параметров на основе механизмов, напоминающих биологическую эволюцию. Это компьютерная программа, реализующая математическую модель эволюции в природе. В GA используются как аналог механизма генетического наследования, так и аналог естественного отбора. Генетический алгоритм включает три операции: воспроизводство, скрещивание и мутация. При описании алгоритма зачастую сохраняется биологическая терминология, и задача кодируется таким образом, чтобы ее решение могло быть представлено в виде некоторого числового или битового вектора ("хромосома"). Поскольку для работы алгоритма в качестве информации об оптимизируемой функции используются лишь ее значения в рассматриваемых точках пространства поиска, и не требуется вычислений ни производных, ни каких-либо иных характеристик, то данный алгоритм применим к широкому классу функций, в частности, не имеющих аналитического описания. Вследствие этого GA эффективно применяется для решения задач как дискретной, так и непрерывной оптимизации. GA является очень гибким, и будучи построенным в предположении, что известен лишь минимум информации (как это часто бывает в задачах оптимизации сложных техниче-

ских и экономических систем), этот алгоритм является эффективным особенно в тех задачах, где не существует общеизвестных алгоритмов решения или высока степень априорной неопределенности.

При решении оптимизационных задач генетическими алгоритмами в ряде случаев гены представляют вещественными числами (*real-coded*) [1, 4–6], хромосоме — в виде вектора вещественных чисел. Длина хромосомы совпадает с длиной вектора, являющегося решением оптимизационной задачи, и каждый ген отвечает за одну переменную.

В процессе работы GA [1, 4–6] случайным образом создается некоторое количество начальных векторов ("начальная популяция"). Они оцениваются с использованием функции эффективности (*Fitness*)  $F(X)$ . Для каждого вектора рассчитывается значение  $F(X)$ , которое определяет вероятность выживания организма, представленного данным вектором. На основании значений  $F(X)$  выбираются векторы (*селекция*), допущенные к скрещиванию, к которым применяются операторы GA: *скрещивание*, т. е. *кроссовер*, и мутация (*mutation*), создавая таким образом следующее поколение.

В работе генетических алгоритмов могут быть выделены две стадии:

- 1) подготовительная;
- 2) основная циклическая стадия, предусматривающая выбор индивидов из текущей популяции (отбор или селекция), скрещивание и/или мутацию, вычисление функций эффективности для всех осо-

бей, формирование нового поколения. Если выполняются условия останова, то происходит конец циклического процесса и останов алгоритма, в противном случае осуществляется переход к началу нового цикла.

Генетическим алгоритмам присущи следующие недостатки, характерные для большинства эвристических методов:

- не определен класс задач, для которых применение этого метода является эффективным;
- трудности реализации и существенное увеличение затрат на поиск в условиях наличия ограничений в виде систем неравенств и равенств;
- очень большие объемы вычислений и крайне медленная скорость сходимости;
- отсутствие оценки точности полученного приближенного решения, а также четких рекомендаций о выборе параметров алгоритмов.

В данной работе выделяется достаточно широкий класс задач оптимизации в условиях ограничений, для которых можно существенно повысить эффективность применения генетических алгоритмов, существенно сократив затраты на поиск и определив оценку точности полученных решений на каждой стадии работы алгоритма.

В предлагаемых в данной работе, как и в работах автора [2, 3], ГА вся допустимая область поиска оптимальных решений разбивается на несколько подобластей существенно меньшей размерности —  $H^r$ ,  $r = 1, \dots, R$ . На основе свойств допустимых решений исключаются из рассмотрения области, не содержащие допустимых решений, что позволяет существенно сократить область поиска. Для каждой из этих подобластей вычисляются оценки нижней границы оптимального решения —  $\xi[F^r(X)]$ . ГА реализуется в каждой из выделенных подобластей, что позволяет осуществлять процесс вычислений на многопроцессорных ЭВМ. Если в процессе вычислений в какой-либо из подобластей  $H^r$  получен вектор  $X^r$ , для которого  $F(X^r|X^r \in G)$ , для которого  $F(X^r|X^r \in G) \leq \xi[F(X^p)]$ , то подобласти  $H^p$  исключаются из дальнейшего рассмотрения и реализация ГА в этой подобласти  $H^p$  в дальнейшем не реализуется. Если в процессе вычислений в некоторой подобласти поиска  $H^r$  построен некоторый вектор  $\bar{X}^r$ , удовлетворяющий соотношениям

$$F(\bar{X}^r|\bar{X}^r \in G) - \min_{1 \leq r \leq R} \xi[F(X^r|X^r \in H^r)] \leq \varepsilon,$$

где  $\varepsilon$  — требуемая точность получения решения, то ГА завершает свою работу.

Кроме того, во 2-й части работы для широкого круга задач оптимизации рассматриваются операторы проектирования генерируемых векторов решений  $X \in Q$ , не принадлежащих области ограничений  $G$ , на границу области  $G \cap Q$ .

## 1. Некоторые свойства сепарабельно квазимонотонных функций

При реализации ГА в пределах  $n$ -мерного параллелепипеда  $Q \subseteq H$  и в случае сложной конфигурации области ограничений  $G$  возникают серьезные вычислительные трудности, связанные с генерацией векторов, принадлежащих заданной области ограничений —  $X \in (G \cap H)$ .

Рассматриваются модификации ГА оптимизации для решения следующей многоэкстремальной задачи невыпуклого математического программирования:

$$f_0^*(X) = \min_{X \in (H \cap G)} f_0(X); \quad (1)$$

$$G = \{X | f_i(X) \leq b_i, i = 1, \dots, m\}; \quad (2)$$

$$H = \{X = (x_1, \dots, x_n) | h_j^1 \leq x_j \leq h_j^2, j = 1, \dots, n\}. \quad (3)$$

Область  $G$  может иметь достаточно сложную конфигурацию, и оптимизационная задача может оказаться многоэкстремальной или задачей математического программирования в несвязанных областях. Это обстоятельство значительно увеличивает затраты на поиск и в большинстве случаев существенно снижает его эффективность.

Функции  $f_i(X)$ ,  $i = 0, 1, \dots, m$ , могут быть невыпуклыми, недифференцируемыми и даже разрывными с непрерывными и целочисленными переменными. Существенным является то, что они обладают следующими свойствами: для любых фиксированных значений  $\bar{y}_k \in H_k$ ,  $k = 1, \dots, n$ ,  $k \neq j$ , всех остальных переменных и любых двух значений переменной  $x_j$ , равных  $x_j = g_j$  и  $x_j = l_j$ , где  $h_j^1 \leq g_j < l_j \leq h_j^2$ , выполняется одно из пары условий:

$$f(\bar{y}_1, \dots, \bar{y}_{j-1}, g_j, \bar{y}_{j+1}, \dots, \bar{y}_n) \leq f(\bar{y}_1, \dots, \bar{y}_{j-1}, l_j, \bar{y}_{j+1}, \dots, \bar{y}_n); \quad (4)$$

$$f(\bar{y}_1, \dots, \bar{y}_{j-1}, g_j, \bar{y}_{j+1}, \dots, \bar{y}_n) \geq f(\bar{y}_1, \dots, \bar{y}_{j-1}, l_j, \bar{y}_{j+1}, \dots, \bar{y}_n). \quad (5)$$

Если  $QMF(j)$  — неубывающая функция по переменной  $x_j$ , то справедливо выражение (4), а если  $QMF(j)$  — невозрастающая функция, то справедливо выражение (5).

Следовательно, экстремальное значение  $f_i(X)$  по каждой переменной (при фиксированном значении всех остальных переменных вектора  $X$ ) всегда находится на границе  $Q$ . В дальнейшем, как и в работе [2], будем называть такие функции сепарабельно квазимонотонными по переменной  $x_j$  и обозначать  $QMF(j)$ . Если  $QMF(j)$  — неубывающая функция по переменной  $x_j$ , то справедливо выражение (4), а если  $QMF(j)$  — невозрастающая функция, то справедливо выражение (5).

В работах автора [2] установлены свойства допустимых решений задачи и определены оценки нижней границы оптимальных решений, как для всей системы ограничений задачи, так и для отдельных подобластей допустимых планов. Показано [2], что класс таких функций достаточно широк и приведены многочисленные примеры таких функций.

Рассмотрим поведение и свойства таких функций в системе  $n$ -мерных параллелепипедов:

$$H^r = H_1^r \times H_2^r \times \dots \times H_j^r \times \dots \times H_n^r,$$

$$\text{где } H_j^r = \{x_j^r | h_j^{1r} \leq x_j^r \leq h_j^{2r}\}, r = 1, \dots, R. \quad (6)$$

Из условий (2) и (3) следует, что для любых значений  $\bar{y}_k \in H^r, k = 1, \dots, n, k \neq j$ ,

$$\begin{aligned} \min_{x_j \in H_j^r} f(\bar{y}_1, \dots, \bar{y}_{j-1}, x_j, \bar{y}_{j+1}, \dots, \bar{y}_n) = \\ = f(\bar{y}_1, \dots, \bar{y}_{j-1}, x_j^{r,-}, \bar{y}_{j+1}, \dots, \bar{y}_n); \end{aligned} \quad (7)$$

$$\begin{aligned} \max_{x_j \in H_j^r} f(\bar{y}_1, \dots, \bar{y}_{j-1}, x_j, \bar{y}_{j+1}, \dots, \bar{y}_n) = \\ = f(\bar{y}_1, \dots, \bar{y}_{j-1}, x_j^{r,+}, \bar{y}_{j+1}, \dots, \bar{y}_n), \end{aligned} \quad (8)$$

где

$$x_j^{r,-} = \begin{cases} h_{js}^{1r} \\ h_j^{2r} \end{cases} \text{ и } x_j^{r,+} = \begin{cases} h_{js}^{1r} \\ h_j^{2r} \end{cases}. \quad (9)$$

**Утверждение 1.** Аргумент минимума и максимума сепарабельно квазимоноотонных функций, обладающих свойствами (4), (5), (7)–(9) на  $n$ -мерном параллелепипеде (6), достигается при значении переменных, принадлежащих множеству  $\Lambda_s$ , т. е.

$$\begin{aligned} \arg \min_{(x_1, x_2, \dots, x_n) \in H_s} f(x_1, x_2, \dots, x_j, \dots, x_n) \cap \hat{\Lambda}_s = \emptyset, \\ \arg \max_{(x_1, x_2, \dots, x_n) \in H_s} f(x_1, x_2, \dots, x_j, \dots, x_n) \cap \hat{\Lambda}_s = \emptyset. \end{aligned} \quad (10)$$

Доказательство утверждения 1 приведено в работе автора [2].

### Следствие утверждения 1.

Экстремальные значения сепарабельно квазимоноотонных функций следует искать среди вершин  $n$ -мерного параллелепипеда (6), что сводится к перебору и вычислению значений функции в  $2^n$  точках множества  $\Lambda_s$ .

Однако для задач большой размерности такой объем вычислений может оказаться значительным. Отмеченные свойства  $QMF(j)$  функций позволяют вычислить значение нижней границы оптимального решения задачи  $\xi_0 \{f_0^*(X) | X \in (H \cap G)\}$ .

Разобьем все пространства поиска  $H$  на  $R$  подобластей (не пересекающихся  $n$ -мерных параллеле-

пипедов меньших размеров  $H^r, r = 1, \dots, R$ ), удовлетворяющих условиям  $H^r \cap H^p = \emptyset, r, p = 1, \dots, R; \bigcup_{r=1}^R H^r = H$ . Рассмотрим поведение и свойства таких функций в системе  $n$ -мерных параллелепипедов.

Аналогично результатам, приведенным в утверждении 1, могут быть вычислены нижние границы оптимального решения для каждого из параллелепипедов  $H^r - \xi_0^r \{f_0^*(X) | X \in (H^r \cap G)\}, r = 1, \dots, R$ . Тогда значение нижней границы оптимального решения задачи вычисляется по формуле

$$\begin{aligned} \xi_1 \{f_0^*(X) | X \in (H \cap G)\} = \\ = \max_{1 \leq r \leq R} \xi_0^r \{f_0^*(X) | X \in (H^r \cap G)\}. \end{aligned}$$

### Утверждение 2.

$$\xi_1 \{f_0^*(X) | X \in (H \cap G)\} \geq \xi_0 \{f_0^*(X) | X \in (H \cap G)\},$$

т. е. значение  $\xi_1 \{f_0^*(X) | X \in (H \cap G)\} = \max_{1 \leq r \leq R} \xi_0^r \times \{f_0^*(X) | X \in (H^r \cap G)\}$  является более точной оценкой оптимального решения.

Особый интерес представляют функции  $f(x_1, \dots, x_j, \dots, x_n)$ , сепарабельные свойства которых по всем переменным не зависят от значений других переменных. Обозначим эти функции  $SQMF$ . Минимум и максимум таких функций по каждой из переменных  $x_j$  не зависит от значения остальных переменных  $x_k \in H_{ks}, k = 1, \dots, n, k \neq j$  и всегда достигается при одном и том же граничном значении области определения этой переменной, т. е. при значении  $h_{js}^1$  или  $h_{js}^2$ .

Примерами таких функций являются линейные функции, некоторый класс степенных и логарифмических функций произведения переменных при условии, что каждая из переменных не может принимать отрицательные значения, и многие другие функции, приведенные в работе автора [2].

В работе автора [2] доказан ряд утверждений и теорем, обосновывающих итеративный процесс выделения областей, не содержащих допустимых планов для систем нелинейных неравенств с двухсторонними ограничениями, функции левых частей которых  $f_i(X), i = 1, \dots, m$ , — суть функции класса  $SQMF$ :

$$\begin{aligned} y_{jis}^{\min} = \arg \min_{x_j \in H_{js}} \{f_i(x_1, \dots, x_{j-1}, x_j, x_{j+1}, \dots, x_n) | x_k \in H_{ks}, \\ k = 1, \dots, n, k \neq j\}, j = 1, \dots, n. \end{aligned} \quad (11)$$

Для описания этого алгоритма обозначим:

$J_i^+$  и  $J_i^-$  — соответственно подмножества индексов  $j \in J$ , для которых минимальное значение функции по данной переменной, т. е. значение  $y_{jis}^{\min}$ ,

достигается при значениях  $y_{jis}^{\min} = h_{js}^1$ , т. е. по данной переменной функция является неубывающей, и  $y_{jis}^{\min} = h_{js}^2$ , т. е. по данной переменной функция является невозрастающей;

$I_j^+$  и  $I_j^-$  — соответственно подмножества индексов функций  $f_i(X)$ ,  $i = 1, \dots, m$ , являющихся неубывающими и невозрастающими функциями по переменной  $x_j$ ,  $j = 1, \dots, n$ .

Ниже приводится алгоритм выделения в  $n$ -м параллелепипеде  $H_s$  областей, не содержащих допустимых планов.

### Алгоритм 1.

Введем счетчик числа шагов  $M$ , в течение которых  $H_{s+1} = H_s$ , т. е. не происходит изменения  $n$ -мерного параллелепипеда ни по одной из переменных. В начале процесса положим  $s = 0$ ,  $M = 0$ ,  $H_{j,0} = H_j$ ,  $j = 1, \dots, n$ .

На каждом  $s$ -м шаге итеративного процесса в цикле для каждой переменной  $j = 1, \dots, n$  выполняем вычисления (11) при подстановке значений

остальных переменных  $x_k = y_{kis}^{\min}$ ,  $k = 1, \dots, n$ ,  $k \neq j$ . Если выполняется хотя бы одно из неравенств вида

$$f_i(y_{1is}^{\min}, \dots, y_{jis}^{\min}, \dots, y_{nis}^{\min}) > b_i, \quad i = 1, \dots, m, \quad (12)$$

то система ограничений задачи в пределах  $X \in H$  является несовместной, и алгоритм завершает работу, т. е.  $H_s = \emptyset$ .

Определим значения:

$$\begin{aligned} \bar{t}_{jis}^- &= \\ &= \arg \max_{x_j \in H_{js}} f_i(y_{1,i,s}^{\min}, \dots, y_{j-1,i,s}^{\min}, x_j, y_{j+1,i,s}^{\min}, \dots, \\ &\quad \dots, y_{n,i,s}^{\min}), \quad i \in J_i^+; \quad \bar{t}_{js}^- = \max_{i \in I_j^+} \bar{t}_{jis}^+; \end{aligned}$$

$$\begin{aligned} \bar{t}_{jis}^+ &= \\ &= \arg \min_{x_j \in H_{js}} f_i(y_{1,i,s}^{\min}, \dots, y_{j-1,i,s}^{\min}, x_j, y_{j+1,i,s}^{\min}, \dots, y_{n,i,s}^{\min}), \\ &\quad i \in J_i^-; \quad \bar{t}_{js}^+ = \min_{i \in I_j^-} \bar{t}_{jis}^-; \end{aligned}$$

$$x_{js} \in [\bar{t}_{js}^-, \bar{t}_{js}^+], \quad j = 1, \dots, n.$$

Если для какого-то индекса  $j = 1, \dots, n$  выполняется неравенство  $\bar{t}_{js}^- > \bar{t}_{js}^+$ , то система ограничений для этого параллелепипеда является несовместной и алгоритм завершает свою работу.

Если на некотором  $s_1$ -м шаге справедливо  $H_{s_1+1} = H_{s_1}$  (т. е.  $\bar{H}_{js_1} = \emptyset$ ), то значение  $M$  увеличиваем на 1. Если  $M = n$ , то полученное подмно-

жество  $H_{s_1+1}$  не может быть подвергнуто дальнейшему изменению (сужению), и итеративный процесс останавливается, т.е. дальнейшее исключение областей, не содержащих допустимых планов, на основе описанного алгоритма невозможно.

Если  $H_{s_1+1} \subset H_{s_1}$  и при этом  $M > 0$ , то полагаем  $M = 0$  и переходим к следующему шагу процесса.

Ясно, что через конечное число шагов процесса мы либо придем к факту о несовместности исходной системы ограничений, либо получим  $n$ -мерный параллелепипед  $\tilde{H} = H_{s_1+1} \neq \emptyset$ , который не может быть в дальнейшем подвергнут изменениям.

Отметим, что успешное завершение этого итеративного процесса и получение в результате этого  $n$ -мерного параллелепипеда  $\tilde{H} = H_{s_1+1} \neq \emptyset$  не гарантирует, что система ограничений исходной задачи совместна, а является только необходимым условием существования допустимых решений. Обозначим этот параллелепипед меньших размеров  $\tilde{H} = Q^r = H^r \cap G \neq \emptyset$ .

## 2. Модифицированные генетические алгоритмы оптимизации в условиях системы ограничений

В работе предлагаются два подхода реализации модифицированных ГА.

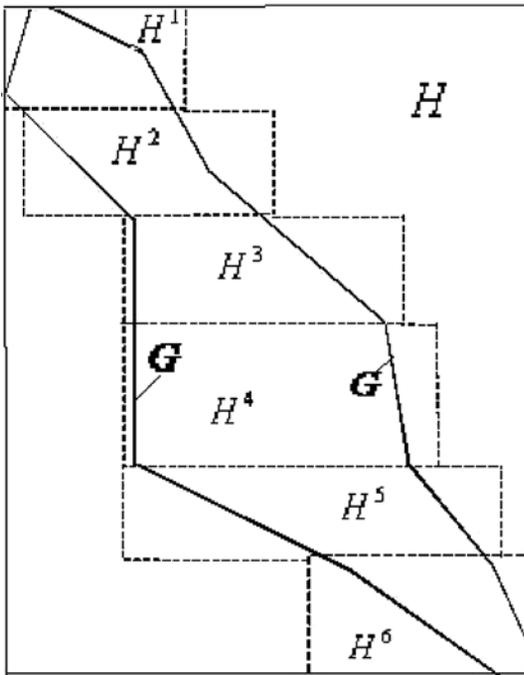
**Первый подход** связан с разбиением всего пространства поиска  $H$  на  $R$  подобластей (не пересекающихся  $n$ -мерных параллелепипедов меньших размеров  $H^r$ ,  $r = 1, \dots, R$ ), удовлетворяющих условиям

$$H^r \cap H^p = \emptyset, \quad r, p = 1, \dots, R; \quad \bigcup_{r=1}^R H^r = H, \quad (13)$$

исключением из рассмотрения областей, не содержащих допустимых и оптимальных решений, и реализацией ГА независимо в каждом из этих перспективных  $n$ -мерных параллелепипедов. Иллюстрация этого подхода представлена на рисунке, где область ограничений задачи  $G$  занимает лишь незначительную часть площади определения границ переменных задачи  $H$ .

**Второй подход** предусматривает проектирование каждого из векторов, полученных в процессе реализации стандартных процедур ГА и не принадлежащих области ограничений задачи, на границу области ограничений задачи, что позволит сохранить для рассмотрения каждую сгенерированную в процессе работы алгоритма особь и существенно сократить затраты на поиск.

На основе свойств допустимых решений алгоритмами 1 выполняется исключение в каждом из множеств  $H^r$  областей, не содержащих допустимых решений (сокращение размеров и исключение из рассмотрения некоторых из этих областей), а также вычисление для каждого из перспективных под-



Разбиение пространства поиска  $H$  на шесть прямоугольников меньших размеров  $H^r$

множеств нижней границы оптимальных решений —  $\xi\{f_0^*(X|H^r)\}$ . Ясно, что

$$\xi\{f_0^*(X)\} = \min_{1 \leq r \leq R} \xi\{f_0^*(X|H^r)\}. \quad (14)$$

Реализация генетического алгоритма проводится параллельно в каждой из областей  $H^r$  независимо друг от друга.

На начальном этапе в каждой из подобластей  $H^{r,0}$  случайным образом создается некоторая начальная популяция, состоящая из  $N^{r,0}$  особей, т. е. числовых векторов  $X_l^{r,0} \in (H^r \cap G)$ ,  $l = 1, 2, \dots, N^{r,0}$ .

На каждом  $k$ -м шаге процесса,  $k = 0, 1, \dots, K$ , в каждой из подобластей  $H^r$ ,  $r = 1, \dots, R$ , рассматривается популяция, состоящая из  $N^{r,k}$  особей. Упорядочим все числовые векторы  $X_l^{r,0}$  в порядке убывания значений  $f(X_l^{r,0})$ :

$$\begin{aligned} \tilde{U}^{r,k} = \{ & X_1^{r,k} > X_2^{r,k} > \dots > X_l^{r,k} > X_{l+1}^{r,k} > \dots > \\ & X_q^{r,k} > \dots > X_N^{r,k} | f_0(X_l^{r,k}) \leq f_0(X_{l+1}^{r,k}) \}. \end{aligned} \quad (15)$$

По итогам отбора из  $N^{r,k}$  особей этой популяции выбирается только  $n^{r,k} < N^{r,k}$  особей, для которых значение критерия оптимальности  $f(X_l^{r,k}) \leq f(X_q^{r,k})$ , войдут в итоговую популяцию (или на их основе будет создана новая популяция). Остальные особи погибают.

Эти наиболее приспособленные особи получают возможность воспроизводить новое потомство с помощью перекрестного скрещивания и мутации с другими особями популяции. Описанные в генетических алгоритмах операции размножения (или кроссовер) и мутации в различных алгоритмах определяются по-разному [1, 4–6]. В процессе мутации методами локального случайного поиска в заданных алгоритмом определенных узких пределах изменяются некоторые компоненты векторов  $X_l^{r,k}$ . В процессе кроссовера выполняется некоторым разумным способом смешивание свойств лучших родителей, т. е. построение на основе смешивания компонент некоторых двух векторов  $X_{l_1}^{r,k}$  и  $X_{l_2}^{r,k}$  и построение двух новых векторов  $X_{l_1}^{r,(k+1)}$  и  $X_{l_2}^{r,(k+1)}$ .

Отметим, что векторы оптимизируемых переменных, полученные в процессе операций селекции или скрещивания и не принадлежащие области ограничений задачи  $Y_l^{r,(k+1)} \notin G$ , необходимо спроектировать на область ограничений (2), (3), т. е. найти релевантные значения  $X^r \in \{G \cap H^r\}$ . Такое проектирование может быть осуществлено, например, на основании следующего алгоритма. Пусть  $X_l^{r,k} \in \{G \cap H^r\}$  — вектор оптимизируемых переменных, преобразованием мутации которого получен вектор  $Y_p^{r,(k+1)} \notin G$ . Соединим векторы  $Y_p^{r,(k+1)} \notin G$  и  $X_l^{r,k} \in \{G \cap H^r\}$  отрезком прямой  $L$ . В пределах данного отрезка ищется точка  $X_l^{r,(k+1)} \in \{G \cap H^r\}$ , которая находится достаточно близко к границе области  $G$ , что может быть получено, например, алгоритмом деления отрезка пополам. В случае операций кроссовера полученный в процессе скрещивания двух векторов  $X_{l_1}^{r,k} \in (G \cap H^r)$  и  $X_{l_2}^{r,k} \in (G \cap H^r)$  новый вектор  $Y_l^{r,(k+1)} \in H^r$ , но  $Y_l^{r,(k+1)} \notin G$ , соединяется отрезками прямых  $L_1$  и  $L_2$  соответственно с векторами  $X_{l_1}^{r,k}$  и  $X_{l_2}^{r,k}$ . Как и в предыдущем случае, в пределах данных отрезков находятся две точки  $\bar{Y}_\zeta^{r,(k+1)} \in \{G \cap H^r\}$ ,  $\zeta = 1, 2$ , которые лежат достаточно близко к границе области  $G$ . Из двух полученных векторов выбирается тот  $X_l^{r,(k+1)}$ , для которого  $X_l^{r,(k+1)} = \arg \min_{\zeta=1,2} f_0(X_\zeta^{r,(k+1)})$ .

Описанные выше процедуры позволяют эффективно использовать каждый сгенерированный вектор.

Используемые в ГА две операции приводят к появлению новых особей, т. е. векторов  $X_l^{r, (k+1)}$ ,  $l = 1, 2, \dots, \bar{N}^{r, (k+1)}$ . При этом размер популяции, рассматриваемой на  $(k + 1)$ -м шаге процесса, каждой  $r$ -й области становится большим и равен

$$N^{r, (k+1)} = N^{r, k} + \bar{N}^{r, (k+1)}. \quad (16)$$

То есть часто используется так называемая стратегия элитизма, при которой несколько лучших индивидуумов переходят в следующее поколение без изменений, не участвуя в кроссовере и отборе.

Если в процессе вычислений будет получено какое-либо решение  $\bar{X}^r \in H^r \cap G$ , значение критерия оптимальности которого удовлетворяет соотношению

$$f(\bar{X}^r) \leq \xi\{f_0^*(X|H^p)\}, p = 1, \dots, R_1, R_1 < R, \quad (17)$$

то области  $H^p$ ,  $p = 1, \dots, R_1$ , исключаются из рассмотрения и число рассматриваемых областей сокращается:  $q = 1, \dots, R_2$ , где  $R_2 = R - R_1$ .

### 3. Вычислительная схема алгоритма

Вычислительная схема алгоритма предусматривает выполнение следующих процедур:

3.1. Многомерный параллелепипед  $H$  разбиваем на  $R$   $n$ -мерных параллелепипедов  $H^r$ ,  $r = 1, \dots, R$ , меньших размеров, удовлетворяющих соотношениям (13), по формулам (6).

3.2. В соответствии с алгоритмом 1 в каждом из параллелепипедов  $H^r$  исключаем области, не содержащие допустимых планов, в результате чего получаем параллелепипеды меньших размеров  $Q^r = H^r \cap G$ .

3.3. Исключаем из рассмотрения те  $n$ -мерные параллелепипеды, для которых в процессе выполнения алгоритма 1 определится, что  $H^r \cap G \neq \emptyset$ . Проводим переиндексацию всех подлежащих дальнейшему рассмотрению параллелепипедов

$$Q^p \subseteq H^p, p = 1, \dots, R_1,$$

меньших размеров, где

$$Q^p = Q_1^p \times Q_2^p \times \dots \times Q_j^p \times \dots \times Q_n^p, \\ Q_j^p = \{x_j^p | \bar{h}_j^{1p} \leq x_j^p \leq \bar{h}_j^{2p}\}, p = 1, \dots, R_1.$$

$$\text{Здесь } \bar{h}_j^{1p} \geq h_j^{1p}, \bar{h}_j^{2p} \leq h_j^{2p}, p = 1, \dots, R_1.$$

3.4. Для каждого из рассматриваемых параллелепипедов  $Q^p$ ,  $p = 1, \dots, R_1$ , вычислим нижнюю границу оптимального значения функции цели для векторов  $X^p \in Q^p$ , по формулам

$$\xi[f_0^*(X^p|X^p \in Q^p)] = \min_{\mu \leq l \leq M} f_0(Z^{p, \mu}),$$

где  $Z^{p, \mu}$  — вектор переменных задачи, определяющий одну из вершин ( $\mu$ -ю вершину)  $n$ -мерного параллелепипеда  $Q^p$ ,  $M = 2^n$  — число вершин этого параллелепипеда.

Заметим, что если  $f_0(X^p)$  обладают свойствами (7)–(9), т. е. являются квазимонотонными функциями в сильном смысле (оптимальное значение таких функций по каждой переменной достигается на концах отрезка ее допустимых значений вне зависимости от значений других переменных), то нижняя граница оптимального значения функции цели вычисляется по формулам, требующим существенно меньшего объема вычислений:

$$\xi[f_0^*(X^p|X^p \in Q^p)] = \\ = f_0(z_1^{p, -}, z_2^{p, -}, \dots, z_j^{p, -}, \dots, z_n^{p, -}). \quad (18)$$

Здесь  $z_j^{p, -}$  — значение  $j$ -й координаты вектора переменных  $X^p \in Q^p$ , вычисляемая по формулам, аналогичным (7), (9).

3.5. Вычислим нижнюю границу функции цели в решении задачи по формуле

$$\xi\{f_0^*(X)\} = \min_{1 \leq p \leq R_1} \xi\{f_0^*(X^p|X^p \in Q^p)\}. \quad (19)$$

3.6. Реализуем итеративные процессы на основе выполнения модифицированных генетических алгоритмов в каждом из образованных  $n$ -мерных параллелепипедов  $Q^p$ ,  $p = 1, \dots, R_1$ . На каждом шаге этого итеративного процесса выполняем следующие вычисления:

3.6.1. В каждой из подобластей  $Q^{r, 0}$  случайным образом создается некоторая начальная популяция, состоящая из  $N^{r, 0}$  особей, т. е. числовых векторов  $X_l^{r, 0} \in Q^r$ ,  $l = 1, 2, \dots, N^{r, 0}$ . Если в процессе генерации получен вектор  $Y_l^{r, 0} \notin Q^r = (H^r \cap G)$ , то проектируем эти вектора на границу области  $Q^r$  в процессе поиска соответствующего значения, например, на отрезке прямой, соединяющей вектор  $Y_l^{r, 0}$  с вектором  $Y^r = (\bar{y}_1^r, \dots, \bar{y}_j^r, \dots, \bar{y}_n^r)$ , где  $\bar{y}_j^r = 0,5(\bar{h}_j^{r, +} + \bar{h}_j^{r, -})$ ,  $j = 1, 2, \dots, n$ , алгоритмами деления этого пополам.

3.6.2. Для каждого из векторов  $X_l^{r, 0} \in Q^r$  вычисляем значение целевой функции  $f_0(X_l^{r, 0})$ ,  $l = 1, 2, \dots, N^{r, 0}$ , а также, положив  $k = 0$ , вычисляем

$$\hat{f}_0(X^r, k) = \min_{1 \leq l \leq N^{r, 0}} f_0(X_l^{r, k}), r = 1, \dots, R_1, \\ \text{и } f_0^*(X^k) = \min_{1 \leq r \leq R_1} \hat{f}_0(X^r, k). \quad (20)$$

Если для некоторой  $p$ -й подобласти справедливо выражение

$$\xi\{f_0^*(X^p | X^p \in Q^p)\} \geq f_0^*(X^k), \quad (21)$$

то она исключается из дальнейшего рассмотрения, т. е. в этом параллелепипеде ГА в дальнейшем не реализуется.

Если справедливо неравенство

$$f_0^*(X^k) - \xi\{f_0^*(X)\} \leq \varepsilon, \quad (22)$$

где  $\varepsilon$  — заданная точность решения задачи, то алгоритм завершает свою работу, и вектор  $X^k$  является решением задачи. В противном случае переходим к шагу 3.6.3.

3.6.3. Выполняем переиндексацию всех оставшихся перспективных подобластей  $Q^p$ ,  $p = 1, \dots, R_1$ . В каждой из этих перспективных подобластей на  $k$ -м шаге процесса проводим следующие вычисления:

А. Упорядочим  $N^{p,k}$  особей (векторов  $X_l^{p,k}$ ,  $l = 1, \dots, N^{p,k}$ ) в последовательность  $\tilde{U}^{p,k}$  по убыванию значений  $f_0(X_l^{p,k})$ :

$$\begin{aligned} \tilde{U}^{p,k} = \{ & X_1^{p,k}, X_2^{p,k}, \dots, X_g^{p,k}, X_{g+1}^{p,k}, \\ & \dots, X_N^{p,k} \} f_0(X_g^{p,k}) \leq f_0(X_{g+1}^{p,k}); \end{aligned} \quad (23)$$

выбираем первые  $g$  членов из последовательности (23), положив

$$g = N^{p, (k+1)}.$$

Б. Используя процедуру мутации и создавая потомство преобразованием дубликата родителя, проводим локальную вариацию в достаточно узких пределах нескольких выбранных случайным образом компонент каждого вектора переменных  $X_l^{p,k}$  (каждого гена). Значение нового гена выбирается из этого диапазона случайным образом. При этом используют равномерное, нормальное, треугольное и другие законы распределения. Каждый из генов изменяется в соответствии с выражением  $x_{l,j}^{p,k} = y_{l,j}^{p,k} \pm \delta_j(0, \sigma_j)$ .

Здесь  $x_{l,j}^{p,k}$ ,  $y_{l,j}^{p,k}$  — соответственно новое и старое значение  $j$ -го гена;  $\delta_j(0, \sigma_j)$  — случайное число, генерируемое в соответствии с заданным законом распределения с математическим ожиданием, равным 0, среднеквадратическим отклонением  $\sigma_j > 0$ .

Рекомендуется выбирать значение  $\sigma_j \leq 0,2y_{l,j}^{p,k}$ . Стратегией (1 + 1) — ЕС предусматривается создание из одного родителя одного члена потомства. После выполнения этих процедур, сохраняются также и родители потомства.

В. Оператор скрещивания (кроссовер) порождает одного или нескольких потомков от двух хромосом (т. е. одного вектора  $X_l^{p, (k+1)}$  из двух векторов  $X_{l_1}^{p,k}$  и  $X_{l_2}^{p,k}$ ). Большинство *real-coded* алгоритмов генерируют новые векторы в окрестности родительских пар случайным обменом значений соответствующих элементов компонент этих векторов. При скрещивании происходит случайный выбор позиции разделителя  $w$ ,  $w = 1, \dots, n$ , где  $n$  — число элементов в строке. Затем значения первых  $w$  элементов первой строки записываются в соответствующие элементы второй, а значения первых  $w$  элементов второй строки — в соответствующие элементы первой. В результате получаем две новые строки, каждая из которых является комбинацией частей двух родительских строк. Пользуясь биологической терминологией, этот оператор порождает одного или нескольких потомков от двух хромосом. Таким образом, образуется еще  $N^{p, (k+1)}$  дополнительных векторов  $X_l^{p, (k+1)}$ .

Отметим, что если в процессах мутации или кроссовера будет получен вектор  $Y_l^{r,0} \notin Q^r = (H^r \cap G)$ , то выполняем проецирование этих векторов на границу области  $Q^r$  алгоритмами, описанными выше в разд. 2.

Г. Общее число векторов, образованных и сохраненных на этом  $(k+1)$ -м шаге, которое обозначим  $N^{p, (k+1)}$ , упорядочим в последовательность  $\tilde{U}^{p, (k+1)}$  в соответствии с выражением (23). В каждой из подобластей выбираем первые  $g$  членов из последовательности (23), положив  $\bar{g} = N^{p, (k+1)}$ . Переходим к выполнению п. 3.6 алгоритма.

Через некоторое число шагов либо будет выполнено условие (22) и алгоритм завершит свою работу, либо через наперед заданное число шагов  $\bar{K}$  алгоритм завершит свою работу, выбрав решение, удовлетворяющее соотношениям (20) для  $k = \bar{K}$  в качестве оптимального решения задачи.

Автором были проведены вычислительные эксперименты по анализу эффективности рассмотренных выше методов. Решались задачи 3—10 переменных. При этом в качестве функции цели и левых частей ограничений задачи выбирались линейные, квадратические, аддитивно-мультипликативные, степенные и логарифмические функции, а также другие функции, виды которых приведены в работе автора [2]. Как показали проведенные исследования и вычислительные эксперименты, использование описанных выше модификаций позволяет существенно повысить эффективность работы ГА, оценить точность полученных приближенных решений и сократить объемы вычислений. Одинаковые по точности результаты были получены при объеме вычислений в 1,8—3,5 раз меньшем, чем при использовании генетических алгоритмов в традиционной модификации.

## Заклучение

Определен класс многоэкстремальных задач оптимизации в условиях ограничений, для которых могут быть вычислены оценки оптимального решения задачи, а также выделены и исключены из рассмотрения области, не содержащие допустимых и оптимальных решений. Рассмотрены алгоритмы выделения таких областей, что позволило сократить область генерации, преобразования и воспроизводства как отдельных хромосом, так и целого потомства.

Предложены модификации генетических алгоритмов (GA) для решения экстремальных задач в условиях ограничений, в которых допустимая область поиска решений разбивается на несколько подобластей существенно меньшей размерности. На основе свойств допустимых решений исключаются из рассмотрения области, не содержащие допустимых и неоптимальных решений, что позволяет существенно сократить область поиска. GA реализуется в каждой из выделенных подобластей. При этом результаты, полученные в каждой области, сравниваются. Это позволяет реализовать процесс вычислений на многопроцессорных ЭВМ.

Предложены алгоритмы проецирования генерируемых (т. е. полученных в процессе мутации и скрещивания) векторов на область ограничений задачи.

Предложенные формулы вычисления оценки оптимального значения функции цели задачи позволяют остановить работу алгоритма, когда получено решение, удовлетворяющее заданным априори требованиям точности, а также в процессе решения исключать целые области, не содержащие оптимальных решений.

Проведенные вычислительные эксперименты подтвердили эффективность предложенных методов для рассматриваемых в работе важных для практических приложений функций достаточно широкого класса.

## Список литературы

1. Емельянов В. В., Курейчик В. В., Курейчик В. М. Теория и практика эволюционного моделирования. М.: Физматлит, 2003. 432 с.
2. Зак Ю. А. Методы многоэкстремальной оптимизации в условиях ограничений для сепарабельно квазимонотонных функций // Известия РАН. Теория и системы управления. 2011. № 2. С. 38–55.
3. Зак Ю. А. Адаптивные методы глобального случайного поиска в решении одного класса многоэкстремальных задач в условиях ограничений // Информационные технологии. 2012. № 1. С. 20–27.
4. Гладков Л. А., Курейчик В. В., Курейчик В. М. Генетические алгоритмы: учебное пособие. 2-е изд. М.: Физматлит, 2006. 320 с.
5. Курейчик В. М., Лебедев Б. К., Лебедев О. К. Поисковая адаптация: теория и практика. М.: Физматлит, 2006. 272 с.
6. Goldberg D. E. Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley, 1989.

Yu. A. Zack, Dokt.-Ing., Deutschland, Aachen, Deutschland  
E-mail: yuriy\_zack@hotmail.com, сайт: www.optimorum.de

## Modified Genetic Algorithm for Solving Multiextremal Problems within the Constraints

*Proposed modifications of genetic algorithms for solving extremal problems in the constraints, which by simple formulas can be calculated by assessing the optimal solutions and the boundary values of the left sides of constraint functions of the problem. Permissible scope of the search of solutions is divided into several sub-areas significantly smaller dimension, in each of which implements an evolutionary strategy. Use of estimates of optimal solutions, with the exception of areas that do not contain admissible and optimal solutions, the use of projection operators on the constraints of the problem area can improve the efficiency of genetic algorithms for the considered classes of functions.*

**Keywords:** genetic algorithms, separable monotone functions, domains containing admissible and optimal plans, estimates of optimal solutions

## References

1. Emeljanov V. V., Kurejchik V. V., Kurejchik V. M. Teorija i praktika evoluzionnogo modelirovanija. M.: Fizmatgis, 2003. 432 p.
2. Zack Yu. A. Metodi mnogoekestremalnoj optimisazii v uslovijach ograničenij dlya separabelno kvasimonotonnich funkcij. *Izvestija RAN. Teorija i sistemi upravljenja*. 2011. N. 2. P. 38–55.
3. Zack Yu. A. Adaptivnije metodi globalnogo poiska v reshenii odnogo klassa mnogoekestremalnich zadach v uslovijach ograničenij. *Informazionnije tehnologii*, 2012. N. 1. P. 20–27.
4. Gladkov L. A., Kurejchik V. V., Kurejchik V. M. *Geneticheskie algoritmi: Uchebnoje posobie*. 2-e izdanie. M.: Fizmatgis, 2006. 320 p.
5. Kurejchik V. M., Lebedev B. K., Lebedev O. K. *Poiskovaja adaptazija: teorija i praktika*. M.: Fizmatgis, 2006. 272 p.
6. Goldberg D. E. Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley, 1989.

**Р. В. Воронов**, канд. техн. наук, доц., e-mail: rvoronov76@sampo.ru,  
**А. П. Мошевикин**, канд. физ.-мат. наук, доц., e-mail: alexmou@lab127.karelia.ru,  
 Петрозаводский государственный университет

## Применение условной энтропии при формировании рекомендаций по размещению базовых станций в локальных системах позиционирования\*

*Описывается одно из решений задачи оптимального размещения базовых станций на местности для улучшения точности локации мобильных объектов. Для этого используется понятие информационной энтропии. При оценке выбранной конфигурации базовых станций предлагается минимизировать суммарную энтропию уровней сигналов во всех зонах территории. Описываемый подход может быть использован в системах локального позиционирования, основанных на построении карт уровней сигналов базовых станций, например, WiFi, ZigBee, Bluetooth, nanoLOC.*

**Ключевые слова:** локальная система позиционирования, карта уровней сигнала, условная энтропия

### Введение

Глобальные спутниковые системы навигации (ГЛОНАСС, GPS, Galileo) невозможно использовать в закрытых помещениях, например, в зданиях, шахтах, подземных сооружениях, поэтому разрабатываются специальные системы обнаружения объектов. При этом для определения местоположения мобильных объектов используют различные методы, в том числе основанные на определении уровней сигнала в беспроводных сетях радиостанций [1—3]. В помещении устанавливают стационарные базовые станции сети, каждая из которых имеет свою площадь покрытия. Необходимым условием для функционирования системы локации является трудоемкая предварительная ее настройка — разбиение пространства помещения на зоны и определение для каждой из них среднего значения уровня принимаемых сигналов от базовых станций. Локация объекта определяется на основе регистрируемых им уровней сигналов от базовых станций путем выбора зоны с наиболее близкими средними значениями уровней сигналов. В статье рассматривается задача автоматического определения в здании мест наилучшего расположения базовых станций. Такая задача возникает в устанавливаемых, а также в действующих системах локации, например, при изменении конфигурации помещений, перестановке мебели, добавлении или удалении перегородок и т. д.

Актуальность работы обусловлена ростом внедрения и использования систем локации в закрытых помещениях. В ходе инсталляции таких систем всегда стараются уменьшить число используемых базовых станций в целях минимизации стоимости

оборудования. Поэтому зачастую применяют следующий итерационный сценарий. Сначала устанавливают "узловые" базовые станции в некоторых местах здания, вручную обучают систему, строят карты уровней сигнала, а затем по мере необходимости в разных частях здания добавляют еще несколько станций, улучшая точность локации.

Стандартный подход при оценивании точности локации заключается в экспериментальном тестировании системы локации и оценивании среднего значения расстояния между реальным местоположением объекта и местоположением, определенным системой локации. Недостатком такого подхода являются временные затраты на проведение такого эксперимента.

Для количественной оценки эффективности схемы расположения базовых станций предлагается использовать понятие *информационной энтропии*. В математической модели все здание представляет собой множество зон. Физически зонам соответствуют небольшие комнаты, части комнат или коридоров. Разбиение на зоны плана помещения осуществляется, например, путем накладывания прямоугольной сетки — в этом случае каждая зона представляет собой прямоугольный участок. Как правило, зоны не содержат внутри себя стен или перегородок. Задача локации объекта заключается в определении зоны, в которой находится объект. Точность локации характеризуется правильностью указания зоны.

Пусть имеется множество возможных позиций установки стационарных базовых станций беспроводной сети. Предположим, что для каждой такой позиции известны распределения вероятностей уровней сигналов во всех зонах от расположенной в ней базовой станции. Отметим, что распределения вероятностей уровней сигналов и их параметры могут быть предложены и оценены на основе математических моделей распределения сигнала в помещениях [4—6]. Используемые на практике алгоритмы локации оценивают условные вероятности возможного расположения объекта в зонах помещения после регистрации объектом уровней сигналов от

\* Исследования, описанные в статье, проводились в рамках деятельности МИП ООО "Наносети" и ЗАО "РТЛ-Сервис" а также поддерживались Петрозаводским государственным университетом (Программа стратегического развития ПетрГУ на 2012—2016 гг.), Министерством образования и науки РФ (гос. контракт 14.BVV.21.0162), Фондом СР МФП в НТС, Американским фондом гражданских исследований и развития (CRDF) и Министерством экономического развития Республики Карелии.

базовых станций. Таким образом, можно говорить про неопределенность информации о местоположении объекта. Информационная энтропия позволяет оценивать такую информационную неопределенность. Чем меньше информационная энтропия, оценивающая неопределенность местоположения объекта, тем выше точность локации. Если имеется ограничение на число используемых базовых станций, то можно искать такое подмножество позиций расположения базовых станций, для которого суммарная информационная энтропия для всех зон на данной территории примет минимальное значение.

Описанные ниже подходы и алгоритмы были применены при разработке технологии локального позиционирования RealTrac [7–9]. Было выявлено, что использование конфигурации базовых станций с минимальной оценкой информационной энтропии обеспечивало наилучшую "стандартную" точность локации объектов. Испытание системы локации с оптимальным расположением базовых станций гарантировало среднюю точность локации объектов, равную примерно 2 м. Напротив, применение неудачных способов мест расположения базовых станций приводило к средним ошибкам локации, достигающим 5 м и более.

### Математическая модель

Приведем определение информационной условной энтропии [10].

Пусть имеется два опыта  $\alpha$  и  $\beta$  с исходами соответственно  $A_1, \dots, A_M$  и  $B_1, \dots, B_N$ . Тогда условная энтропия  $H_\alpha(\beta)$  опыта  $\beta$  при условии выполнения опыта  $\alpha$  вычисляется по формуле

$$H_\alpha(\beta) = - \sum_{j=1}^N \sum_{i=1}^M p(B_j, A_i) \log_2 p(B_j|A_i),$$

где  $p(B_j, A_i)$  — вероятность того, что опыт  $\beta$  имеет исход  $B_j$ , а опыт  $\alpha$  имеет исход  $A_i$ ;  $p(B_j|A_i)$  — вероятность того, что опыт  $\beta$  имеет исход  $B_j$  при условии, что опыт  $\alpha$  имеет исход  $A_i$ . Свойства условной энтропии можно найти в работе [10].

Далее в математической модели условная энтропия будет применена для количественной оценки выбираемых мест расположения базовых станций системы локации.

Введем необходимые обозначения:

$X$  — дискретное множество зон помещения, в которых может находиться мобильный объект;

$|X|$  — число элементов (мощность) множества  $X$ ;

$p(x)$  — априорная вероятность нахождения мобильного объекта в зоне  $x \in X$ ;

$D$  — дискретное множество всех возможных позиций расположения стационарных базовых станций. Для краткости будем его называть "множеством базовых станций", а вместо термина "базовая станция, расположенная в позиции  $d \in D$ " будем использовать "базовая станция  $d \in D$ ";

$S$  — дискретное множество допустимых значений уровней сигнала от всех базовых станций;

$S(\Omega)$  — дискретное множество значений уровней сигнала от базовых станций множества  $\Omega \subset D$ ;

$s_d$  — уровень сигнала, регистрируемого мобильным объектом от базовой станции  $d \in D$  ( $s_d \in S$ );

$s$  — вектор уровней сигналов, регистрируемых мобильным объектом от множества базовых станций  $D$  или любого его подмножества;

$p(s, x)$  — вероятность одновременного выполнения двух событий: мобильный объект находится в зоне  $x \in X$  и им был зарегистрирован вектор уровней сигнала  $s$ ;

$p(s|x)$  — вероятность того, что мобильным объектом зарегистрирован вектор уровней сигнала  $s$  при условии, что он находится в зоне  $x \in X$ ;

$p(s_d|x)$  — вероятность того, что мобильным объектом зарегистрирован уровень сигнала  $s_d$  от базовой станции  $d \in D$  при условии, что он находится в зоне  $x \in X$ ;

$p(x|s)$  — вероятность того, что мобильный объект находится в зоне  $x \in X$  при условии, что зарегистрирован вектор уровней сигнала  $s$ .

Отметим, что вероятности  $p(x|s)$  оцениваются в ходе работы системы локации при определении местоположения мобильного объекта.

Общепринятым [8] является предположение, состоящее в том, что для каждой зоны  $x \in X$  регистрация мобильным объектом уровней сигналов от базовых станций независима в совокупности, т. е.

$$p(s|x) = \prod_{d \in D} p(s_d|x).$$

Вероятность зафиксировать вектор уровней сигналов  $s$  рассчитывается в соответствии с формулой полной вероятности:

$$p(s) = \sum_{x' \in X} p(x')p(s|x').$$

Пусть  $\Omega \subset D$  — некоторое подмножество базовых станций. Предположим, что опыт  $\beta$  состоит в определении зоны  $x \in X$ , в которой находится мобильный объект, а опыт  $\alpha(\Omega)$  — в регистрации вектора уровней сигнала от базовых станций множества  $\Omega \subset D$ . Тогда условная энтропия  $H_{\alpha(\Omega)}(\beta)$  будет означать неопределенность нахождения мобильного объекта после обработки зарегистрированных им уровней сигнала от базовых станций. Приведем формулу для расчета  $H_{\alpha(\Omega)}(\beta)$ :

$$\begin{aligned} H_{\alpha(\Omega)}(\beta) &= - \sum_{s \in S(\Omega)} \sum_{x \in X} p(x, s) \log_2 p(x|s) = \\ &= - \sum_{s \in S(\Omega)} \sum_{x \in X} p(x, s) \log_2 \frac{p(x, s)}{p(s)} = \\ &= - \sum_{s \in S(\Omega)} \sum_{x \in X} p(x)p(s|x) \log_2 \frac{p(x)p(s|x)}{\sum_{x' \in X} p(x')p(s|x')} = \\ &= - \sum_{s \in S(\Omega)} \sum_{x \in X} p(x) \prod_{d \in \Omega} p(s_d|x) \log_2 \frac{p(x) \prod_{d \in \Omega} p(s_d|x)}{\sum_{x' \in X} p(x') \prod_{d \in \Omega} p(s_d|x')}. \end{aligned}$$

В простейшем случае можно считать априорные вероятности  $p(x)$  одинаковыми:

$$p(x) = \frac{1}{|X|}, x \in X,$$

т. е. изначально, до измерения уровней сигналов, все зоны равновероятны. В этом случае формула для  $H_{\alpha(\Omega)}(\beta)$  примет более простой вид:

$$F(\Omega) = H_{\alpha(\Omega)}(\beta) = -\frac{1}{|X|} \sum_{s \in S(\Omega)} \sum_{x \in X} \prod_{d \in \Omega} p(s_d|x) \log_2 \frac{\prod_{d \in \Omega} p(s_d|x)}{\sum_{x' \in X} \prod_{d \in \Omega} p(s_d|x')}. \quad (1)$$

Полученное значение условной энтропии  $H_{\alpha(\Omega)}(\beta)$  можно интерпретировать следующим образом. Если  $H_{\alpha(\Omega)}(\beta) = 0$ , то после регистрации уровней сигналов определяется та единственная зона, в которой находится мобильный объект. Если же условная энтропия принимает максимально возможное значение  $H_{\alpha(\Omega)}(\beta) = \log_2 |X|$ , то это означает, что объект с равной вероятностью может находиться в любой из зон множества  $X$ . Если  $H_{\alpha(\Omega)}(\beta)$  принимает значения между нулем и  $\log_2 |X|$ , то, как правило, после обработки уровней сигналов от базовых станций будут выделены зоны, в которых может с большей вероятностью, чем в других зонах, находиться мобильный объект. Чем ближе  $H_{\alpha(\Omega)}(\beta)$  к нулю, тем меньше будет выделяться вероятных зон нахождения объекта, т. е. локация будет точнее.

Применим свойство условной энтропии не возрастать при добавлении еще одного опыта в условие. Пусть  $\Omega \subset \tilde{\Omega} \subset D$ , т. е.  $\tilde{\Omega}$  — множество базовых станций, являющееся надмножеством  $\Omega$ . Тогда

$$H_{\alpha(\tilde{\Omega})}(\beta) \leq H_{\alpha(\Omega)}(\beta).$$

Это означает, что условная энтропия не увеличивается при добавлении новых базовых станций. Следовательно,  $H_{\alpha(\Omega)}(\beta)$  примет минимальное значение при  $\Omega = D$ , т. е. при применении всех базовых станций.

Сформулируем задачу оптимизации. Пусть задан предельный уровень  $\lambda$  условной энтропии, необходимо найти минимальное по числу элементов подмножество базовых станций  $\Omega \subset D$ , для которого  $F(\Omega) \leq \lambda$ :

$$|\Omega| \rightarrow \min, \quad (2)$$

$$\Omega \subset D, \quad (3)$$

$$F(\Omega) \leq \lambda. \quad (4)$$

Очевидно, что задача (2)—(4) имеет решение тогда и только тогда, когда  $\lambda \geq F(D)$ .

Для решения задачи (2)—(4) необходимо выделить все возможные места расположения базовых станций — так будет построено множество  $D$ . Затем для каждой пары зона  $x \in X$  — базовая станция  $d \in D$ , необходимо определить распределения вероятностей  $p(s_d|x)$  уровней сигнала.

Очевидно, что задача (2)—(4) может быть решена методом полного перебора всех подмножеств множества  $D$ , что потребует времени, пропорционального  $2^{|D|}$ . На практике это возможно, когда число базовых станций не велико (2—3 десятка).

Докажем, что задача (2)—(4) является труднорешаемой. Для этого сформулируем ее как задачу разрешимости P1:

*Пусть задано рациональное число  $\lambda$  и положительное целое число  $K \leq |D|$ . Существует ли такое подмножество  $\Omega \subset D$ , что  $|\Omega| \leq K$  и  $F(\Omega) \leq \lambda$ ?*

Нам потребуется вспомогательная задача, известная под названием МЕТРИЧЕСКАЯ РАЗМЕРНОСТЬ [11], формулируемая следующим образом:

*Заданы граф  $G = (V, E)$  и положительное целое число  $K \leq |V|$ . Существует ли такое подмножество  $V' \subset V$ , что  $|V'| \leq K$ , и для каждой пары вершин  $u, v \in V'$  найдется такая вершина  $w \in V'$ , что длина кратчайшего пути от  $u$  к  $w$  отлична от длины кратчайшего пути между  $v$  и  $w$ ? В случае существования такое подмножество  $V'$  называется базисом метрик графа  $G$ .*

Известно, что задача МЕТРИЧЕСКАЯ РАЗМЕРНОСТЬ является NP-полной [11].

**Утверждение.** Задача P1 является NP-полной.

**Доказательство.** Принадлежность задачи P1 классу NP очевидна — достаточно предъявить искомое подмножество  $\Omega \subset D$  и проверить условия задачи. Покажем, что к задаче P1 за полиномиальное время сводится задача МЕТРИЧЕСКАЯ РАЗМЕРНОСТЬ. Пусть  $\delta(u, v)$  — длина кратчайшего пути между вершинами  $u$  и  $v$ ,  $X = V$ ,  $D = V$ ,  $S = \{s|\exists u, v \in V, \delta(u, v) = s\}$  — множество значений весов кратчайших путей,  $\lambda = 0$ , вероятности

$$p(s_d|x) = \begin{cases} 1, & \text{если } \delta(x, d) = s_d, \\ 0, & \text{иначе} \end{cases}$$

(интерпретация такова: мобильный объект, находясь в зоне  $x \in X$ , всегда регистрирует от базовой станции  $d \in D$  сигнал одного уровня  $\delta(x, d)$ ).

Докажем, что существует  $\Omega \subset D$  и  $|\Omega| \leq K$  и  $F(\Omega) = 0$  тогда и только тогда, когда в графе  $G$  имеется базис метрик и мощности не более  $K$ .

Пусть  $\Omega \subset D$  и  $|\Omega| \leq K$  и  $F(\Omega) = 0$ . Это означает, что для каждой пары  $(s, x)$ , где  $s \in S(\Omega)$ ,  $x \in X$ , выполнено одно из двух условий:

- либо  $p(s, x) = 0$ ;
- либо в сумме  $\sum_{x' \in X} p(s|x')$  все слагаемые, кроме

$p(s|x)$ , равны нулю.

При этом  $p(s|x) = 1$ .

Это означает, что для каждого вектора  $s \in S$  существует не более одной зоны  $x \in X$ , для которой  $p(s|x) = 1$ . Положим  $V' = \Omega$ . Покажем, что для любых  $u, v \in V'$  найдется  $w \in V'$  и что  $\delta(u, w) \neq \delta(v, w)$ . Пусть это не так. Тогда найдутся  $x, y \in X$ , что для любых  $w \in V'$   $\delta(x, w) = \delta(y, w)$ . Пусть  $s = \{\delta(x, w)|w \in V'\}$ . Тогда, в силу построения  $p(s|x) = 1$  и  $p(s|x') = 1$  обнаруживается противоречие.

Обратно, пусть существует  $V' \subset V, |V'| \leq K$  и для любых  $u, v \in V$  найдется  $w \in V'$  с  $\delta(u, w) \neq \delta(v, w)$ . Положим  $\Omega = V'$ . Тогда для каждой зоны  $x \in X$  существует  $s = \{\delta(x, d) | d \in \Omega\}$ , для которого  $p(s|x) = 1$ , при этом для всех  $x' \in X, x' \neq x$  будет  $p(s|x') = 0$ . Тогда  $F(\Omega) = 0$ . Утверждение доказано.

Труднорешаемость задачи (2)–(4) означает необходимость использования на практике приближенных методов ее решения.

На практике может быть полезна следующая задача. Задано число  $K$ . Необходимо найти подмножество базовых станций  $\Omega \subset D$ , число элементов которого  $|\Omega|$  равно  $K$  и энтропия  $F(\Omega)$  принимает минимальное значение:

$$F(\Omega) \rightarrow \min, \quad (5)$$

$$\Omega \subset D, \quad (6)$$

$$|\Omega| = K. \quad (7)$$

В данном случае множество  $D$  определяет места возможного расположения базовых станций, в задаче требуется выбрать из них ровно  $K$ .

Аналогично задаче (2)–(4), задача (5)–(7) является труднорешаемой и эффективно она может быть решена только с помощью приближенного алгоритма, в качестве которого предлагается следующий жадный эвристический алгоритм.

**Шаг 1.** Находим базовую станцию  $d \in D$ , для которой достигается минимум условной энтропии  $F(\{d\})$ . Положим  $\Omega_1 = \{d\}$ . Переход на шаг 2.

**Шаг  $k$ .** Находим базовую станцию  $d \in D$ , для которой достигается минимум условной энтропии  $F(\Omega_{k-1} \cup \{d\})$ . Положим  $\Omega_k = \Omega_{k-1} \cup \{d\}$ . Если  $k = K$ , то положим  $\Omega = \Omega_k$  и СТОП; иначе — переход на шаг  $k + 1$ .

В данном алгоритме на первом шаге находится такое местоположение первой базовой станции, для которого будет минимальна условная энтропия определения местоположения мобильного объекта при регистрации сигнала от данной базовой станции. Найденное местоположение базовой станции фиксируется и в дальнейшем не меняется.

Далее ищется местоположение второй базовой станции, для которого будет минимальна условная энтропия определения местоположения мобильного объекта при регистрации уровней сигналов от первой и второй базовых станций. Найденное местоположение второй базовой станции фиксируется и также в дальнейшем не меняется.

Далее так поступают для поиска местоположения третьей, четвертой и т. д. до  $K$ -й базовых станций.

### Тестирование алгоритма

Для проверки адекватности разработанной модели и работоспособности разработанного алгоритма были проведены тестовые испытания. В трех примерах определялись места расположения базовых станций в помещении, моделируемом специально построенной прямоугольной решеткой. Затем приводятся результаты расчета на реальных данных.

Пусть в качестве множества зон  $X$  взята прямоугольная решетка. Затухание уровня сигнала между базовой станцией и приемником в какой-нибудь зоне решетки моделировалось так: если базовая станция  $d$  находится в зоне  $y = (u, v)$ , то среднее значение уровня сигнала в зоне  $x = (i, j)$  принималось равным  $s_y[x] = C - \max\{|u - i|, |v - j|\}$ , где  $C$  — некоторая константа. В этом случае уровень сигнала ослабевает на одну единицу при переходе в соседнюю клетку. Для примера в табл. 1 представлены средние значения уровней принятого сигнала при условии, что  $C = 63$  и базовая станция находится в клетке (1, 3).

Таблица 1

Средние значения уровней принятого сигнала

$i \backslash j$	1	2	3	4
1	61	62	63	62
2	61	62	62	62
3	61	61	61	61

### Пример 1.

Пусть условные вероятности уровней принятого сигнала в клетках рассчитываются по формулам

$$p(s_d|x) = \begin{cases} 1, & \text{если } s_d = s_y[x]; \\ 0, & \text{иначе.} \end{cases}$$

Это означает, что отсутствует случайная компонента в величине затухания уровня сигнала.

Рассмотрим результаты вычислений для решетки  $2 \times 4$ . Пусть первая базовая станция находится в зоне (1, 1). В табл. 2 приведены значения условной энтропии, вычисленные по формуле (1), в зависимости от расположения второй базовой станции.

Таблица 2

Значения условной энтропии для первого эксперимента

$i \backslash j$	1	2	3	4
1		0,75	0,5	0,5
2	0,75	0,75	0,5	0,5

Из табл. 2 видно, что вторая базовая станция должна находиться в любой из ячеек третьего или четвертого столбца.

### Пример 2.

Пусть условные вероятности уровней принятого сигнала в клетках рассчитываются по формулам

$$p(s_d|x) = \begin{cases} 0,4, & \text{если } s_d = s_y[x]; \\ 0,2, & \text{если } |s_d - s_y[x]| = 1; \\ 0,1, & \text{если } |s_d - s_y[x]| = 2; \\ 0, & \text{иначе.} \end{cases}$$

Это означает, что присутствует некоторая случайная компонента в величине затухания уровня сигнала.

Рассмотрим результаты вычислений для решетки  $2 \times 4$ . Пусть первая базовая станция находится в зоне (1, 1). В табл. 3 приведены значения условной энтропии в зависимости от расположения второй базовой станции.

Таблица 3

Значения условной энтропии для второго эксперимента

$i \backslash j$	1	2	3	4
1		2,343	2,323	2,187
2	2,190	2,343	2,323	2,187

Из табл. 3 видно, что вторая базовая станция должна находиться в любой из ячеек четвертого столбца.

Сравнение результатов первого и второго примеров указывает на то, что наличие случайной компоненты влияет на выбор оптимального расположения базовых станций.

### Пример 3.

Пусть

$$s_y[x] = C - \min\{\max\{|u - i|, |v - j|\}, R\}.$$

В этом случае сигнал перестает ослабевать через  $R$  клеток. Точнее — на расстоянии  $R$  клеток от базовой станции сигнал ослабел настолько, что по уровню принятого сигнала клетки неразличимы друг от друга.

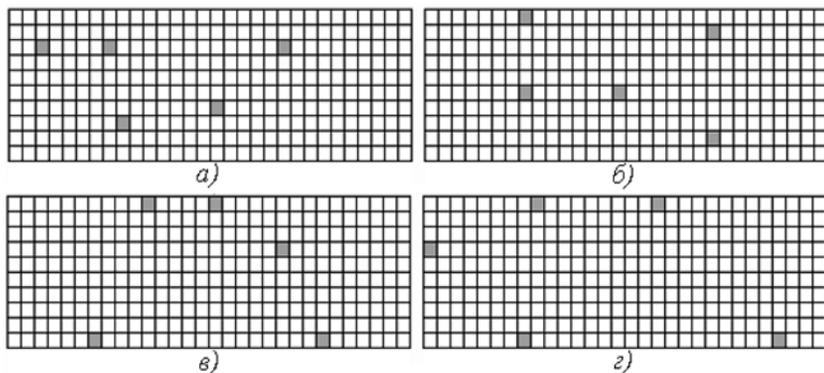


Рис. 1. Пример размещения пяти базовых станций для различных  $R$ : а —  $R = 5$ ; б —  $R = 10$ ; в —  $R = 20$ ; г —  $R = 30$

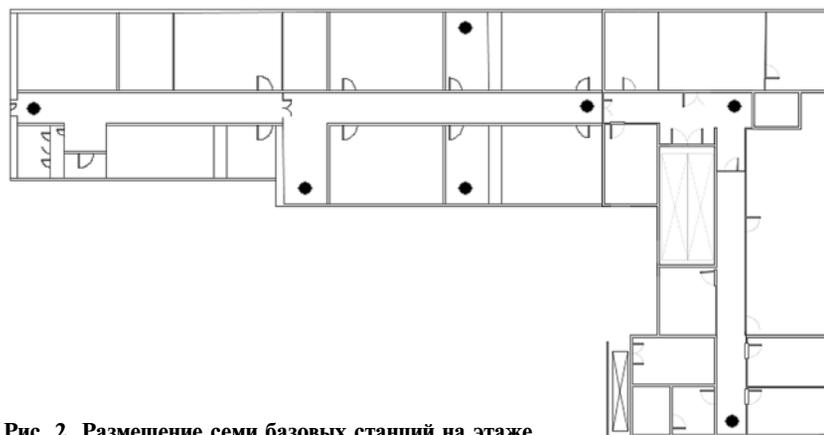


Рис. 2. Размещение семи базовых станций на этаже

Пусть условные вероятности уровней принятого сигнала в клетках рассчитываются по формулам

$$p(s_d|x) = \begin{cases} 0,4, & \text{если } s_d = s_y[x]; \\ 0,2, & \text{если } |s_d - s_y[x]| = 1; \\ 0,1, & \text{если } |s_d - s_y[x]| = 2; \\ 0, & \text{иначе.} \end{cases}$$

Рассмотрим результаты вычислений для решетки  $10 \times 30$  для пяти базовых станций. На рис. 1 для различных значений  $R$  серым цветом отмечены местоположения базовых станций, найденные с помощью приближенного жадного алгоритма.

Видно, что чем больше значение  $R$ , тем менее плотно располагаются базовые станции.

**Эксперимент на реальных данных.** Далее был проведен вычислительный эксперимент на данных, собранных в ходе эксплуатации системы локации на одном из этажей корпуса ИТ-парка ПетрГУ. Часть помещения была разбита на 125 квадратных зон размером  $2 \times 2$  м. С помощью описанного выше алгоритма были выбраны места расположения семи базовых станций (рис. 2).

Тестирование системы локации показало, что полученная расстановка базовых станций обеспечила в 75 % случаев ошибки в обнаружении мобильных объектов, не превосходящие 2,5 м. Такая точность локации была признана вполне допустимой и подтвердила практическую применимость разработанного алгоритма.

### Заключение

Для решения задачи оптимального размещения базовых станций на местности в системах локального позиционирования предложено использовать понятие информационной энтропии. Малые значения энтропии уровней сигналов на практике соответствуют высокой точности локации. Поэтому при оценке выбранной конфигурации базовых станций предлагается минимизировать суммарную энтропию во всех зонах территории.

Такая методика позволяет также выявлять участки здания, в которых неопределенность расположения мобильного объекта наиболее высока. С практической точки зрения это формирует набор рекомендаций по добавлению стационарных базовых станций в систему в целях улучшения точности позиционирования мобильных объектов. В статье также приведены примеры работы разработанных алгоритмов.

Предложенный подход и алгоритмы были применены при разработке технологии локального позиционирования RealTrac.

## Список литературы

1. Elnahrawy E., Li X., Martin R. The limits of localization using signal strength: A comparative study // *Proceedings of IEEE SECON*. 2004. P. 406–414.
2. Galov A. S., Moschevikin A. P., Voronov R. Combination of RSS localization and ToF ranging for increasing positioning accuracy indoors // *Proc. of the 11th International Conference ITST—2011*. 2011. P. 299–304.
3. Воронов Р. В., Малодушев С. В. Динамическое создание карт уровня wifi-сигналов для систем локального позиционирования // *Системы и средства информатики*. 2014. Т. 24, № 1. С. 79–91.
4. Molisch A. F., Balakrishnan K., Chong C.-C., Emami S., Fort A., Karedal J., Kunisch J., Schantz H., Schuster U., Siwiak K. IEEE 802.15.4a channel model-final report // *Tech. Rep., Document IEEE 802.1504-0062-02-004a*. 2005.
5. Ata O. W., Ala'Eddin M. S., Jawadeh M. L., Amro A. I. An indoor propagation model based on a novel multi wall attenuation loss formula at frequencies 900 MHz and 2.4 GHz // *Wireless Personal Communications*. 2013. V. 69, N. 1. P. 23–36.
6. Chrysikos T., Georgopoulos G., Kotsopoulos S. Site-specific validation of ITU indoor path loss model at 2,4 GHz // *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks & Workshops*, 2009. P. 1–6.
7. Moschevikin A., Voronov R., Galov A., Soloviev A. Using Pressure Sensors for Floor Identification in Wireless Sensors Networks // *Proc. of the 1st IEEE International Symposium on Wireless Systems within the Conferences on Intelligent Data Acquisition and Advanced Computing System IDAACS-SWS*, Offenburg, Germany, September 20–21, 2012, P. 2–6.
8. Galov A., Moschevikin A. Bayesian filters for TOP and RSS measurements for indoor positioning of a mobile object // *Proc. of the International Conference on Indoor Positioning and Indoor Navigation (IPIN—2013)*, Montbeliard, France, October 28–31, 2013. P. 310–317.
9. Moschevikin A., Galov A., Soloviev A., Mikov A., Volkov A., Reginya S. Realtrac technology overview // *Evaluating AAL Systems Through Competitive Benchmarking (ser. Communications in Computer and Information Science)* / Eds. Springer Berlin Heidelberg. 2013. V. 386. P. 60–71. [Online]. Available: <http://dx.doi.org/0.1007/978-3-642-41043-76>.
10. Яглом А. М., Яглом И. М. Вероятность и информация. М.: Наука, 1973.
11. Гэри М., Джонсон Д. Вычислительные машины и труднорешаемые задачи. М.: Мир, 1982.

R. V. Voronov, Associate Professor, e-mail: rvoronov@sampo.ru,

A. P. Moschevikin, Associate Professor, e-mail: alexmou@lab127.karelia.ru, Petrozavodsk State University

## Use of Conditional Entropy for Optimal Disposition of Base Stations in Local Positioning Systems

*The efficiency of local positioning systems depends on the number of used based stations. To increase the accuracy of localization, base stations should be deployed as dense as possible. However this practice overloads radio bandwidth and is not commercially attractive. This paper describes a solution for optimal disposition of base stations to improve the accuracy of localization of a mobile device in wireless networks. This solution is based on the concept of information entropy. The results of simulations are discussed. The best configuration of base stations corresponds to the minimum of the total entropy of the received signal levels for all zones within a territory. The described approach might be used in local positioning systems based on the fingerprinting technique (WiFi, ZigBee, Bluetooth); it was applied in the developed RealTrac technology utilizing nanoLOC radio standard.*

**Keywords:** local positioning systems, received signal strength, base stations, fingerprinting, conditional entropy

### References

1. Elnahrawy E., Li X., Martin R. The limits of localization using signal strength: A comparative study. *Proceedings of IEEE SECON*. 2004. P. 406–414.
2. Galov A. S., Moschevikin A. P., Voronov R. Combination of RSS localization and ToF ranging for increasing positioning accuracy indoors. *Proc. of the 11th International Conference on ITS Telecommunications ITST—2011*. P. 299–304.
3. Voronov R. V., Malodushev S. V. Dinamicheskoe sozhdanie kart urovnja wifi-signalov dlja sistem lokal'nogo pozicionirovanija. *Sistemy i sredstva informatiki*. 2014. V. 24, N. 1. P. 79–91.
4. Molisch A. F., Balakrishnan K., Chong C.-C., Emami S., Fort A., Karedal J., Kunisch J., Schantz H., Schuster U., Siwiak K. IEEE 802.15.4a channel model-final report. *Tech. Rep., Document IEEE 802.1504-0062-02-004a*, 2005.
5. Ata O. W., Ala'Eddin M. S., Jawadeh M. I., Amro A. I. An indoor propagation model based on a novel multi wall attenuation loss formula at frequencies 900 MHz and 2,4 GHz. *Wireless Personal Communications*. 2013. V. 69, N. 1. P. 23–36.
6. Chrysikos T., Georgopoulos G., Kotsopoulos S. Site-specific validation of ITU indoor path loss model at 2,4 GHz. *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks & Workshops*. 2009. P. 1–6.
7. Moschevikin A., Voronov R., Galov A., Soloviev A. Using Pressure Sensors for Floor Identification in Wireless Sensors Networks. *Proc. of the 1th IEEE International Symposium on Wireless Systems within the Conferences on Intelligent Data Acquisition and Advanced Computing Systems IDAACS-SWS*. Offenburg, Germany, September 20–21, 2012. P. 2–6.
8. Galov A., Moschevikin A. Bayesian filters for ToF and RSS measurements for indoor positioning of a mobile object. *Proc. of the International Conference on Indoor Positioning and Indoor Navigation (IPIN—2013)*. Montbeliard, France, October 28–31. 2013. P. 310–317.
9. Moschevikin A., Galov A., Soloviev A., Mikov A., Volkov A., Reginya S. Realtrac technology overview. *Evaluating AAL Systems Through Competitive Benchmarking (ser. Communications in Computer and Information Science)*. Eds. Springer Berlin Heidelberg. 2013. V. 386. P. 60–71. [Online]. Available: <http://dx.doi.org/10.1007/978-3-642-41043-76>.
10. Jaglom A. M., Jaglom I. M. *Verojatnost' i informacija*. М.: Nauka, 1973.
11. Gjeri M., Dzhonson D. *Vychislitel'nye mashiny i trudnoreshaemye zadachi*. М.: Mir, 1982.

УДК 621.3.049.77

А. Н. Соловьев, д-р техн. наук, проф., гл. науч. сотр., e-mail: Alexander.Soloviev@dinfo.ru,  
А. Л. Стемповский, д-р техн. наук, проф., акад. РАН, директор, e-mail: xxx@ippm.ru,  
Институт проблем проектирования в микроэлектронике РАН (ИППМ РАН)

## Методы повышения отказоустойчивости работы устройства управления микросистемы за счет введения структурной избыточности

*Повышение отказоустойчивости является одной из ключевых задач, возникающих при разработке вычислительных микросистем, управляющих военными и гражданскими объектами в жестких условиях космоса, высокого уровня радиации и других экстремальных воздействий. Вместе с тем, в настоящее время в основном используются архаичные методы многократного дублирования, ведущие часто к неоправданной избыточности. Данная статья посвящена методам повышения отказоустойчивости одного из ключевых компонентов микросистемы — устройства управления — за счет введения управляемой структурной избыточности. Для обеспечения необходимого уровня помехозащищенности (обнаружение ошибок с заданной степенью кратности) в качестве операции свертки временной диаграммы устройства управления предлагается использовать операцию деления на образующий многочлен в двоичном поле Галуа. Полученная контрольная свертка обеспечивает обнаружение возникающих сбоев и функциональных ошибок. Рассмотрен синтез схемы контроля, выполняющей операцию формирования контрольной сигнатуры. При этом уровень помехозащищенности определяется выбором степени образующего примитивного многочлена. В качестве примера использования предложенного подхода рассмотрена реализация микросистемы, выполняющей операцию свертки.*

**Ключевые слова:** помехоустойчивость, микросистема, устройство управления, избыточность, поля Галуа

### Введение

Повышение отказоустойчивости вычислительных систем, управляющих военными и гражданскими объектами в жестких условиях космоса, высокого уровня радиации и других экстремальных воздействий, является одной из ключевых задач, возникающих при разработке **специализированных управляющих микросистем**. Разрабатываемые в настоящее время микросистемы изготавливаются на основе современных промышленных глубоко субмикронных и нанометровых технологий (в том числе в виде "микросистем на кристалле"). По мере роста степени интеграции и уменьшения технологических размеров возрастает роль повышения надежности и отказоустойчивости проектируемых устройств под воздействием различных источников помех и сбоев.

Вместе с тем, несмотря на бурное развитие современных технологий, для обеспечения требуемой отказоустойчивости микросистем в основном используются архаичные методы многократного дублирования [1—3], ведущие часто к неоправданной избыточности. При этом основным недостатком данного подхода является неработоспособность для случая, когда ошибки "накрывают" ос-

новную и дублирующие системы. В то же время методологическим недостатком является использование "единого подхода" (метода мажорирования) для всей микросистемы без учета специфики работы ее основных компонентов (арифметического устройства, устройства управления, памяти и т. п.). Это, в свою очередь, не позволяет достичь высокого уровня надежности для случаев интенсивных помех и сбоев, возникающих при работе устройства в отмеченных выше экстремальных условиях эксплуатации. Вопросы повышения помехозащищенности приобретают еще большее значение и важность для случаев применения методов формирования активных помех в целях нарушения работы микросистемы.

Поэтому разработка концептуальных основ и методов построения микросистем, обеспечивающих достижение требуемой надежности функционирования за счет введения управляемой структурной избыточности, являются **актуальной задачей**.

Данная статья посвящена методам повышения отказоустойчивости одного из ключевых компонентов микросистемы — устройства управления (УУ), обеспечивающего формирование управляющих сигналов в соответствии с заданной последовательностью работы микросистемы. При этом в ка-

честве первого шага рассматривается подход, основанный на **обнаружении** ошибок (сбоев) в работе УУ микросистемы и последующем формировании команды на повторное выполнение последнего фрагмента работы микросистемы, в ходе выполнения которого проявился и был обнаружен сбой.

### 1. Структура микросистемы

Существуют различные подходы, обеспечивающие повышение отказоустойчивости микросистем, построенные на основе нетрадиционных структурных реализаций: модулярная арифметика [4—5], клеточные автоматы [6], специализированные устройства [7—8].

В предлагаемом варианте рассматривается повышение отказоустойчивости для микросистем с традиционной архитектурой, структурная схема которой представлена на рис. 1 и содержит следующие основные компоненты:

- арифметико-логическое устройство, обеспечивающее выполнение основных математических преобразований;
- устройство управления (УУ), обеспечивающее формирование управляющих сигналов в соответствии с заданной последовательностью работы микросистемы;
- устройство ввода-выхода для приема/передачи входных и выходных данных;
- память, обеспечивающая хранение данных;
- схема контроля, выполняющая проверку работы всех основных компонентов микросистемы.

В предлагаемом варианте повышения помехоустойчивости микросистемы УУ обеспечивает последовательное выполнение основной управляющей программы по шагам: после выполнения каждого временного отрезка  $\Delta T_j$  схема контроля обеспечивает проверку корректности работы основных компонентов системы и, в частности, УУ. Если ошибка не обнаружено, то схема контроля выдает соответствующий признак в УУ, по которому последнее переходит к выполнению следующего временного интервала  $\Delta T_{j+1}$ . Если схема контроля обнаружила наличие ошибки, то она формирует признак, по которому УУ повторяет выполнение последнего



Рис. 1. Структурная схема микросистемы

фрагмента. Ниже представлена реализация схемы контроля УУ, функционирующей в соответствии с предложенным принципом повышения помехозащищенности микросистемы.

### Методы повышения помехозащищенности устройства управления

В основе работы УУ лежит формирование временной диаграммы для управления работой остальных компонентов микросистемы. На рис. 2 представлен типовой вид временной диаграммы, имеющей (для примера) семь выходных управляющих сигналов ( $Q_1, \dots, Q_7$ ).

В общем случае управляющие выходы  $Q_{i,1}, \dots, Q_{i,K}$  поступают на управляющие входы основных компонентов микросистемы и обеспечивают ее работу по заданному алгоритму. При этом индекс  $i$  определяет значения управляющего сигнала  $Q_{i,K}$  в момент времени  $t_i$ , для которого  $t_i \in \Delta T_j$ . В процессе работы УУ и формирования временных диаграмм ( $Q_{i,1}, \dots, Q_{i,K}$ ) возможно возникновение сбоев и отказов, которые могут проявиться в неверном значении выходных сигналов в текущий момент времени  $t_i$ . Например, вместо значений  $Q_{i,1} = 1$  может появиться значение  $Q_{i,1} = 0$ , что приведет к неправильной работе микросистемы.

Основная идея предлагаемого метода защиты УУ состоит в том, чтобы последовательно "свернуть" все значения  $Q_{i,1}, \dots, Q_{i,K}$  для всех тактов работы микросистемы ( $t_1, \dots, t_i, \dots, t_n$ ) внутри заданного интервала времени  $\Delta T_j$  и получить итоговую сигнатурную свертку:  $G1, \dots, GR$ , где  $R$  — длина (число двоичных разрядов) итоговой сигнатуры. Полученная сигнатура будет сравниваться с эталонной сигнатурой ( $b1, \dots, bR$ ), которая ранее была рассчитана путем моделирования работы УУ. Различие между **полученной**  $G1, \dots, GR$  и **эталонной**  $g1, \dots, gR$  (хранящейся в памяти микросистемы) сигнатурами позволит определить наличие ошибки на выходе УУ и сформировать сигнал для повторного выполнения интервала  $\Delta T_j$ . Если сигнатуры совпали, то формируется команда для перехода к следующему временному интервалу  $\Delta T_{j+1}$ .

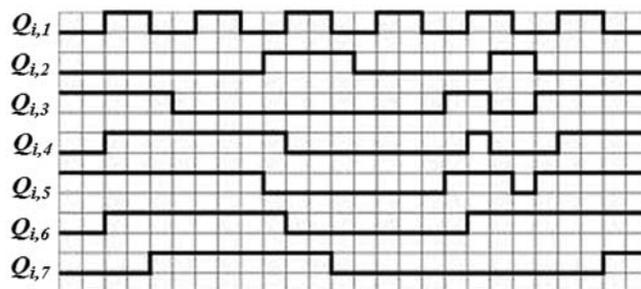


Рис. 2. Типовой вид временной диаграммы, формируемой устройством управления

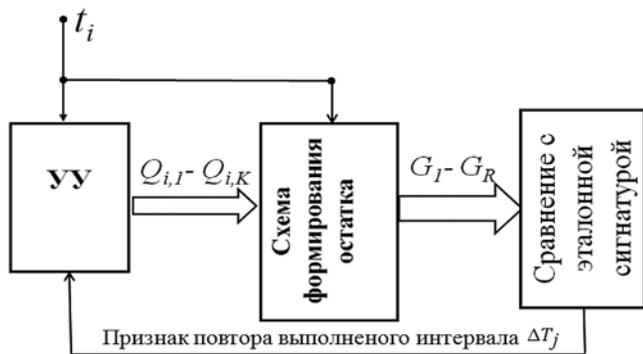


Рис. 3. Структура схемы контроля устройства управления

На рис. 3 представлена обобщенная структура схемы контроля УУ, работающей в соответствии с предлагаемым выше подходом.

Для обеспечения необходимого уровня коррекции (обнаружение ошибок с заданной степенью кратности) в качестве операции свертки предлагается использовать операцию деления на образующий многочлен ( $G(x)$ ) в двоичном поле Галуа (GF) [4]. При этом в качестве входных данных для формирования текущего значения сигнатуры  $G1_i, \dots, GR_i$  предлагается использовать:

- текущие значения  $Q_{i,1}, \dots, Q_{i,K}$ , полученные для такта  $t_i$ ;
- значения свертки  $G1_{i-1}, \dots, GR_{i-1}$ , полученной на предыдущем такте  $t_{i-1}$ .

Итоговая свертка ( $G1, \dots, GR$ ) для интервала  $\Delta T_j$  (для которого проводится оценка корректности работы УУ) будет определяться как остаток от деления двоичного полинома  $Y(x)$ , характеризующего всю последовательность  $Q_{i,1}, \dots, Q_{i,K}$  за интервал  $\Delta T_j$ ,

$$Y(x) = Q_{1,n}x^{kn-1} + Q_{2,n}x^{kn-2} + \dots + Q_{k,n}x^{kn-k} + \dots + Q_{1,i}x^{ki-1} + Q_{2,i}x^{ki-2} + \dots + Q_{k,i}x^{ik-k} + \dots + Q_{1,1}x^{k-1} + Q_{2,1}x^{k-2} + \dots + Q_{k,1}x^0$$

на образующий полином  $g_R(x)$  степени  $R$ :

$$G1, \dots, GR = \text{остаток от деления} \left[ \frac{Y(x)}{g_R(x)} \right]. \quad (1)$$

Корректирующие свойства полученной свертки (максимальная кратность обнаруживаемых ошибок) будет полностью определяться степенью и видом образующего многочлена  $g(x)$ . Согласно [9] при выборе в качестве образующего примитивного многочлена [10] степени  $R$  величина  $\gamma$ , определяемая как отношение числа обнаруживаемых ошибок к общему числу ошибок, возникающих в УУ и проявляющихся (появляющихся) хотя бы на одном из его выходов, будет определяться как

$$\gamma \geq 1 - 2^{-R}. \quad (2)$$

## Синтез схемы контроля

Рассмотрим синтез схемы контроля, выполняющей операцию формирования сигнатуры (деления на  $g_R(x)$ ). В качестве примера рассмотрим работу схемы для следующего случая:

- в каждый момент  $t_i$  УУ формирует диаграмму для пяти выходных сигналов  $Q_{i,1} - Q_{i,5}$ ;
- в качестве образующего выбран примитивный многочлен  $g_3(x) = x^3 + x + 1$ . При этом остаток от деления  $Q_{i,1} - Q_{i,5}$  на полином 3-й степени  $G(x)$  имеет три разряда и будет обозначаться как  $G1_i, G2_i, G3_i$ .

Структура устройства формирования остатка представлена ниже на рис. 4.

Устройство работает следующим образом. Схема контроля, на вход которой в момент  $t_i$  поступают:

- вектор управляющих сигналов, формируемых УУ:  $Q_{i,1} - Q_{i,5}$ ;
- значения остатка  $G1_{i-1}, G2_{i-1}, G3_{i-1}$  с предыдущего такта  $t_{i-1}$ , хранящегося в регистровой памяти, формирует значение остатка  $G1_i, G2_i, G3_i$  для текущего  $t_i$ , который записывается в регистровую память.

После выполнения всех тактов  $t_1, \dots, t_i, \dots, t_n$  внутри заданного временного интервала  $\Delta T_j$  в регистровой памяти оказывается значение остатка (сигнатуры)  $G1, G2, G3$ , которое сравнивается со значением эталонной сигнатуры  $b1, b2, b3$ , хранящейся в памяти микросистемы.

В соответствии с (2) выбор в качестве образующего примитивного многочлена 3-й степени обеспечивает обнаружение не менее 88 % всех ошибок, возникающих в УУ. Функциональная схема деления выходного вектора  $Q_{i,1} - Q_{i,5}$  на  $g_3(x) = x^3 + x + 1$  и формирования остатков  $G1, G2, G3$  для двух начальных тактов  $t_1, t_2$  показана на рис. 5 (знак  $\oplus$  обозначает суммирование по модулю 2).

Рассмотрим формирование остатка  $G1, G2, G3$  за два такта ( $t_1$  и  $t_2$ ) работы УУ. При этом УУ формирует:

- вектор  $Q_{1,1}, Q_{1,2}, Q_{1,3}, Q_{1,4}, Q_{1,5}$  на такте  $t_1$ ;
  - вектор  $Q_{2,1}, Q_{2,2}, Q_{2,3}, Q_{2,4}, Q_{2,5}$  для такта  $t_2$ .
- Таким образом, выходным управляющим вектором УУ для тактов  $t_1$  и  $t_2$  является вектор  $Y = Q_{1,1}, Q_{1,2}, Q_{1,3}, Q_{1,4}, Q_{1,5}, Q_{2,1}, Q_{2,2}, Q_{2,3}, Q_{2,4}, Q_{2,5}$ ,

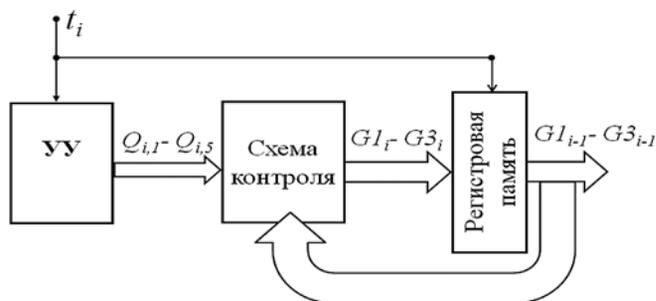


Рис. 4. Структура устройства формирования остатка

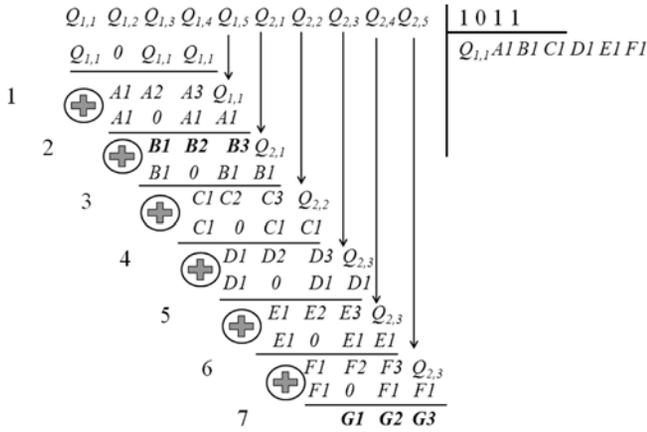


Рис. 5. Функциональная схема деления выходного вектора на образующий полином  $G(x)$

представленный в качестве исходного на рис. 5. При делении вектора  $Y$  на образующий полином формируются следующие промежуточные остатки, представленные на рис. 5:

$$\begin{aligned}
 A1, A2, A3 &\Rightarrow A1 = Q_{1,2}; \\
 A2 &= Q_{1,1} \oplus Q_{1,3}; A3 = Q_{1,1} \oplus Q_{1,4}; \\
 B1, B2, B3 &\Rightarrow B1 = Q_{1,1}; \\
 B2 &= Q_{1,1} \oplus Q_{1,2} \oplus Q_{1,4}; B3 = Q_{1,2} \oplus Q_{1,5}; \\
 C1, C2, C3 &\Rightarrow A1 = B2; \\
 C2 &= B1 \oplus B3; C3 = B1 \oplus Q_{2,1}; \\
 D1, D2, D3 &\Rightarrow D1 = B1 \oplus B3; \\
 D2 &= B1 \oplus B2 \oplus Q_{2,1}; D3 = B2 \oplus Q_{2,2}; \\
 E1, E2, E3 &\Rightarrow E1 = B1 \oplus B2 \oplus Q_{2,1}; \\
 E2 &= B1 \oplus B2 \oplus B3 \oplus Q_{2,2}; E3 = B1 \oplus B3 \oplus Q_{2,3}; \\
 F1, F2, F3 &\Rightarrow F1 = B1 \oplus B2 \oplus B3 \oplus Q_{2,2}; \\
 F2 &= B2 \oplus B3 \oplus Q_{2,1} \oplus Q_{2,3}; F3 = B1 \oplus B3 \oplus Q_{2,1} \oplus Q_{2,4}; \\
 G1, G2, G3 &\Rightarrow G1 = B2 \oplus B3 \oplus Q_{2,1} \oplus Q_{2,3}; \\
 G2 &= B3 \oplus Q_{2,1} \oplus Q_{2,2} \oplus Q_{2,4}; \\
 G3 &= B1 \oplus B2 \oplus B3 \oplus Q_{2,2} \oplus Q_{2,5}.
 \end{aligned} \quad (3)$$

Необходимо отметить, что:

- остаток для  $t_1$  обозначен как  $B1, B2, B3$ ;
- остаток для  $t_2$  обозначен как  $G1, G2, G3$ .

На рис. 5 они выделены полужирным шрифтом.

В итоге, имеем базовое выражение (3), определяющее связь между:

- промежуточным остатком ( $B1, B2, B3$ ) для  $t_1$ ;
- значениями входных векторов для  $t_2$  ( $Q_{2,1}, Q_{2,2}, Q_{2,3}, Q_{2,4}, Q_{2,5}$ );
- значением остатка для  $t_2$  ( $G1, G2, G3$ ).

Проведя следующие замены для перехода от  $t_2$  к произвольному такту  $t_i$ :

$$\begin{aligned}
 t_1 &\Rightarrow t_{i-1}; t_2 \Rightarrow t_i; Q_{2,1}, Q_{2,2}, Q_{2,3}, Q_{2,4}, Q_{2,5} \Rightarrow \\
 &\Rightarrow Q_{i,1}, Q_{i,2}, Q_{i,3}, Q_{i,4}, Q_{i,5}; \\
 B1, B2, B3 &\Rightarrow G1_{i-1}, G2_{i-1}, G3_{i-1}; \\
 G1, G2, G3 &\Rightarrow G1_i, G2_i, G3_i.
 \end{aligned}$$

получим итоговое выражение:

$$\begin{aligned}
 G1_i &= G2_{i-1} \oplus G3_{i-1} \oplus Q_{i,1} \oplus Q_{i,3}; \\
 G2_i &= G3_{i-1} \oplus Q_{i,1} \oplus Q_{i,2} \oplus Q_{i,4}; \\
 G3_i &= G1_{i-1} \oplus G2_{i-1} \oplus G3_{i-1} \oplus Q_{i,2} \oplus Q_{i,5},
 \end{aligned} \quad (4)$$

определяющее **схему контроля**, входящую в состав выделенной структуры (см. рис. 4), которая для каждого такта  $t_i$  формирует значения  $G1_i, G2_i, G3_i$ . Полученная структура схемы контроля, соответствующая выражению (4), представлена на рис. 6. При этом  $\oplus$  обозначает сумматор по модулю 2.

Для синтезированной схемы формирования остатка (см. рис. 4 и 5) рассмотрим примеры появления ошибок и механизм их обнаружения. Пусть при правильной работе УУ оно формирует следующий выходной вектор:

$$\begin{aligned}
 Q_{1,1} &= 1, Q_{1,2} = 0, Q_{1,3} = 1, Q_{1,4} = 1, Q_{1,5} = 0, \\
 Q_{2,1} &= 1, Q_{2,2} = 0, Q_{2,3} = 1, Q_{2,4} = 1, Q_{2,5} = 0.
 \end{aligned}$$

При этом значение эталонной сигнатуры будет иметь вид

$$b1 = 1, b2 = 0, b3 = 1. \quad (5)$$

Рассмотрим случай возникновения вектора ошибки

$$1, 0, 0, 0, 1, 1, 0, 0, 0, 1,$$

который искажает выходной вектор и делает его равным:

$$\begin{aligned}
 Q_{1,1} &= 0, Q_{1,2} = 0, Q_{1,3} = 1, Q_{1,4} = 1, Q_{1,5} = 1, \\
 Q_{2,1} &= 0, Q_{2,2} = 0, Q_{2,3} = 1, Q_{2,4} = 1, Q_{2,5} = 1.
 \end{aligned}$$

Тогда итоговое значение сигнатуры, сформированное схемой контроля, будет иметь вид

$$G1 = 0, G2 = 0, G3 = 0,$$

который отличается от значения эталонной сигнатуры (5). В этом случае система контроля вырабатывает признак наличия ошибки и запускает УУ для повторного выполнения данного интервала.

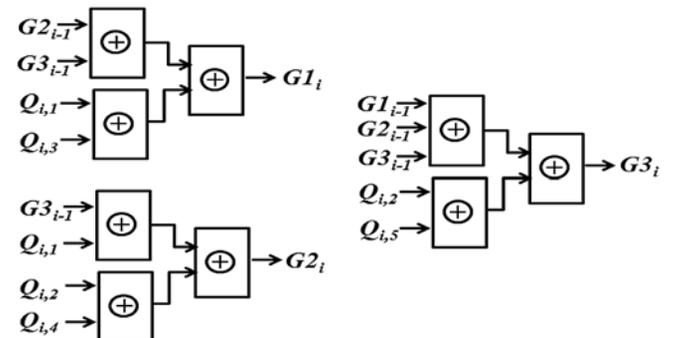


Рис. 6. Схема контроля для устройства управления

### Пример работы микросистемы со схемой защиты устройства управления

В качестве примера использования предложенного подхода рассмотрим реализацию микросистемы, выполняющей операцию свертки:

$$S(x, a) = \sum_{j=1}^m a_j x_j, \quad (6)$$

где  $(a_1, \dots, a_j, \dots, a_m)$  и  $(x_1, \dots, x_j, \dots, x_m)$  — входные векторы.

На рис. 7 представлена структура микросистемы, реализующей заданную свертку (выражение (6)).

Система работает следующим образом. Реализация свертки (6) обеспечивается за счет последовательного выполнения интервалов  $\Delta T_j$ , на каждом из которых осуществляется преобразование

$$S_{j+1} = S_j + a_j x_j. \quad (7)$$

При этом для каждого  $\Delta T_j$  УУ формирует диаграмму управляющих сигналов  $(Q_{i,1}, \dots, Q_{i,K})$ , для которых ранее были определены эталонные сигнатуры  $(b1, \dots, bR)$ , хранящиеся в памяти схемы контроля. При этом входные векторы  $(a_j, x_j)$  расположены в стековой памяти микросистемы. Для каждого интервала  $\Delta T_j$  происходит выполнение (7). Параллельно схема контроля определяет значение текущей сигнатуры  $(G1, \dots, GR)$  и сравнивает ее с эталонной. По результатам сравнения схема контроля формирует "признак корректности". При этом если ошибка не была замечена, то в стековой памяти осуществляется замена:

$$S_j = S_{j+1}; a_j = a_{j+1}; x_j = x_{j+1}.$$



Рис. 7. Структура микросистемы

Если ошибка была замечена, то УУ повторяет формирование управляющей диаграммы  $(Q_{i,1} - Q_{i,N})$  для текущего интервала  $\Delta T_j$ .

### Заключение

Статья посвящена разработке методов повышения отказоустойчивости специализированных микросистем, обеспечивающих управление военными и гражданскими объектами в экстремальных условиях эксплуатации, а также в условиях активного воздействия искусственных радиопомех. Предлагаемая методика основана на достижении требуемой надежности функционирования за счет введения структурно-информационной избыточности. В качестве базовых преобразований для определения наличия ошибки предложено использование двоичных полей Галуа. Данная статья посвящена методам повышения отказоустойчивости одного из ключевых компонентов микросистемы — устройства управления (УУ). При этом в качестве первого шага рассматривается подход, основанный на обнаружении ошибок (сбоев) в работе УУ. В статье последовательно представлены: описание предлагаемого метода повышения отказоустойчивости УУ; формальный алгоритм структурного синтеза устройства контроля; пример построения микросистемы на основе предложенного подхода.

### Список литературы

1. **Шишкевич А. А.** Оценка показателей надежности вычислительных устройств с трехкратным мажорированием при отказах и сбоях // Изв. вузов. Электроника. 2013. № 4. С. 84—88.
2. **Пат. РФ № 2047899.** Способ обеспечения отказоустойчивости вычислительных систем / В. С. Козлов, Ф. А. Листенгорт, В. А. Меркулов, В. Г. Сиренко, А. М. Смаглий, А. В. Шагин // URL: <http://ru-patent.info/20/45-49/2047899.html>
3. **Карпов С., Попович А.** Практический подход к резервированию вычислительно-управляющей системы средствами сети SpaceWire // Компоненты и технологии. 2010. № 4. С. 77—80.
4. **Амербаев В. М., Стемповский А. Л., Соловьев Р. А.** Параллельные вычисления в кольце гауссовых чисел над полем Галуа GF(p) // Проблемы разработки перспективных микро- и наноэлектронных систем — 2012. Сб. научн. тр. / Под общ. ред. А. Л. Стемповского. М.: ИППМ РАН. 2012. С. 517—520.
5. **Stempkovskiy A., Amerbaev V., Solovyev R., Telpuhov D.** Efficient Calculation of Cyclic Convolution by Means of Fast Fourier Transform in a Finite Field // Proc. of 11th EAST-WEST DESIGN & TEST SYMPOSIUM (EWDTS 2013). 2013.
6. **Стемповский А. Л.** Отказоустойчивые архитектуры микросистемных вычислительных систем // Информационные технологии и вычислительные системы. 2001. Вып. 2/3. С. 40—50.
7. **А. с. № 276219.** Вычислительное устройство / Л. Н. Преснухин, В. А. Бархоткин, А. Н. Соловьев.
8. **А. с. № 1080828.** Специализированное вычислительное устройство / Л. Н. Преснухин, В. А. Бархоткин, А. Н. Соловьев.
9. **Касперский К.** Полиномиальная арифметика и поля Галуа или информация, воскресшая из пепла // Системный администратор. 2003. № 10. С. 84—90.
10. **Блейхут Р.** Теория и практика кодов, контролируемых ошибок. М.: Мир, 1986.

## Methods to Improve Noise Tolerance of the Control Equipment Microsystems by Structural Redundancy Introduction

Noise tolerance increase is one of the key problems arising in the development of computing microsystems control military and civilian objects under harsh conditions of space, high levels of radiation and other extreme effects. However, at present, mainly used archaic methods of multiple overlapping, often leading to unjustified redundancy. This article focuses on how to improve noise tolerance of a key component of the microsystem — device management — through the introduction of controlled structural redundancy. To ensure the necessary level of noise tolerance (error detection with a given degree of multiplicity) as a convolution operation timing chart control device is proposed to use the operation of division by the generator polynomial in binary Galois field. The resulting control convolution provides detection and functional failures occurring errors. The article describes the synthesis of control circuitry that performs an operation of forming a control signature. At the same level of noise tolerance determined by the choice of degree forming primitive polynomial. As an example of implementation of the proposed approach is considered a microsystem, performing the convolution operation.

**Keywords:** noise tolerance, microsystem, control unit, redundancy, Galois field

### References

1. **Shishkevich A. A.** Ocenka pokazatelej nadezhnosti vychislitel'nyh ustrojstv s mazhrirovaniem pri otkazah i sbojah. *Izv. VUZov. Jelektromka*. 2013. N. 4. P. 84—88.
2. **Pat RF № 2047899** Sposob obespechenija otkazoustojchivosti vychislitel'nyh sistem / V. S. Kozlov, F. A. Listengort, V. A. Merkulov, V. G. Sirenko, A. M. Smaglij, A. V. Shhagin. URL: <http://ru-patent.info/20/45-49/2047899.html>
3. **Karpov S., Popovich A.** Prakticheskij podhod k rezervirovaniju vychislitel'no-upravljajushhej sistemysredstvami seti SpaceWire. *Komponenty i tehnologii*. 2010. N. 4. P. 77—80.
4. **Amirbaev V. M., Stempkovskij A. L., Solov'ev R. A.** Parallelnye vychislenija v kol'ce gaussovyh chisel nad polem Galua GF(p). *Problemy razrabotki perspektivnyh mikro- i nanojelektronnyh sistem* — 2012. Sb. nauchn. tr. / Pod obshh. red. A. L. Stempkovskogo. M.: IPPM RAN. 2012. P. 517—520.
5. **Stempkovskij A., Amerbaev V., Solovyev R., Telpuhov D.** Efficient Calculation of Cyclic Convolution by Means of Fast Fourier Transform in a Finite Field. *Proc. of 11th EAST-WEST DESIGN & TEST SYMPOSIUM (EWDTS 2013)*. 2013.
6. **Stempkovskij A. L.** Otkazoustojchivye arhitektury mikrojelektronnyh vychislitel'nyh sistem. *Informacionnye tehnologii i vychislitel'nye sistemy*. 2001. Vyp. 2/3. P. 40—50.
7. **A. s. № 276219.** *Vychislitel'noe ustrojstvo*. L. N. Presnuhin, V. A. Barhotkin, A. N. Solov'ev.
8. **A. c. № 1080828.** *Specializirovannoe vychislitel'noe ustrojstvo*. L. N. Prenuhin, V. A. Barhotkin, A. N. Solov'ev.
9. **Kasperskij K.** Polinomial'naja arifmetika i polja Galua ili informacija, voskresshajaja iz pepla. *Sistemnyj administrator*. 2003. N. 10. P. 84—90.
10. **Blejhut R.** *Teorija i praktika kodov, kontrolirujushhih oshibki*. M.: Mir, 1986.

УДК 004.272.2

**В. М. Амербаев**, акад. НАН РК, д-р техн. наук, проф., гл. науч. сотр.,  
**Р. А. Соловьев**, канд. техн. наук, нач. отдела, e-mail: ZF-Turbo@yandex.ru,  
**Д. В. Тельпухов**, канд. техн. наук, науч. сотр.,  
Институт проблем проектирования в микроэлектронике РАН

## Метод вычисления циклической свертки на базе БПФ с использованием чисел Прота

В конечном поле может быть определено преобразование, аналогичное дискретному преобразованию Фурье, которое в дальнейшем может быть эффективно реализовано с помощью быстрых алгоритмов. Это преобразование можно использовать для вычисления свертки длинных последовательностей целых чисел методами целочисленной арифметики. Рассмотрен метод реализации операции свертки двух векторов с помощью аппарата модулярной арифметики над конечным полем по модулям типа Прота. Реализовано устройство, выполняющее операцию циклической свертки предложенным методом. Проведено сравнение с аналогичными устройствами, построенными на двоичной арифметике.

**Ключевые слова:** дискретное преобразование Фурье, система остаточных классов, числа Прота, циклическая свертка, конечное поле

## Введение

Свертка — это базовая операция в задачах цифровой обработки сигналов, используемая, в частности, в задачах цифровой фильтрации [1]. Высокие требования к пропускной способности для обработки данных в реальном времени определяют необходимость разработки более эффективных методов вычислений базовых функций, среди которых особое место занимает циклическая свертка. Необходимость эффективной реализации данной процедуры обусловлена ее активным использованием в целом ряде современных мобильных систем, в частности, в навигационных системах для реализации одной из ключевых вычислительных процедур — разрешения фазовых неоднозначностей [2]. Зачастую требования к быстродействию бывают настолько высоки, что некоторые наиболее емкие в вычислительном аспекте операции выносятся за пределы микроконтроллера и реализуют аппаратно.

В дискретном случае различают свертки двух видов: линейную и циклическую. Рассмотрим линейную свертку. Пусть имеется два дискретных сигнала  $a(n)$ ,  $n = 0, \dots, N - 1$ , и  $b(n)$ ,  $n = 0, \dots, M - 1$ . В общем случае длины этих сигналов  $N$  и  $M$  могут различаться. Линейной сверткой сигналов  $a(n)$  и  $b(n)$  называется дискретный сигнал вида

$$s(n) = a * b = \sum_{m=0}^n a(m)b(n-m), \\ n = 0, \dots, N + M - 2. \quad (1)$$

Для вычисления линейной свертки сигналы  $a(n)$  и  $b(n)$  сдвигают относительно друг друга, почленно перемножают и складывают. При этом предполагается, что  $a(n) = 0$  при  $n < 0$  и  $n > N$ ,  $b(n) = 0$  при  $n < 0$  и  $n > M$ .

В случае циклической свертки предполагается, что дискретные сигналы  $a(n)$  и  $b(n)$  — периодические с одинаковым периодом  $N$  отсчетов. Тогда круговой сверткой сигналов  $a(n)$  и  $b(n)$  называется сигнал вида

$$s(n) = \sum_{m=0}^{N-1} a(m)b(n-m), \quad n = 0, \dots, N - 1. \quad (2)$$

Результат циклической свертки также имеет длину  $N$  отсчетов. Используя циклическую свертку, можно рассчитать линейную свертку двух сигналов. Для этого необходимо каждый из сигналов  $a(n)$  и  $b(n)$  длительностью  $M$  и  $N$  отсчетов соответственно дополнить нулями до длины  $M + N - 1$ . Как видно из (1) и (2), для вычисления линейной свертки требуется  $MN$  умножений, а при вычислении линейной свертки через циклическую —  $(N + M - 1)^2$  умножений и сложений. И хотя на первый взгляд вычисление линейной свертки через циклическую кажется нерациональным, на практике чаще бывает удобно вычислять именно циклическую свертку

благодаря так называемой теореме о свертке [3, 4]. Суть теоремы заключается в следующем: спектр циклической свертки есть произведение спектров сворачиваемых сигналов:

$$S(k) = A(k) * B(k),$$

где  $A(k)$  и  $B(k)$  — спектры сворачиваемых сигналов,  $S(k)$  — спектр циклической свертки двух сигналов.

На практике пользуются этим методом, так как применение быстрых алгоритмов для получения спектра сигналов обеспечивает существенное сокращение вычислительных операций при вычислении как циклических, так и линейных сверток. Однако этот метод имеет ряд недостатков. Во-первых, поворачивающие коэффициенты, участвующие в преобразовании Фурье, принадлежат к области вещественных чисел и представление их в двоичной форме приводит к накоплению ошибок представления. Кроме того, для поддержания высокого быстродействия необходимо обеспечить округление промежуточных результатов, что приводит к ошибкам округления. Таким образом, данный метод приводит к существенной потере точности, а быстродействие, тем не менее, продолжает оставаться узким местом для разрабатываемых микроэлектронных устройств.

## 1. Быстрое преобразование Фурье в конечном поле

Для преодоления обозначенных недостатков предлагается использовать аппарат модулярной арифметики. Суть предлагаемого подхода заключается в том, чтобы использовать переход в спектральную область с помощью теоретико-числового преобразования Фурье и затем применить теорему о свертке. Базисом для этого преобразования служат так называемые числа Прота.

Пусть  $F$  — конечное поле Галуа  $GF(p)$  и  $d$  — делитель  $p - 1$  (возможно  $d = p - 1$ ), тогда  $\gamma$  является первообразным корнем из единицы степени  $d$  в мультипликативной группе  $F^*$  ненулевых элементов поля, если  $\gamma \in F$ ,  $\gamma^d = 1$  и  $\gamma^l \neq 1$  для любого натурального  $l < d$ .

Первообразный корень степени  $d$  из единицы существует, так как по определению мультипликативная группа  $F^*$  является циклической порядка  $(p - 1)$  [5].

В теории чисел известны так называемые числа Прота, которые представляют собой простые числа вида  $q \cdot 2^t + 1$ , где  $q$  является нечетным положительным целым числом, удовлетворяющим условию  $q < 2^t$ . В дальнейших рассуждениях будем использовать обобщение чисел Прота. Простые числа вида  $q \cdot 2^t + 1$ , где не обязательно выполнено условие  $q < 2^t$ , будем называть простыми числами типа Прота.

Поскольку любое простое  $p$  представимо в виде  $p = q \cdot 2^t + 1$  ( $t \geq 1$ ), то в качестве делителя  $(p - 1)$  можно выбрать  $d = 2^t$ . Тогда элемент  $\gamma = |w^q|_p$ , где

$w$  — примитивный элемент поля  $GF(p)$ , порождает циклическую группу корней из единицы порядка  $2^t$ .

Преобразование последовательности чисел  $(a_n)$  ( $0 \leq n \leq 2^t - 1$ ), являющихся элементами  $F$ , в последовательность чисел

$$A(k) = \left| \sum_{n=0}^{2^t-1} a_n \gamma^{nk} \right|_p \quad (3)$$

принято называть дискретным преобразованием Фурье (ДПФ) над конечным полем  $GF(p)$ .

Преобразование (3) является аналогом дискретного преобразования Фурье тригонометрического ба-

зиса, в котором поворачивающие множители  $e^{ij\frac{2\pi}{N}}$  заменяются на  $\gamma^{ij}$ , а операции умножения и сложения комплексных чисел — на соответствующие операции в  $GF(p)$ .

Поскольку коэффициенты преобразования (3) имеют те же свойства симметрии и периодичности, что и коэффициенты ДПФ тригонометрического базиса, а также число отсчетов равно степени двойки, то для вычисления значения ДПФ можно применить алгоритмы быстрых преобразований Фурье (БПФ), в частности алгоритм Кули—Тьюки [6]. Покажем, что вычисление циклической свертки в поле  $GF(p)$  может быть реализовано с помощью ДПФ в этом поле:

$$S(k) = \sum_{n=0}^{2^t-1} s(n) \gamma^{nk}, \quad k = 0 \dots 2^t - 1. \quad (4)$$

Подставив выражение для циклической свертки, получим:

$$S(k) = \sum_{n=0}^{2^t-1} \left[ \sum_{m=0}^{2^t-1} a(m) b(n-m) \right] \gamma^{nk}.$$

Поменяем порядок суммирования:

$$S(k) = \sum_{m=0}^{2^t-1} a(m) \left[ \sum_{n=0}^{2^t-1} b(n-m) \gamma^{nk} \right]. \quad (5)$$

Представим коэффициент  $\gamma^{nk}$  в виде

$$\gamma^{nk} = \gamma^{(n-m+m)k} = \gamma^{(n-m)k} \gamma^{mk}. \quad (6)$$

Подставив (6) в (5), получим:

$$S(k) = \sum_{m=0}^{2^t-1} a(m) \left[ \sum_{n=0}^{2^t-1} b(n-m) \gamma^{(n-m)k} \gamma^{mk} \right] = \left[ \sum_{m=0}^{2^t-1} a(m) \gamma^{mk} \right] \left[ \sum_{n=0}^{2^t-1} b(n-m) \gamma^{(n-m)k} \right] = A(k) B(k).$$

Таким образом, теоретико-числовой спектр циклической свертки есть произведение теорети-

ко-числовых спектров сворачиваемых сигналов, и для ее вычисления может быть применен алгоритм быстрого преобразования Фурье (БПФ).

## 2. Численный пример вычисления циклической свертки с помощью БПФ в конечном поле

Рассмотрим детальный пример, который покажет, как вычислить циклическую свертку через произведение спектров сворачиваемых сигналов. Положим  $p = 13(3 \cdot 2^2 + 1)$ ,  $t = 2$ ,  $q = 3$ , требуется вычислить свертку двух сигналов:

$$a_k = \{3, 0, 1, 2\};$$

$$b_k = \{4, 3, 5, 1\}.$$

### 2.1. Предварительные вычисления

Известно [7], что первообразный корень для  $p = 13$  равен  $w = 2$ . Тогда  $\gamma = |w^q|_p = |2^3|_{13} = 8$ . В последующих вычислениях нам также потребуется инверсный вычет для  $2^2$  по модулю  $p$ :  $(2^t)^{-1}|_p$ .

Таким образом,  $(2^t)^{-1}|_p = (4)^{-1}|_{13} = 10$ , так как  $|4 \cdot 10|_{13} = |40|_{13} = 1$ .

### 2.2. Поиск ДПФ для входных сигналов

Найдем теоретико-числовое ДПФ для входных векторов  $a_k$  и  $b_k$ :

$$\begin{aligned} A(0) &= \|a_0 \gamma^0|_{13} + |a_1 \gamma^0|_{13} + |a_2 \gamma^0|_{13} + |a_3 \gamma^0|_{13}|_{13} = \\ &= \|3 \cdot 1|_{13} + |0 \cdot 1|_{13} + |1 \cdot 1|_{13} + |2 \cdot 1|_{13}|_{13} = \\ &= |3 + 0 + 1 + 2|_{13} = |6|_{13} = 6. \end{aligned}$$

Аналогично:

$$A(1) = \|a_0 \gamma^0|_{13} + |a_1 \gamma^1|_{13} + |a_2 \gamma^2|_{13} + |a_3 \gamma^3|_{13}|_{13} = 12;$$

$$A(2) = \|a_0 \gamma^0|_{13} + |a_1 \gamma^2|_{13} + |a_2 \gamma^4|_{13} + |a_3 \gamma^6|_{13}|_{13} = 2;$$

$$A(3) = \|a_0 \gamma^0|_{13} + |a_1 \gamma^3|_{13} + |a_2 \gamma^6|_{13} + |a_3 \gamma^9|_{13}|_{13} = 5.$$

Расчет ведется по формуле (3) с подстановкой элементов вектора, значения  $\gamma$  и соответствующих степеней. Затем берется вычет по модулю 13. Аналогично считается спектр для  $B(k)$ . После вычисления ДПФ имеем:

$$A(k) = \{6, 12, 2, 5\}; \quad B(k) = \{0, 2, 5, 9\}.$$

### 2.3. Поэлементное умножение

Совершив покомпонентное умножение двух векторов  $A$  и  $B$  по модулю 13, получаем вектор свертки в частотной области. Обозначим его как  $S^*$ .

$$S^*(0) = |A(0) \cdot B(0)|_{13} = |6 \cdot 0|_{13} = 0;$$

$$S^*(1) = |A(1) \cdot B(1)|_{13} = |12 \cdot 2|_{13} = 11;$$

$$S^*(2) = |A(2) \cdot B(2)|_{13} = |2 \cdot 5|_{13} = 10;$$

$$S^*(3) = |A(3) \cdot B(3)|_{13} = |5 \cdot 9|_{13} = 6.$$

Таким образом, в результате умножения получаем спектр циклической свертки  $S^*(k) = \{0, 11, 10, 6\}$ . Для того чтобы получить циклическую свертку во временной области, обозначаемую как  $s^*(k)$ , необходимо к полученному вектору применить обратное ДПФ.

#### 2.4. Обратное дискретное преобразование Фурье вектора $S^*(k)$

Для расчета обратного ДПФ используется формула

$$s^*(k) = \left| \left| (2^t)^{-1} \right|_p \left| \sum_{n=0}^{2^t-1} S^*(n) \gamma^{-nk} \right|_p \right|_p, \\ k = 0 \dots 2^t - 1.$$

Заметим, что в этой формуле, в отличие от (3), используются коэффициенты  $(2^t)^{-1}_p$  и отрицательная степень  $\gamma$  по модулю  $2^t$ .

Рассмотрим процесс вычисления  $s^*(1)$ , значение остальных элементов вектора ищется аналогично. Выше мы уже нашли значение  $(2^t)^{-1}_p = 10$ , которое используется в вычислениях:

$$s^*(1) = \left| 10 \cdot \left| 0 \cdot 8^{-0 \cdot 114} \right|_{13} + \left| 11 \cdot 8^{-1 \cdot 114} \right|_{13} + \right. \\ \left. + \left| 10 \cdot 8^{-2 \cdot 114} \right|_{13} + \left| 6 \cdot 8^{-3 \cdot 114} \right|_{13} \right|_{13} = 7.$$

Полностью искомым вектор выглядит так:

$$s^*(k) = \{10, 7, 8, 1\}.$$

#### 2.5. Верификация

Рассчитаем циклическую свертку классическим способом по формуле (2), чтобы проверить полученный результат:

$$s(0) = |a_0 b_0 + a_1 b_3 + a_2 b_2 + a_3 b_1|_{13} = 10;$$

$$s(1) = |a_0 b_1 + a_1 b_0 + a_2 b_3 + a_3 b_2|_{13} = 7;$$

$$s(2) = |a_0 b_2 + a_1 b_1 + a_2 b_0 + a_3 b_3|_{13} = 8;$$

$$s(3) = |a_0 b_3 + a_1 b_2 + a_2 b_1 + a_3 b_0|_{13} = 1.$$

Как видно,  $(s^*(k) = s(k))$ , что совпадает с утверждением о том, что теоретико-числовой спектр циклической свертки есть произведение теоретико-числовых спектров сворачиваемых сигналов. Для вычисления здесь использовалось стандартное ДПФ, однако на практике эффективнее использовать БПФ.

### 3. Вычисление циклической свертки в модулярной арифметике по составному модулю $P = p_1 \cdot p_2 \cdot \dots \cdot p_n$

Преыдушие рассуждения были связаны с вычислениями в конечном поле, в то время как следующий шаг будет связан с модулярной арифметикой.

Модулярная арифметика определяется набором  $N$  целых попарно взаимно простых чисел  $\{p_1, p_2, \dots, p_N\}$ ,

называемых модулями. Произведение этих чисел  $M = p_1 \cdot p_2 \cdot \dots \cdot p_N$  определяет динамический диапазон системы. Это означает, что любое число в пределах этого диапазона  $[0 \dots M)$  имеет его уникальное представление в модулярной арифметике, т. е. любое произвольное целое число  $X$  меньше, чем  $M$ , может быть представлено в модулярной арифметике набором  $\{x_1, x_2, x_3, \dots, x_N\}$ , где  $x_i = |X|_{p_i}$ .

При условии, что результат выполнения операции не выходит за выбранный диапазон  $M$ , арифметические операции сложения, вычитания и умножения выполняются над вычетами параллельно, т. е. независимо по каждому модульному основанию:

$$|X + Y|_M = \{|x_1 + y_1|_{p_1}, |x_2 + y_2|_{p_2}, \dots, |x_N + y_N|_{p_N}\}.$$

Для реализации модулярного вычислителя циклической свертки размерности  $2^t$  необходимо выбирать простые числа  $p_i$  такими, чтобы среди делителей  $(p_i - 1)$  присутствовал  $d_i = 2^t$ . Такие числа Прота будем называть однотипными. Это требование обеспечивает возможность реализации теоретико-числового БПФ по составному модулю  $P$ . Этот модуль требуется выбирать таким, чтобы полная система вычетов  $\mathbb{Z}_P$  покрывала динамический диапазон значений свертки.

Преимущества данного подхода основываются на параллельной организации модулярных структур, а также на целочисленной природе теоретико-числового преобразования Фурье. Вкупе два этих свойства позволяют строить аппаратные вычислители циклической свертки с улучшенными показателями быстродействия и точности вычислений.

Для проверки эффективности предлагаемых методов было реализовано устройство для выполнения операции свертки двух сигналов методами модулярной арифметики. Высокий уровень параллелизма вычислений достигнут благодаря конвейерному исполнению всех стадий обработки информации (рис. 1):

- использованы систолические прямые преобразователи, описанные в работах [8, 11];
- прямые и обратные преобразователи Фурье внутри модульных каналов реализованы по конвейерной схеме Radix-2 Single-path Delay Feedback [9]. Причем прямой БПФ построен по алгоритму с прореживанием по частоте, в то время как обратный БПФ прореживается по времени. Это позволяет строить весь тракт обработки без генератора адреса данных для бит-реверсивного перераспределения потока информации;
- стадии обработки данных (БПФ, умножение, обратное БПФ) по каждому модульному каналу разделены регистрами и оформлены в отдельные ступени конвейера;
- обратные преобразователи реализованы на базе перевода в полиадический код, что также обеспечивает высокий уровень параллелизма вычислений [10, 11].

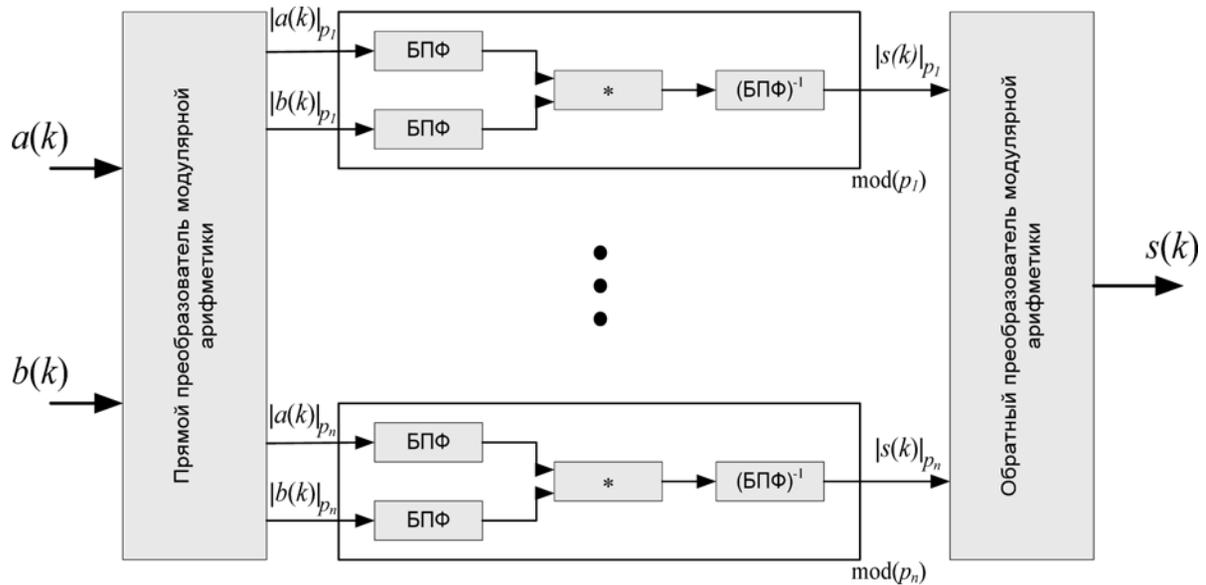


Рис. 1. Вычисление циклической свертки двух сигналов

Также для нахождения наиболее эффективного базиса оснований, необходимого для корректного вычисления модулярной свертки, было реализовано программное средство [12].

#### 4. Экспериментальные результаты

В процессе исследования было обнаружено, что конвейерные прямые и обратные преобразователи обладают меньшей задержкой на критическом пути, чем модульные каналы циклической свертки. Таким образом, самым медленным участком всего конвейерного тракта является канал вычисления свертки по самому большому модулю, и тактовая частота этого участка определяет тактовую частоту работы всего устройства. Учитывая данный факт, экспериментальные оценки производительности модулярных сверток были получены с помощью синтеза

только части устройства — модульного канала самой большой разрядности.

Для оценки эффективности предлагаемого метода были также спроектированы устройства, реализующие свертку двух сигналов в традиционной двоичной арифметике с помощью аналогичных устройств БПФ.

Для построения модели устройства выбран маршрут проектирования цифровых интегральных схем (ИС) на основе библиотек стандартных ячеек. В маршруте используется:

- поведенческое описание устройства на языке Verilog HDL;
- средства логического синтеза Synopsys Design Compiler;
- средства статического временного анализа Synopsys Prime Time;
- библиотека стандартных ячеек Nangate Open Cell Library с проектными нормами 45 нм.

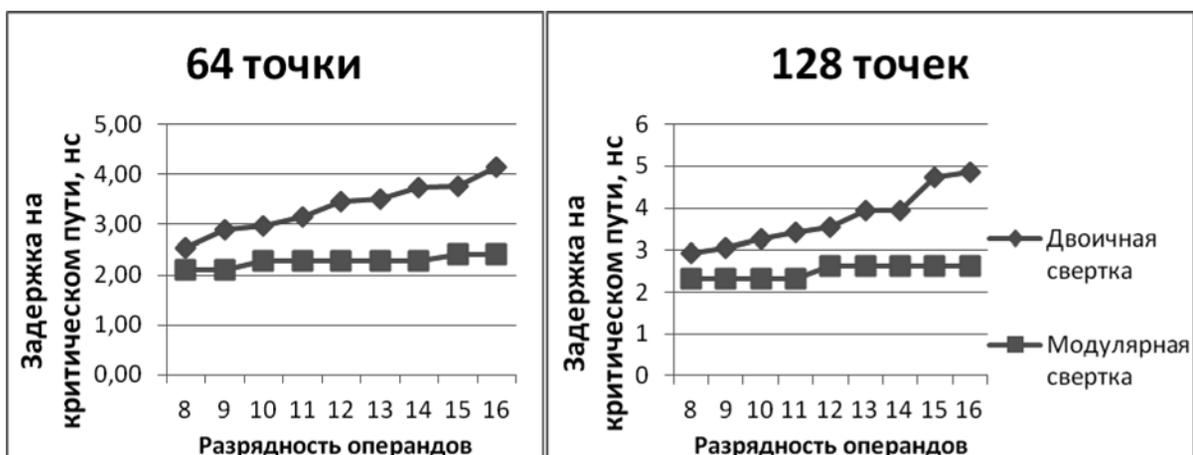


Рис. 2. Экспериментальные оценки производительности двоичных и модулярных сверток

Полученные оценки производительности представлены на рис. 2.

Рис. 2 показывает, что использование предлагаемого метода теоретико-числового быстрого преобразования Фурье по однотипным числам Прота, вкпе с разработанными схемами прямых и обратных преобразователей, позволяет строить устройства вычисления циклической свертки, демонстрирующие 10–50 %-ное увеличение производительности по сравнению с двоичными аналогами.

#### Список литературы

1. **Smith S. W.** *The Scientist & Engineer's Guide to Digital Signal Processing*. California: California Technical Pub, 1997, 626 p.
2. **Алексеев В. Е., Соловьев А. Н.** Определение вектора многоантенной GPS системы на основе процедуры разрешения фазовых неоднозначностей // Изв. вузов. ЭЛЕКТРОНИКА. 2011. № 5 (91). С. 75–82.
3. <http://mathworld.wolfram.com/ConvolutionTheorem.html>

4. **Bracewell R. N.** *The Fourier Transform and its Applications*. Columbus: McGraw-Hill Higher Education, 1986. 624 p.
5. **Adamson I. T.** *Introduction to Field Theory*. London: Oliver & Boyd, 1964. 181 p.
6. **Cooley J. W., Tukey J. W.** An Algorithm for the Machine Computation of Complex Fourier Series // *Mathematics Computation*. 1965. V. 19. P. 297–301.
7. **Виноградов И. М.** *Основы теории чисел*. М.-Л.: Гостехиздат, 1952. 180 с.
8. **Амербаев В. М., Балака Е. С., Константинов А. В., Тельпухов Д. В.** Методы построения прямых преобразователей модулярной логарифметики, ориентированных на ЦОС // Тр. IV Всероссийской научно-техн. конф. "Проблемы разработки перспективных микро- и нанoeлектронных систем — 2010". Под общ. ред. А. Л. Стемповского. М.: ИППМ РАН, 2010. С. 374–377.
9. **Wold E. H., Despain A. M.** Pipeline and parallel-pipeline FFT processors for VLSI implementation // *IEEE Trans. Comput.* 1984. V. 33 (5). P. 414–426.
10. **Тельпухов Д. В.** Построение обратных преобразователей модулярной логарифметики для устройств цифровой обработки сигналов // *Информационные технологии*. 2011. № 4. С. 60–64.
11. **Omondi A., Premkumar B.** *Residue Number System: Theory and Implementation*. London: Imperial College Press, 2007. 296 p.
12. <http://vscripts.ru/2012/prime-set-generator.php>

**V. M. Amerbaev**, Chief Researcher, **R. A. Solovyev**, Head of Department, **D. V. Telpukhov**, Research Associate, e-mail: [nofrost@inbox.ru](mailto:nofrost@inbox.ru), Institute for Design Problems in Microelectronics

## Cyclic Convolution Calculation Method Based on FFT Using Proth Numbers

*Over a finite field a transformation similar to discrete Fourier transform can be defined that can be efficiently implemented using fast algorithms. This transformation can be used to calculate different types of convolutions of long integer sequences in finite fields. Cyclic convolution is often used for calculating regular convolution via so-called convolution theorem. Residue number system can serve as a basis for combining multiple finite field convolutions into full-scale integer convolution. In this paper a method of computing of two vectors convolution by means of residue number system arithmetic over Proth-type modulo finite field is considered. The device performing the operation of cyclic convolution based on the proposed method was implemented. Comparison to binary analogues was carried out and showed a 10–50 % growth of performance for the range of input data from 8 to 16 bit.*

**Keywords:** Discrete Fourier transform, residue number system, Proth numbers, cyclic convolution, finite field

#### References

1. **Smith S. W.** *The Scientist & Engineer's Guide to Digital Signal Processing*. California: California Technical Pub, 1997. 626 p.
2. **Alekseev V. E., Solov'ev A. N.** Определение вектора многоантенной GPS системы на основе процедуры разрешения фазовых неоднозначностей. *Izv. vuzov. JELETRONIKA*. 2011. N. 5 (91). P. 75–82.
3. <http://mathworld.wolfram.com/ConvolutionTheorem.html>
4. **Bracewell R. N.** *The Fourier Transform and its Applications*. Columbus: McGraw-Hill Higher Education, 1986. 624 p.
5. **Adamson I. T.** *Introduction to Field Theory*. London: Oliver & Boyd, 1964, 181 p.
6. **Cooley J. W., Tukey J. W.** An Algorithm for the Machine Computation of Complex Fourier Series. *Mathematics Computation*. 1965. V. 19. P. 297–301.
7. **Vinogradov I. M.** *Osnovy teorii chisel*. М.-Л.: Gostehizdat, 1952. 180 p.

8. **Amerbaev V. M., Balaka E. S., Konstantinov A. V., Tel'puhov D. V.** Metody postroeniya prjamyh preobrazovatelej moduljarnoj logarifmetiki, orientirovannyh na COS. *Tr. IV Vserossijskoj nauchno-tehn. konf. "Problemy razrabotki perspektivnyh mikro- i nanojelektronnyh sistem — 2010"*. Pod obshh. red. A. L. Stempkovskogo). М.: ИППМ РАН, 2010. P. 374–377.
9. **Wold E. H., Despain A. M.** Pipeline and parallel-pipeline FFT processors for VLSI implementation. *IEEE Trans. Comput.* 1984. V. 33 (5). P. 414–426.
10. **Tel'puhov D. V.** Postroenie obratnyh preobrazovatelej moduljarnoj logarifmetiki dlja ustrojstv cifrovoj obrabotki signalov. *Infor-macionnye tehnologii*. 2011. N. 4. P. 60–64.
11. **Omondi A., Premkumar B.** *Residue Number System: Theory and Implementation*. London: Imperial College Press, 2007. 296 p.
12. <http://vscripts.ru/2012/prime-set-generator.php>

# МОДЕЛИРОВАНИЕ И ОПТИМИЗАЦИЯ MODELING AND OPTIMIZATION

УДК 519.873:519.218.4:004.056.53

**А. И. Перегуда**, д-р техн. наук, проф., e-mail: pereguda@iate.obninsk.ru,  
Обнинский институт атомной энергетики — филиал федерального государственного  
автономного образовательного учреждения высшего профессионального образования  
"Национальный исследовательский ядерный университет "МИФИ"

## Математическая модель надежности информационной системы, состоящей из объекта защиты и $N$ систем безопасности

*Рассмотрена математическая модель надежности информационной системы, состоящей из объекта защиты информации и  $N$  систем безопасности, парирующих возможные попытки несанкционированного доступа к информации. Математической моделью надежности изучаемой системы является суперпозиция случайных процессов восстановления и регенерации. Анализ математической модели надежности информационной системы позволяет вычислить математическое ожидание случайного времени до осуществления несанкционированного доступа, а также и асимптотическую оценку вероятности несанкционированного доступа к защищаемой информации до момента времени  $t$ .*

**Ключевые слова:** надежность, система безопасности, случайные величины, наработка до отказа, случайный процесс, математическое ожидание времени, функция распределения

### Введение

Использование различных автоматизированных систем, построенных на базе вычислительной техники, сделало актуальной проблему обеспечения информационной безопасности. Для решения задачи обеспечения защиты информации от несанкционированного доступа требуется обеспечить определенный уровень надежности таких систем, включая надежность их аппаратных средств и программного обеспечения. Эта задача рассматривалась в ряде работ [1–3]. При анализе надежности таких систем недостаточно просто учитывать структуру каждой системы безопасности в отдельности и особенностей их восстановления после отказа, но и возможность попыток несанкционированного доступа к информации, что позволит более обоснованно сформулировать вывод о надежности информационной системы. Математические модели надежности систем защиты информации приводятся в работах, например в [4, 5]. Здесь рассматривается модель надежности информационной системы, состоящей из объекта защиты и  $N$  систем безопасности.

### Постановка задачи

Пусть информационная система состоит из объекта защиты и  $N$  систем безопасности. Каждая из систем безопасности парирует только один вид

атак на объект защиты. Системы безопасности не являются абсолютно надежными и могут оказаться в неработоспособном состоянии. После отказа системы безопасности выполняется ее полное восстановление к исходному состоянию. Предполагается, что отказы систем безопасности являются независимыми. Системы безопасности имеют сложную структуру:  $j$ -я система безопасности включает  $M_j$  подсистем, а  $k$ -я подсистема  $j$ -й системы безопасности состоит из  $C_{j,k}$  элементов.

Обозначим  $\chi_{i,j}$ ,  $i = 1, 2, \dots, j = 1, 2, \dots, N$ , случайное время до  $i$ -й атаки на объект защиты, обнаруживаемой  $j$ -й системой безопасности. Пусть  $\chi_{i,j}$ ,  $i = 1, 2, \dots, j = 1, 2, \dots, N$ , — независимые случайные величины, причем для каждого фиксированного  $j$  величины  $\chi_{i,j}$ ,  $i = 1, 2, \dots$ , распределены одинаково с функцией распределения  $F_{\chi_j}(t)$ . Случайное время отражения  $i$ -й атаки на объект защиты, обнаруженной  $j$ -й системой безопасности, обозначим  $\gamma_{i,j}$ ,  $i = 1, 2, \dots, j = 1, 2, \dots, N$ . При этом считаем, что  $\gamma_{i,j}$ ,  $i = 1, 2, \dots, j = 1, 2, \dots, N$ , — это независимые случайные величины, и для каждого фиксированного  $j$  величины  $\gamma_{i,j}$ ,  $i = 1, 2, \dots$  имеют одну и ту же функцию распределения  $F_{\gamma_j}(t)$ . Предполагаем, что моменты завершения отражения атаки являются моментами регенерации процесса функционирования системы

защиты информации. На функции распределения  $F_{\chi_j}(t)$  и  $F_{\gamma_j}(t)$  накладываются дополнительные ограничения, а именно, предполагается, что соответствующие распределения нерешетчатые и имеют конечные математические ожидания.

Пусть  $\xi_{i,j,k,l}$ ,  $i = 1, 2, \dots, j = 1, 2, \dots, N, k = 1, 2, \dots, M_j, l = 1, 2, \dots, C_{j,k}$ , обозначает случайную наработку до  $i$ -го отказа  $l$ -го элемента  $k$ -й подсистемы  $j$ -й системы безопасности. Предположим, что  $\xi_{i,j,k,l}$ ,  $i = 1, 2, \dots, j = 1, 2, \dots, N, k = 1, 2, \dots, M_j, l = 1, 2, \dots, C_{j,k}$  — это независимые случайные величины, и для каждого фиксированного набора значений индексов  $i, l, k$  величины  $\xi_{i,j,k,l}$ ,  $i = 1, 2, \dots$ , одинаково распределены с функцией распределения  $F_{\xi_{j,k,l}}(t)$ . Элементы подсистемы безопасности восстанавливаются только после обнаружения отказа соответствующей подсистемы безопасности. Случайное время восстановления  $k$ -й подсистемы  $j$ -й системы безопасности после ее  $i$ -го отказа обозначим  $\eta_{i,j,k}$ ,  $i = 1, 2, \dots, j = 1, 2, \dots, N, k = 1, 2, \dots, M_j$ . При этом считаем, что  $\eta_{i,j,k}$ ,  $i = 1, 2, \dots, j = 1, 2, \dots, N, k = 1, 2, \dots, M_j$ , являются независимыми случайными величинами и для каждой фиксированной пары значений индексов  $j, k$  величины  $\eta_{i,j,k}$ ,  $i = 1, 2, \dots$ , имеют одну и ту же функцию распределения  $F_{\eta_{j,k}}(t)$ . Предполагается, что моменты окончания восстановления подсистемы безопасности являются моментами регенерации процесса функционирования подсистемы безопасности. Как и ранее, на функции распределения  $F_{\xi_{j,k,l}}(t)$  и  $F_{\eta_{j,k}}(t)$  наложим

два ограничения: распределения нерешетчатые и имеют конечные математические ожидания. Обнаружение отказа подсистемы безопасности может происходить либо немедленно, либо только во время периодического контроля подсистемы безопасности. Предполагаем, что все системы безопасности, а также все подсистемы систем безопасности относятся к монотонным системам. Структурную функцию  $k$ -й подсистемы  $j$ -й системы безопасности будем обозначать  $\varphi_{j,k}(x_{j,k,1}, x_{j,k,2}, \dots, x_{j,k,C_{j,k}})$ , а структурную функцию  $j$ -й системы безопасности будем обозначать  $\psi_j(x_{j,1}, x_{j,2}, \dots, x_{j,M_j})$ . Пусть  $v$  — случайное число циклов регенерации до того, как одна из систем безопасности не сможет парировать атаку и произойдет несанкционированный доступ к защищаемой информации. Случайное время до этого события обозначим  $\omega$ . Тогда процесс функционирования рассматриваемой системы защиты информации можно рассматривать как наложение альтернирующих процессов восстановления.

Целью данной работы является получение среднего времени работы информационной системы до

осуществления несанкционированного доступа  $M\omega$  и асимптотической оценки вероятности несанкционированного доступа к защищаемой информации до момента времени  $t$ , т. е.  $P(\omega \leq t)$ .

## Основные результаты

Поскольку процесс функционирования системы защиты информации описывается как наложение альтернирующих процессов восстановления, то для случайного времени до осуществления несанкционированного доступа мы можем записать:

$$\begin{aligned} \omega = & \sum_{i=1}^{v-1} (\min(\chi_{i,1}, \chi_{i,2}, \dots, \chi_{i,N}) + \\ & + \gamma_{i,1} J_{\chi_{i,1} < \min(\chi_{i,2}, \chi_{i,3}, \dots, \chi_{i,N})} + \\ & + \gamma_{i,2} J_{\chi_{i,2} < \min(\chi_{i,1}, \chi_{i,3}, \dots, \chi_{i,N})} + \dots + \\ & + \gamma_{i,N} J_{\chi_{i,N} < \min(\chi_{i,1}, \chi_{i,2}, \dots, \chi_{i,N-1})}) + \\ & + \min(\chi_{v,1}, \chi_{v,2}, \dots, \chi_{v,N}), \end{aligned}$$

где  $J_A$  — индикатор события  $A$ . Обозначим  $\alpha_i$  — случайное время до  $i$ -й атаки на объект защиты и  $\beta_i$  — случайное время отражения этой атаки.

Очевидно

$$\begin{aligned} \alpha_i = & \min(\chi_{i,1}, \chi_{i,2}, \dots, \chi_{i,N}) \text{ и} \\ \beta_i = & \gamma_{i,1} J_{\chi_{i,1} < \min(\chi_{i,2}, \chi_{i,3}, \dots, \chi_{i,N})} + \\ & + \gamma_{i,2} J_{\chi_{i,2} < \min(\chi_{i,1}, \chi_{i,3}, \dots, \chi_{i,N})} + \dots + \\ & + \gamma_{i,N} J_{\chi_{i,N} < \min(\chi_{i,1}, \chi_{i,2}, \dots, \chi_{i,N-1})}. \end{aligned}$$

Тогда функцию распределения времени до осуществления несанкционированного доступа определим так:

$$F_{\omega}(t) = P(\omega \leq t) = P\left(\sum_{i=1}^{v-1} (\alpha_i + \beta_i) + \alpha_v \leq t\right).$$

Поскольку процесс функционирования системы защиты информации — регенерирующий процесс, то вероятность того, что произойдет несанкционированный доступ к защищаемой информации на  $n$ -м цикле регенерации, определяется формулой

$$P(v = n) = q(1 - q)^{n-1},$$

где  $q$  — это вероятность осуществления несанкционированного доступа в интервалах времени

$$\left[ \sum_{i=1}^r (\alpha_i + \beta_i), \sum_{i=1}^{r+1} (\alpha_i + \beta_i) \right), \forall r \in \{0, 1, 2, \dots\}.$$

Для дальнейшего преобразования приведенных соотношений воспользуемся интегральным преоб-

разованием Лапласа—Стилтьеса. Запишем преобразование Лапласа—Стилтьеса функции распределения  $F_\omega(t)$  в виде

$$\begin{aligned}\tilde{F}_\omega(s) &= \int_0^\infty e^{-st} dF_\omega(t) = M e^{-s\omega} = M e^{-s\omega} = \\ &= \sum_{n=1}^\infty M(e^{-s\omega}|v=n) P(v=n).\end{aligned}$$

Условное математическое ожидание  $M(e^{-s\omega}|v=n)$  перепишем в виде

$$\begin{aligned}M(e^{-s\omega}|v=n) &= M \left( e^{-s \left( \sum_{i=1}^{v-1} (\alpha_i + \beta_i) + \alpha_v \right)} \middle| v=n \right) = \\ &= M(e^{-s\alpha v} e^{-s(v-1)\beta} | v=n) = (\tilde{F}_\alpha(s))^n (\tilde{F}_\beta(s))^{n-1}.\end{aligned}$$

Функции распределения  $F_\alpha(t)$  и  $F_\beta(t)$  можно записать в явном виде

$$\begin{aligned}F_\alpha(t) &= P(\min(\chi_{i,1}, \chi_{i,2}, \dots, \chi_{i,N}) < t) = \\ &= 1 - \prod_{j=1}^N (1 - F_{\chi_j}(t)), \\ F_\beta(t) &= \sum_{j=1}^N F_{\gamma_j}(t) \int_0^\infty \left( \prod_{\substack{r=1 \\ r \neq j}}^N (1 - F_{\chi_r}(x)) \right) dF_{\chi_j}(x).\end{aligned}$$

Перепишем теперь преобразование Лапласа—Стилтьеса функции распределения времени до осуществления несанкционированного доступа  $\tilde{F}_\omega(s)$  в следующем виде:

$$\begin{aligned}\tilde{F}_\omega(s) &= \sum_{n=1}^\infty (\tilde{F}_\alpha(s))^n (\tilde{F}_\beta(s))^{n-1} q (1-q)^{n-1} = \\ &= \frac{q \tilde{F}_\alpha(s)}{1 - (1-q) \tilde{F}_\alpha(s) \tilde{F}_\beta(s)}.\end{aligned}$$

Поскольку необходимо вычислить математическое ожидание времени до осуществления несанкционированного доступа, то для этого воспользуемся

известным соотношением  $M_\omega = - \left. \frac{d\tilde{F}_\omega(s)}{ds} \right|_{s=0}$ .

Опуская промежуточные вычисления, получим

$$M_\omega = M\alpha + \frac{1-q}{q} (M\alpha + M\beta), \quad (1)$$

где  $M\alpha = \int_0^\infty \left( \prod_{j=1}^N (1 - F_{\chi_j}(t)) \right) dt$ ,

$$M\beta = \sum_{j=1}^N \int_0^\infty (1 - F_{\gamma_j}(t)) dt \int_0^\infty \left( \prod_{\substack{r=1 \\ r \neq j}}^N (1 - F_{\chi_r}(t)) \right) dF_{\chi_j}(t).$$

Для вычисления вероятности осуществления несанкционированного доступа на цикле регенерации процесса функционирования информационной системы применим предельную теорему [6], из которой следует, что

$$P\left(\frac{q\omega}{M\alpha + M\beta} > t\right) \xrightarrow{q \rightarrow 0} e^{-t}.$$

Следовательно, для искомой вероятности справедливо

$$P(\omega \leq t) \xrightarrow{q \rightarrow 0} 1 - e^{-\frac{qt}{M\alpha + M\beta}}. \quad (2)$$

Записанные выше соотношения  $M\omega$  и  $P(\omega \leq t)$  содержат пока неизвестную вероятность  $q$ . Для того чтобы оценить вероятность осуществления несанкционированного доступа на цикле регенерации процесса функционирования, воспользуемся формулой полной вероятности:

$$\begin{aligned}q &= \sum_{j=1}^N q_j P(\chi_j < \min(\chi_1, \chi_2, \dots, \chi_{j-1}, \chi_{j+1}, \dots, \chi_N)) = \\ &= \sum_{j=1}^N q_j \int_0^\infty \left( \prod_{\substack{r=1 \\ r \neq j}}^N (1 - F_{\chi_r}(t)) \right) dF_{\chi_j}(t), \quad (3)\end{aligned}$$

где  $q_j$  — вероятность осуществления несанкционированного доступа на цикле регенерации из-за отказа  $j$ -й системы безопасности.

Несанкционированный доступ осуществляется на  $i$ -м цикле регенерации из-за отказа  $j$ -й системы безопасности тогда и только тогда, когда  $\chi_{i,j} \in Q_j^-$ , где  $Q_j^-$  — множество интервалов времени, на которых  $j$ -я система безопасности не выполняет своих функций. Следовательно,

$$q_j = \int_0^\infty P(t \in Q_j^-) dF_{\chi_j}(t).$$

Показано, что асимптотическая оценка вероятности осуществления несанкционированного доступа на цикле регенерации из-за отказа  $j$ -й системы безопасности [5] можно записать в виде

$$q_j = \lim_{t \rightarrow \infty} P(t \in Q_j^-).$$

Вероятность нахождения  $j$ -й системы безопасности в исправном состоянии в момент времени  $t$

$$p_j(t) = M(\psi_j(x_{j,1}(t), x_{j,2}(t), \dots, x_{j,M_j}(t))) = h_j(p_{j,1}(t), p_{j,2}(t), \dots, p_{j,M_j}(t)),$$

где  $p_{j,k}(t)$  — вероятность нахождения  $k$ -й подсистемы  $j$ -й системы безопасности в исправном состоянии в момент времени  $t$ . Нетрудно показать, что

$$P(t \in Q_j^-) = 1 - h_j(P(t \in Q_{j,1}^+), P(t \in Q_{j,2}^+), \dots, P(t \in Q_{j,M_j}^+)),$$

где  $Q_{k,j}^+$  — множество интервалов времени, на которых  $k$ -я подсистема  $j$ -й системы безопасности успешно выполняет свои функции. Тогда

$$q_j \approx 1 - h_j(p_{j,1}, p_{j,2}, \dots, p_{j,M_j}),$$

где  $p_{j,k} = \lim_{t \rightarrow \infty} P(t \in Q_{j,k}^+)$ .

Используя формулу полной вероятности, можем записать вероятность нахождения  $k$ -й подсистемы  $j$ -й системы безопасности в исправном состоянии в момент времени  $t$

$$p_{j,k}(t) = P(t \in Q_{j,k}^+) = \int_0^\infty \int P(t \in Q_{j,k}^+ | \xi_{1,j,k} = x, \eta_{1,j,k} = y) dF_{\eta_{j,k}}(y) dF_{\xi_{j,k}}(x) = \iint_{\tau(x,y) \leq t} P(t \in Q_{j,k}^+ | \xi_{1,j,k} = x, \eta_{1,j,k} = y) dF_{\eta_{j,k}}(y) \times dF_{\xi_{j,k}}(x) + \iint_{\tau(x,y) > t} P(t \in Q_{j,k}^+ | \xi_{1,j,k} = x, \eta_{1,j,k} = y) \times dF_{\eta_{j,k}}(y) dF_{\xi_{j,k}}(x) = I_1 + I_2, \quad (4)$$

где  $\xi_{i,j,k}$  — случайная наработка до  $i$ -го отказа  $k$ -й подсистемы  $j$ -й системы безопасности;  $\eta_{1,j,k}$  —

случайная длительность восстановления после  $i$ -го отказа  $k$ -й подсистемы  $j$ -й системы безопасности. Очевидно, что случайные величины  $\xi_{i,j,k}$ ,  $i = 1, 2, \dots$ , и  $\eta_{1,j,k}$ ,  $i = 1, 2, \dots$ , одинаково распределены с функциями распределения  $F_{\xi_{j,k}}(t)$  и  $F_{\eta_{j,k}}(t)$  для каждой фиксированной пары значений индексов  $j, k$ . Для дальнейшего преобразования (4) необходимо рассмотреть, как происходит обнаружение отказа подсистемы и его восстановление.

Пусть отказы в системе безопасности обнаруживаются только во время проведения периодического контроля. Обозначим  $T_{j,k}$  период контроля и  $\theta_{j,k}$  продолжительность периодического контроля  $k$ -й подсистемы  $j$ -й системы безопасности. Отказы в системе безопасности обнаруживаются только во время периодического контроля, а во время контроля система безопасности не выполняет свои защитные функции. Поскольку исправность  $k$ -й подсистемы  $j$ -й системы безопасности периодически проверяется, то процесс функционирования  $k$ -й подсистемы  $j$ -й системы безопасности покрывается случайными непересекающимися отрезками длины  $\tau_{j,k}(\xi_{1,j,k}, \eta_{1,j,k}) = ([x/(T_{j,k} + \theta_{j,k})] + 1) \times (T_{j,k} + \theta_{j,k}) + \eta_{1,j,k}$  — циклами регенерации процесса функционирования  $k$ -й подсистемы  $j$ -й системы безопасности, а  $[x]$  обозначает целую часть  $x$ . Продолжим рассмотрение формулы (4). Область интегрирования в интеграле  $I_2$  такова, что момент регенерации  $\tau_{j,k}(x, y) = ([x/(T_{j,k} + \theta_{j,k})] + 1) \times (T_{j,k} + \theta_{j,k}) + y$  наступил после момента времени  $t$ .

В этом случае событие  $t \in Q_{j,k}^+$  может произойти только тогда, когда попадает в один из интервалов  $(0, T_{j,k}), (T_{j,k} + \theta_{j,k}, 2T_{j,k} + \theta_{j,k}), (2(T_{j,k} + \theta_{j,k}), 2(T_{j,k} + \theta_{j,k}) + \theta_{j,k}), \dots, ((k-1)(T_{j,k} + \theta_{j,k}), (k-1) \times (T_{j,k} + \theta_{j,k}) + \theta_{j,k}), (k(T_{j,k} + \theta_{j,k}), x)$ , где  $k = [x/(T_{j,k} + \theta_{j,k})]$ . Следовательно,

$$I_2 = \iint_{\tau_{j,k}(x,y) > t} \left( \sum_{r=0}^{[x/(T_{j,k} + \theta_{j,k})]} J_{t \in [r(T_{j,k} + \theta_{j,k}), r(T_{j,k} + \theta_{j,k}) + T_{j,k})} + J_{t \in [[x/(T_{j,k} + \theta_{j,k})](T_{j,k} + \theta_{j,k}), x]} \right) dF_{\eta_{j,k}}(y) dF_{\xi_{j,k}}(x) = \int_0^\infty \left( \sum_{m=0}^{[x/(T_{j,k} + \theta_{j,k})] - 1} J_{t \in [m(T_{j,k} + \theta_{j,k}), m(T_{j,k} + \theta_{j,k}) + T_{j,k})} + J_{t \in [[x/(T_{j,k} + \theta_{j,k})](T_{j,k} + \theta_{j,k}), x]} \right) dF_{\xi_{j,k}}(x) = \sum_{r=0}^\infty \int_{r(T_{j,k} + \theta_{j,k})}^{(r+1)(T_{j,k} + \theta_{j,k})} \left( \sum_{m=0}^r J_{m(T_{j,k} + \theta_{j,k}) \leq t} - \sum_{m=0}^{r-1} J_{m(T_{j,k} + \theta_{j,k}) + T_{j,k} \leq t} - J_{x \leq t} \right) dF_{\xi_{j,k}}(x).$$

Опуская аналитические преобразования, приведем сразу результат вычисления:

$$I_2 = 1 - F_{\xi}(t) - \sum_{m=1}^{\infty} (J_{(m-1)(T+\theta) + T \leq t} - J_{m(T+\theta) \leq t})(1 - F_{\xi}(m(T+\theta))) = F_{\zeta_{j,k}}(t) - F_{\xi_{j,k}}(t),$$

где  $F_{\zeta_{j,k}}(t) = 1 - \sum_{m=1}^{\infty} (1 - F_{\xi_{j,k}}(m(T_j, k + \theta_{j,k}))) \times (J_{(m-1)(T_j, k + \theta_{j,k}) + T_j, k \leq t} - J_{m(T_j, k + \theta_{j,k}) \leq t})$  — функция распределения некоторой вспомогательной случайной величины  $\zeta_{j,k}$ .

Слагаемое  $I_1$  в формуле (4) запишем так:

$$I_1 = \iint_{\tau_{j,k}(x,y) \leq t} P(t \in Q_{j,k}^+ | \xi_{1,j,k} = x, \eta_{1,j,k} = y) \times dF_{\eta_{j,k}}(y) dF_{\xi_{j,k}}(x) = \iint_{\tau_{j,k}(x,y) \leq t} p_{j,k}(t - \tau(x,y)) dF_{\eta_{j,k}}(y) dF_{\xi_{j,k}}(x) = \int_0^t p_{j,k}(t-z) dF_{\tau_{j,k}(\xi_{j,k}, \eta_{j,k})}(z).$$

Суммируя  $I_1$  и  $I_2$ , получаем интегральное уравнение в свертках:

$$p_{j,k}(t) = F_{\zeta_{j,k}}(t) - F_{\xi_{j,k}}(t) + \int_0^t p_{j,k}(t-z) dF_{\tau_{j,k}(\xi_{j,k}, \eta_{j,k})}(z), \quad (5)$$

где  $F_{\tau_{j,k}(\xi_{j,k}, \eta_{j,k})}$  — функция распределения случайной величины  $\tau_{j,k}(\xi_{j,k}, \eta_{j,k})$ .

Решение уравнения (5) в терминах преобразования Лапласа—Стилтьеса имеет вид

$$\tilde{p}_{j,k}(s) = \frac{Me^{-s\zeta_{j,k}} - Me^{-s\xi_{j,k}}}{1 - Me^{-s(\eta_{j,k} + ([\xi_{j,k}/(T_j, k + \theta_{j,k})] + 1)(T_j, k + \theta_{j,k})}}, \quad (6)$$

где  $\tilde{p}_{j,k}(s) = \int_0^{\infty} e^{-st} dp_{j,k}(t)$ ,  $Me^{-s\zeta_{j,k}} = \int_0^{\infty} e^{-st} dF_{\zeta_{j,k}}(t)$ ,

$Me^{-s\xi_{j,k}} = \int_0^{\infty} e^{-st} dF_{\xi_{j,k}}(t)$  и  $Me^{-s\tau_{j,k}(\xi_{j,k}, \eta_{j,k})} =$

$$= \int_0^{\infty} e^{-st} dF_{\tau_{j,k}(\xi_{j,k}, \eta_{j,k})}(t).$$

Поскольку вычислить обратное преобразование Лапласа практически невозможно, то, как обычно, находим асимптотическое решение уравнения (5),

для чего, используя теоремы преобразования Лапласа, вычисляем  $\lim_{s \rightarrow 0} \tilde{p}_{j,k}(s)$  при  $s \rightarrow 0$ . Имеем

$$p_{jk} = \lim_{s \rightarrow 0} \tilde{p}_{j,k}(s) = \frac{M\xi_{j,k} - M\zeta_{j,k}}{M(\eta_{j,k} + ([\xi_{j,k}/(T_j, k + \theta_{j,k})] + 1)(T_j, k + \theta_{j,k}))} = K_{\Gamma\text{СБ}_{j,k}}, \quad (7)$$

где  $K_{\Gamma\text{СБ}_{j,k}}$  — коэффициент готовности  $k$ -й подсистемы  $j$ -й системы безопасности. В соотношении (7) неизвестным остается  $M\zeta_{j,k}$ , которое вычислим следующим образом:

$$M\zeta_{j,k} = \int_0^{\infty} (1 - F_{\zeta_{j,k}}(t)) dt = \int_0^{\infty} \sum_{m=1}^{\infty} (J_{(m-1)(T_j, k + \theta_{j,k}) + T_j, k \leq t} - J_{m(T_j, k + \theta_{j,k}) \leq t}) \times (1 - F_{\xi_{j,k}}(m(T_j, k + \theta_{j,k}))) dt = \sum_{m=1}^{\infty} m\theta_{j,k} (F_{\xi_{j,k}}((m+1)(T_j, k + \theta_{j,k})) - F_{\xi_{j,k}}(m(T_j, k + \theta_{j,k}))) = \theta_{j,k} M[\xi_{j,k}/(T_j, k + \theta_{j,k})].$$

Окончательно асимптотическое решение уравнения (5) запишем в виде

$$p_{jk} = \frac{M\xi_{j,k} - \theta_{j,k} M[\xi_{j,k}/(T_j, k + \theta_{j,k})]}{M(\eta_{j,k} + ([\xi_{j,k}/(T_j, k + \theta_{j,k})] + 1)(T_j, k + \theta_{j,k}))},$$

где  $\xi_{j,k} - \theta_{j,k} [\xi_{j,k}/(T_j, k + \theta_{j,k})]$  — наработка до отказа  $k$ -й подсистемы  $j$ -й системы безопасности за вычетом затрат времени на проведение контрольных профилактик.

Полученное соотношение можно несколько упростить, если предположить, что функция распределения  $F_{\xi_{j,k}}(t)$  — относительно монотонна (что выполняется для большинства распределений, применяемых в теории надежности). Тогда  $M[\xi_{j,k}/(T_j, k + \theta_{j,k})] \approx M\xi_{j,k}/(T_j, k + \theta_{j,k}) - 1/2$  и  $M\tau_{j,k}(\xi_{j,k}, \eta_{j,k}) \approx M\eta_{j,k} + M\xi_{j,k} + \frac{T_j, k + \theta_{j,k}}{2}$ . Это позволяет переписать (7) в виде

$$p_{jk} \approx \frac{M\xi_{j,k} \frac{T_j, k}{T_j, k + \theta_{j,k}} + \theta_{j,k}/2}{M\eta_{j,k} + M\xi_{j,k} + \frac{T_j, k + \theta_{j,k}}{2}}.$$

### Пример вычисления показателей надежности

Рассмотрим следующий пример. Пусть информационная система состоит из трех систем безопасности и одного защищаемого объекта. Предположим, что каждая из систем состоит из двух подсистем: первая система безопасности представлена на рис. 1, а составляющие ее подсистемы на рис. 2, 3; вторая система безопасности изображена на рис. 4,

ее подсистемы на рис. 5, 6; третья система — на рис. 7, а соответствующие ей подсистемы на рис. 8, 9. Очевидно, что для рассматриваемой системы:  $N = 3$ ,  $M_1 = 2$ ,  $M_2 = 2$ ,  $M_3 = 2$ . Далее предположим, что

$$F_{\lambda_j}(t) = 1 - e^{-\lambda_{\lambda_j} t}, j = 1, 2, \dots, N; F_{\gamma_j}(t) = 1 - e^{-\lambda_{\gamma_j} t},$$

$$j = 1, 2, \dots, N; F_{\xi_{j,k,l}}(t) = 1 - e^{-\lambda_{\xi_{j,k,l}} t}, j = 1, 2, \dots, N,$$

$$k = 1, 2, \dots, M_j, l = 1, 2, \dots, C_{j,k} \text{ и } F_{\eta_{j,k}}(t) = 1 - e^{-\lambda_{\eta_{j,k}} t},$$

$$j = 1, 2, \dots, N, k = 1, 2, \dots, M_j. \text{ Каждая подсистема}$$

состоит из различного числа элементов:  $C_{1,1} = 4$ ,  $C_{1,2} = 4$ ,  $C_{2,1} = 1$ ,  $C_{2,2} = 2$ ,  $C_{3,1} = 4$ ,  $C_{3,2} = 5$ .

Предположим также, что отказы всех подсистем безопасности обнаруживаются только во время проведения запланированных периодических контролей и во время выполнения контрольных мероприятий подсистемы безопасности выполняют свои защитные функции. В этом случае (7) перепишем так:

$$p_{jk} \approx \frac{M_{\xi_{j,k}}}{M_{\eta_{j,k}} + M_{\xi_{j,k}} + \frac{T_{j,k} + \theta_{j,k}}{2}}.$$

Записанное соотношение будем использовать при дальнейших вычислениях. Оценим среднее время работы информационной системы до осуществления несанкционированного доступа  $M\omega$  и вероятность несанкционированного доступа к защищаемой информации до момента времени  $t$ . Для каждой подсистемы приведем результаты вычисления соответствующих показателей надежности подсистем и систем безопасности. Рассмотрим первую систему безопасности, представленную на рис. 1. Нетрудно получить  $h_1(t)$  — вероятность безотказной работы первой системы безопасности за время  $t$ :

$$h_1(t) = (2p_{1,1}(t) - p_{1,1}(t)^2)p_{1,2}(t).$$

Рассматривая каждую из подсистем безопасности в отдельности, получим: вероятность безотказной работы первой (см. рис. 2, 3) и второй (см. рис. 4–6) систем безопасности; математические ожидания наработок подсистем; асимптотическое соотношение вероятностей нахождения первой и второй подсистем первой системы безопасности в работоспособном состоянии:

$$h_{1,1}(t) = (2p_{1,1,1}(t) - p_{1,1,1}(t)^2)p_{1,1,2}(t)p_{1,1,3}(t)p_{1,1,4}(t);$$

$$p_{1,1} = \frac{M_{\xi_{1,1}}}{M_{\tau_{1,1}}(\xi_{1,1}, \eta_{1,1})};$$

$$M_{\xi_{1,1}} = \frac{2}{\lambda_{\xi_{1,1,1}} + \lambda_{\xi_{1,1,2}} + \lambda_{\xi_{1,1,3}} + \lambda_{\xi_{1,1,4}}} - \frac{1}{2\lambda_{\xi_{1,1,1}} + \lambda_{\xi_{1,1,2}} + \lambda_{\xi_{1,1,3}} + \lambda_{\xi_{1,1,4}}};$$

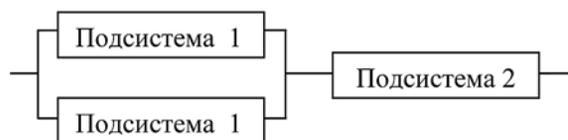


Рис. 1. Блок-схема надежности первой системы безопасности

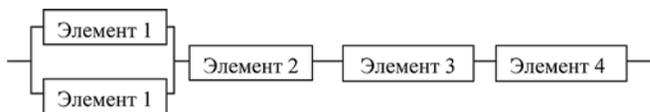


Рис. 2. Блок-схема надежности подсистемы 1 первой системы безопасности

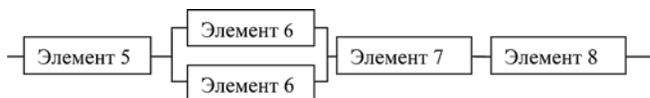


Рис. 3. Блок-схема надежности подсистемы 2 первой системы безопасности

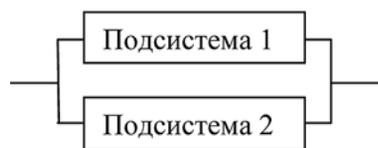


Рис. 4. Блок-схема надежности второй системы безопасности

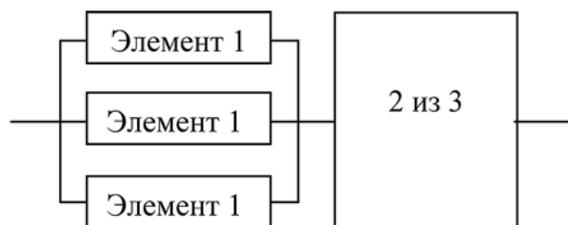


Рис. 5. Блок-схема надежности подсистемы 1 второй системы безопасности



Рис. 6. Блок-схема надежности подсистемы 2 второй системы безопасности

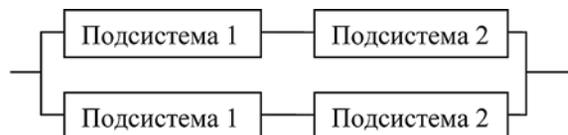


Рис. 7. Блок-схема надежности третьей системы безопасности

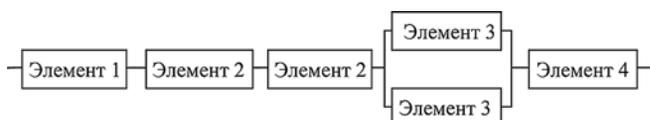


Рис. 8. Блок-схема надежности подсистемы 1 третьей системы безопасности



Рис. 9. Блок-схема надежности подсистемы 2 третьей системы безопасности

$$M\tau_{1,1}(\xi_{1,1}, \eta_{1,1}) = \frac{1}{\lambda_{\eta_{1,1}}} + \frac{2}{\lambda_{\xi_{1,1,1}} + \lambda_{\xi_{1,1,2}} + \lambda_{\xi_{1,1,3}} + \lambda_{\xi_{1,1,4}}} - \frac{1}{2\lambda_{\xi_{1,1,1}} + \lambda_{\xi_{1,1,2}} + \lambda_{\xi_{1,1,3}} + \lambda_{\xi_{1,1,4}}} + \frac{T_{1,1} + \theta_{1,1}}{2};$$

$$h_{1,2}(t) = p_{1,2,1}(t)(2p_{1,2,2}(t) - p_{1,2,2}(t)^2)p_{1,2,4}(t)p_{1,2,3}(t);$$

$$p_{1,2} = \frac{M\xi_{1,2}}{M\tau_{1,2}(\xi_{1,2}, \eta_{1,2})};$$

$$M\xi_{1,2} = \frac{2}{\lambda_{\xi_{1,2,1}} + \lambda_{\xi_{1,2,2}} + \lambda_{\xi_{1,2,3}} + \lambda_{\xi_{1,2,4}}} - \frac{1}{\lambda_{\xi_{1,2,1}} + 2\lambda_{\xi_{1,2,2}} + \lambda_{\xi_{1,2,3}} + \lambda_{\xi_{1,2,4}}};$$

$$M\tau_{1,2}(\xi_{1,2}, \eta_{1,2}) = \frac{1}{\lambda_{\eta_{1,2}}} + \frac{2}{\lambda_{\xi_{1,2,1}} + \lambda_{\xi_{1,2,2}} + \lambda_{\xi_{1,2,3}} + \lambda_{\xi_{1,2,4}}} - \frac{1}{\lambda_{\xi_{1,2,1}} + 2\lambda_{\xi_{1,2,2}} + \lambda_{\xi_{1,2,3}} + \lambda_{\xi_{1,2,4}}} + \frac{T_{1,2} + \theta_{1,2}}{2}.$$

Вероятность осуществления несанкционированного доступа на цикле регенерации из-за отказа первой системы безопасности определяется формулой

$$q_1 \approx 1 - (p_{1,1} + p_{1,2} - p_{1,1}p_{1,2})^2.$$

Рассмотрим вторую систему безопасности (см. рис. 4). Для каждой подсистемы приведем результаты вычисления соответствующих показателей надежности подсистем и систем:

$$h_{2,1}(t) = p_{2,1}(t) + p_{2,2}(t) - p_{2,1}(t)p_{2,2}(t);$$

$$h_{2,1}(t) = 3p_{2,1,1}(t)^2 - 2p_{2,1,1}(t)^3;$$

$$p_{2,1} = \frac{M\xi_{2,1}}{M\tau_{2,1}(\xi_{2,1}, \eta_{2,1})}; M\xi_{2,1} = \frac{5}{6\lambda_{\xi_{2,1,1}}};$$

$$M\tau_{2,1}(\xi_{2,1}, \eta_{2,1}) = \frac{1}{\lambda_{\eta_{2,1}}} + \frac{5}{6\lambda_{\xi_{2,1,1}}} + \frac{T_{2,1} + \theta_{2,1}}{2};$$

$$h_{2,2}(t) = p_{2,2,1}(t)p_{2,2,2}(t);$$

$$p_{2,2} = \frac{M\xi_{2,2}}{M\tau_{2,2}(\xi_{2,2}, \eta_{2,2})}; M\xi_{2,2} = \frac{1}{\lambda_{\xi_{2,2,1}} + \lambda_{\xi_{2,2,2}}};$$

$$M\tau_{2,2}(\xi_{2,2}, \eta_{2,2}) = \frac{1}{\lambda_{\eta_{2,2}}} + \frac{1}{\lambda_{\xi_{2,2,1}} + \lambda_{\xi_{2,2,2}}} + \frac{T_{2,2} + \theta_{2,2}}{2};$$

$$q_2 \approx 1 - (p_{2,1}(t) + p_{2,2}(t) - p_{2,1}(t)p_{2,2}(t));$$

$$h_3(t) = 2p_{3,1}(t)p_{3,2}(t) - (p_{3,1}(t)p_{3,2}(t))^2;$$

$$h_{3,1}(t) = p_{3,1,1}(t)p_{3,1,2}(t)^2(2p_{3,1,3}(t) - p_{3,1,3}(t)^2)p_{3,1,4}(t);$$

$$p_{3,1} = \frac{M\xi_{3,1}}{M\tau_{3,1}(\xi_{3,1}, \eta_{3,1})};$$

$$M\xi_{3,1} = \frac{2}{\lambda_{\xi_{3,1,1}} + 2\lambda_{\xi_{3,1,2}} + \lambda_{\xi_{3,1,3}} + \lambda_{\xi_{3,1,4}}} - \frac{1}{\lambda_{\xi_{3,1,1}} + 2\lambda_{\xi_{3,1,2}} + 2\lambda_{\xi_{3,1,3}} + \lambda_{\xi_{3,1,4}}};$$

$$M\tau_{3,1}(\xi_{3,1}, \eta_{3,1}) = \frac{1}{\lambda_{\eta_{3,1}}} + \frac{2}{\lambda_{\xi_{3,1,1}} + 2\lambda_{\xi_{3,1,2}} + \lambda_{\xi_{3,1,3}} + \lambda_{\xi_{3,1,4}}} - \frac{1}{\lambda_{\xi_{3,1,1}} + 2\lambda_{\xi_{3,1,2}} + 2\lambda_{\xi_{3,1,3}} + \lambda_{\xi_{3,1,4}}} + \frac{T_{3,1} + \theta_{3,1}}{2};$$

$$h_{3,2}(t) = p_{3,2,1}(t)p_{3,2,2}(t)p_{3,2,3}(t)p_{3,2,4}(t)p_{3,2,5}(t);$$

$$p_{3,2} = \frac{M\xi_{3,2}}{M\tau_{3,2}(\xi_{3,2}, \eta_{3,2})};$$

$$M\xi_{3,2} = \frac{1}{\lambda_{\xi_{3,2,1}} + \lambda_{\xi_{3,2,2}} + \lambda_{\xi_{3,2,3}} + \lambda_{\xi_{3,2,4}} + \lambda_{\xi_{3,2,5}}};$$

$$M\tau_{3,2}(\xi_{3,2}, \eta_{3,2}) = \frac{1}{\lambda_{\eta_{3,2}}} + \frac{1}{\lambda_{\xi_{3,2,1}} + \lambda_{\xi_{3,2,2}} + \lambda_{\xi_{3,2,3}} + \lambda_{\xi_{3,2,4}} + \lambda_{\xi_{3,2,5}}} + \frac{T_{3,2} + \theta_{3,2}}{2};$$

$$q_3 \approx 1 - 2p_{3,1}(t)p_{3,2}(t) + (p_{3,1}(t)p_{3,2}(t))^2.$$

Наконец, используя формулу (3), получаем вероятность осуществления несанкционированного доступа к защищаемой информации на цикле регенерации

$$q = \frac{q_1\lambda_{\chi_1} + q_2\lambda_{\chi_2} + q_3\lambda_{\chi_3}}{\lambda_{\chi_1} + \lambda_{\chi_2} + \lambda_{\chi_3}}.$$

Подставляя в (1) и (2) математические ожидания

$$M\alpha = \frac{1}{\lambda_{\chi_1} + \lambda_{\chi_2} + \lambda_{\chi_3}} \text{ и } M\beta = \frac{1}{\lambda_{\chi_1} + \lambda_{\chi_2} + \lambda_{\chi_3}} \left( \frac{\lambda_{\chi_1}}{\lambda_{\gamma_1}} + \frac{\lambda_{\chi_2}}{\lambda_{\gamma_2}} + \frac{\lambda_{\chi_3}}{\lambda_{\gamma_3}} \right),$$

и вычислительную вероятность  $q$ , получаем математическое ожидание времени работы информационной системы до осуществления несанкционированного доступа к информации

$$M\omega = \frac{1}{q_1\lambda_{\chi_1} + q_2\lambda_{\chi_2} + q_3\lambda_{\chi_3}} \times \left( 1 + \frac{(1-q_1)\lambda_{\chi_1} + (1-q_2)\lambda_{\chi_2} + (1-q_3)\lambda_{\chi_3}}{\lambda_{\chi_1} + \lambda_{\chi_2} + \lambda_{\chi_3}} \times \left( \frac{\lambda_{\chi_1}}{\lambda_{\gamma_1}} + \frac{\lambda_{\chi_2}}{\lambda_{\gamma_2}} + \frac{\lambda_{\chi_3}}{\lambda_{\gamma_3}} \right) \right),$$

а асимптотическую оценку вероятности несанкционированного доступа к защищаемой информации до момента времени  $t$  запишем в виде

$$P(\omega \leq t) \xrightarrow{q \rightarrow 0} 1 - e^{-Ht},$$

$$\text{где } H = \frac{q_1 \lambda_{\chi_1} + q_2 \lambda_{\chi_2} + q_3 \lambda_{\chi_3}}{1 + \frac{\lambda_{\chi_1}}{\lambda_{\gamma_1}} + \frac{\lambda_{\chi_2}}{\lambda_{\gamma_2}} + \frac{\lambda_{\chi_3}}{\lambda_{\gamma_3}}}.$$

Аналогично можно получить эти показатели надежности информационной системы, восстановительные мероприятия которой выполняются иначе.

### Заключение

Предложенная математическая модель позволяет вычислять показатели надежности информационной системы, состоящей из объекта защиты и  $N$  систем безопасности, такие как среднее время работы информационной системы, до осуществления несанкционированного доступа к информации и асимптотическую оценку вероятности несанкционированного доступа к защищаемой информации до момента времени  $t$ . Математическая модель надеж-

ности информационной системы учитывает структуру каждой системы безопасности и особенности работоспособного состояния после отказа, а также случайные времена атак на объект защиты. Рассмотрен пример вычисления показателей надежности.

### Список литературы

1. **Щеглов А. Ю.** Защита компьютерной информации от несанкционированного доступа. СПб.: Наука и техника, 2004. 384 с.
2. **Мамиков А. Г., Кульба В. В., Щелков А. Б.** Достоверность, защита и резервирование информации в АСУ. М.: Энергоатомиздат, 1986. 304 с.
3. **Расторгуев С. П.** Программные методы защиты информации в компьютерах и сетях. М.: Изд-во агентства "Яхтсмен", 1993. 188 с.
4. **Перегула А. И., Тимашов Д. А.** Математическая модель надежности систем защиты информации // Информационные технологии. 2009. № 8. С. 10—17.
5. **Перегула А. И., Тимашов Д. А.** Математическая модель надежности информационных систем с эшелонированной защитой от несанкционированного доступа // Информационные технологии. 2012. № 12. С. 61—66.
6. **Kovalenko I. N., Kuznetsov N. Yu. & Pegg P. A.** Mathematical Theory of Reliability of Time Dependent Systems with Practical Applications. John Wiley & Sons. 1997.
7. **Rausand M. & Høyland A.** System Reliability Theory: Model, Statistical Methods and Applications. 2<sup>nd</sup> ed. John Wiley & Sons. 2004.

**A. I. Pereguda**, Professor, e-mail: pereguda@iate.obninsk.ru,  
Obninsk Institute of Nuclear Energetics NRNU MEPhI, Kaluga region

## Mathematical Model of Reliability of Information System, Consisting of Object Protection and $N$ Security Systems

*The article describes the mathematical model of reliability informational system consisting of information protection object and security systems, which parrying possible unauthorized access. Failures information protection object and security subsystems are independent. Each of the security systems parries only one type of attacks on the security object. It is assumed that state of reliability of safety systems checked periodically and after failure detection performed it renewal. Mathematical model of reliability of the information system is a superposition of random renewal processes. The mathematical model takes into account the structure of each of  $N$  systems security, renewal their in state of operability after failure, and as well capabilities attacks in random times. Hence, the moments of closure renewal the security systems it is moments renewal of the information system. The mathematical model of reliability take into account of structural feature each out of  $N$  the systems security, recovery work after a failure, as well attacks in random times on information protection objects. Analysis of mathematical models of reliability information system allows us to calculate the expectation the random time to unauthorized access, as well as an asymptotic estimation of the probability unauthorized access to information system up the time  $t$ . As an example of calculating the reliability of the information system is considered, consisting of object information protection and security of the three systems, each of which consists of two subsystems.*

**Keywords:** reliability, security system, random values, failure time, stochastic process, expectation of the time, distribution function

### References

1. **Szeglov A. Yu.** Zashchita kompyuternoi informazii ot nesankzionirovannogo dostupa. SPb.: Nauka i tehnika, 2004. 384 p.
2. **Mamikonov A. G., Kulba V. V., Szelkov A. B.** Dostovernost, zashchita i rezervirovanie informazii v ASU. M.: Energoatomizdat, 1986. 304 p.
3. **Rastorguev S. P.** Programmnie metodi zashchity informazii v kompyuterax i setiyh. M.: Izd-vo agenstva "Iahtsmen", 1993. 188 p.
4. **Pereguda A. I., Timasov D. A.** Matematizatskaia model nadezhnosti sistem zashchity informazii. *Informazionay tehnologiy.* 2009. N. 8. P. 10—17.
5. **Pereguda A. I., Timasov D. A.** Matematizatskaia model nadezhnosti sistem informazionih sistem s esalonirovanoj zashchitoy ot nesankzionirovannogo dostupa. *Informazionay tehnologiy.* 2012. N. 2. P. 61—66.
6. **Kovalenko I. N., Kuznetsov N. Yu. & Pegg P. A.** *Mathematical Theory of Reliability of Time Dependent Systems with Practical Applications.* John Wiley & Sons, Ltd. 1997.
7. **Rausand M. & Høyland A.** *System Reliability Theory: Model, Statistical Methods and Applications.* 2<sup>nd</sup> ed. John Wiley & Sons, Ltd. 2004.

# СИСТЕМЫ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ CAD-SYSTEMS

УДК 621.382 + 621.373.5

**М. М. Гурарий**, канд. техн. наук, ст. науч. сотр., **М. М. Жаров**, канд. техн. наук, вед. науч. сотр.,  
**С. Г. Русаков**, чл.-корр. РАН, д-р техн. наук, гл. науч. сотр., **С. Л. Ульянов**, д-р техн. наук, зав. отделом,  
Институт проблем проектирования в микроэлектронике РАН (ИППМ РАН), г. Москва,  
e-mail: rusakov@ippm.ru

## Методы моделирования в частотной области периодических режимов автогенераторных схем

*Обсуждаются современные методы моделирования установившихся периодических режимов для анализа автогенераторных схем. Методы ориентированы на применение в программах схемотехнического моделирования, они позволяют анализировать произвольные конфигурации автогенераторных схем и обеспечивают расчет характеристик при внешнем периодическом воздействии произвольной формы.*

**Ключевые слова:** схемотехническое моделирование, автогенераторные схемы, установившийся периодический режим, гармонический баланс, метод искусственного зонда, синхронизация автогенераторов

### Введение

Автономные генераторы являются одним из важнейших компонентов электронных схем и широко применяются для получения сигналов с заданной частотой. При схемотехническом проектировании таких схем базовым видом моделирования является определение установившегося периодического режима. Моделирование многопериодных нелинейных колебаний с помощью стандартного анализа переходных процессов требует значительных временных затрат. Особенно большие затраты связаны с расчетом высокочастотных генераторов ввиду продолжительного временного интервала установления стационарного решения. Для моделирования автогенераторных схем более эффективны специализированные методы, которые позволяют непосредственно находить установившийся периодический режим без моделирования переходного процесса.

Задача определения установившегося периодического режима может быть сформулирована как в частотной, так и во временной области. По сравнению с неавтономными схемами определение установившегося режима автогенераторов вызывает дополнительные вычислительные трудности, связанные с неизвестной заранее частотой колебаний.

Следует отметить, что по ряду причин наиболее эффективным методом в системах моделирования нелинейных радиотехнических схем признан метод гармонического баланса (ГБ) [1]. Метод ГБ относится к классу частотных методов и широко применяется на практике в ряде коммерческих систем

для определения установившегося периодического режима неавтономных схем. В связи с этим актуальной задачей является развитие надежных вычислительных процедур на базе ГБ, обеспечивающих основные виды моделирования автогенераторных схем.

Цель настоящей статьи — проанализировать основные современные подходы решения вычислительных задач, поддерживающих цикл моделирования автогенераторных схем в частотной области.

Ниже рассмотрены вычислительные подходы решения периодических задач для генераторов, а также методы моделирования генераторных схем при малосигнальном внешнем воздействии. Обсуждаются проблема применения метода ГБ при решении периодических задач для генераторов, связанная с притягиванием к тривиальному решению, и вычислительные процедуры на базе комбинации метода ГБ и метода продолжения по параметру для ее решения. Рассмотрены вопросы моделирования режимов синхронизации генераторов с определением условий захвата частоты и моделирования поведения генераторов при их взаимовлиянии.

### 1. Частотные методы анализа установившегося режима

Поведение электронной схемы описывается системой нелинейных дифференциальных уравнений первого порядка [1]

$$r(x(t)) = i(x(t)) + \frac{d}{dt} q(x(t)) + u(t) = 0. \quad (1)$$

Здесь  $x(t)$ ,  $i(x)$ ,  $q(x)$  — векторы узловых переменных, токов, зарядов;  $u(t)$  — вектор входных воздействий. Все векторы имеют размерность  $L$ .

При воздействии на схему периодического входного сигнала с периодом  $T$  (частотой  $\omega = 2\pi/T$ ) установившийся периодический режим в схеме может быть определен решением системы (1) с условием периодичности  $x(0) = x(T)$ .

Другим способом нахождения установившегося режима является метод гармонического баланса, в рамках которого решение (1) представляется в виде ряда Фурье:

$$x_l(t) = \sum_k X_{kl} e^{jk\omega t}, \quad l = 1, \dots, L, \quad (2)$$

где  $X_{kl}$  — это амплитуда  $k$ -й гармоники напряжения в  $l$ -м узле схемы.

Подставив (2) в решение (1), можно получить систему уравнений метода ГБ в виде [1, 2]

$$R(X) = I(X) + j\Omega Q(X) + U = 0. \quad (3)$$

Здесь  $X$ ,  $I$ ,  $Q$  — векторы амплитуд гармоник узловых напряжений, токов и зарядов;  $\Omega$  — блочно-диагональная матрица гармоник собственной частоты  $\omega$ . Векторное выражение (3) — это система алгебраических уравнений относительно всех гармоник всех узловых напряжений схемы.

Для автономных схем входное воздействие отсутствует ( $U = 0$ ), а частота  $\omega$  включается в вектор неизвестных. Для того чтобы число переменных соответствовало числу уравнений вводится дополнительное условие нулевой фазы первой гармоники напряжения в заданном  $p$ -м узле. Это условие записывается как равенство нулю мнимой части соответствующей комплексной гармоники и обеспечивает выбор единственного из бесконечного множества решений, различающихся фазовым сдвигом. Система уравнений ГБ принимает в этом случае следующий вид [1, 2]:

$$R(X, \omega) = I(X) + j\Omega Q(X) = 0, \quad \text{Im}(X_{1,p}) = 0. \quad (4)$$

Для решения системы (4) применяется, как правило, метод Ньютона, на каждом шаге которого решается система линейных алгебраических уравнений

$$J\Delta X + \Lambda\Delta\omega = -R(X, \omega), \quad \text{Im}(\Delta X_{1,p}) = 0, \quad (5)$$

где  $\Delta X$ ,  $\Delta\omega$  — это коррекции решения на шаге Ньютона;  $J = \partial R(X, \omega) / \partial X$  — матрица Якоби системы (4);  $\Lambda = \partial R(X, \omega) / \partial \omega$ . Проблема применения метода Ньютона (5) для решения системы уравнений (4) связана с необходимым учетом вырожденного (тривиального) решения, которое соответствует статической рабочей точке (нулевым амплитудам всех гармоник напряжений и произвольному значению частоты). Для исключения сходимости алгоритма Ньютона к вырожденному решению необходимо задавать начальное приближение, достаточно близкое к требуемому периодическому решению [2–4].

Однако при разработке генераторной схемы для пользователя практически невозможно оценить желательный высокоразмерный вектор начальных гармоник. Поэтому для улучшения сходимости при моделировании генераторных схем введена концепция искусственного зонда [5], который представляет собой соединение источника напряжения или тока с идеальным фильтром первой гармоники. Введение зонда в схему автогенератора позволяет свести автономную систему уравнений к неавтономной системе уравнений с дополнительными условиями [5]:

$$\text{Re}\{I_{probe}(V_{probe}, \omega)\} = 0, \quad \text{Im}\{I_{probe}(V_{probe}, \omega)\} = 0, \quad (6)$$

где  $I_{probe}$ ,  $V_{probe}$  — ток и напряжение искусственного зонда. Вычислительная схема в этом случае состоит из внешнего и внутреннего блоков. Во внешнем блоке решается система двух уравнений (6), относительно амплитуды и частоты искусственного зонда. Во внутреннем блоке решается неавтономная система уравнений гармонического баланса (3) с найденными значениями амплитуды и частоты искусственного зонда. Полученное решение для схемы с зондом является решением для исходной схемы (4), так как выполнение условий нулевого тока (6) эквивалентно отсутствию зонда. Применение такого подхода позволяет задавать начальные значения только для двух неизвестных:  $V_{probe}$  и  $\omega$ . Оценка начального значения частоты  $\omega$  может проводиться на основе алгоритма анализа полюсов передаточной функции линеаризованной схемы в статической рабочей точке [6], и тогда задачей пользователя остается лишь задание корректного значения амплитуды  $V_{probe}$ .

Дальнейшее совершенствование метода основано на полной автоматизации оценки начального приближения. Для этой оценки было предложено использовать значение  $V_{probe}$ , при котором достигается минимум зависимости  $|I_{probe}(V_{probe})|$  [5]. Предложено несколько модификаций этого метода. В работе [7] для улучшения свойств сходимости внешнего итерационного процесса предлагается оптимизационная процедура. В работе [8] внешний итерационный процесс выполняется на основе метода гомотопии. В работе [9] используется полигармонический искусственный зонд, для которого требуются начальные значения высших гармоник. В работе [10] предложена процедура автоматического выбора узла подключения искусственного зонда и алгоритм решения (6) на основе анализа изменения активной и реактивной мощности при вариации амплитуды зонда.

Основной недостаток подхода работы [5] состоит в его недостаточной надежности. Как показано в работах [11,12], для высокочастотных автогенераторов зависимость  $|I_{probe}(V_{probe})|$  часто может не иметь локального минимума, а в тех случаях, когда он присутствует, полученное приближение не всегда обеспечивает сходимость итераций.

Проведенный анализ [11, 12] показал, что источник недостатков метода связан с постоянством частоты при построении зависимости  $|I_{probe}(V_{probe})|$ . Для его устранения предлагается новый тип искусственно встраиваемого зонда с фиксацией не частоты зонда, а фазы его тока, которая принимается нулевой относительно сигнала зонда. При изменении амплитуды в процессе продолжения по параметру разрешается изменение частоты зонда так, чтобы обеспечивалась нулевая фаза тока зонда. Этот метод был назван методом продолжения решения по параметру с применением "зонда с подстройкой частоты" (ПЧ) [11, 12]. Вычислительная схема метода включает процедуру трассировки траектории кривой  $\text{Re}(I_{probe}(V_{probe}, \omega(V_{probe})))$  при определении зависимости  $\omega(V_{probe})$  из решения уравнения

$\text{Im}(I_{probe}(V_{probe}, \omega(V_{probe}))) = 0$ . В процедуре продолжения осуществляется управление длиной шага, а также автоматическое изменение параметра продолжения для вычисления траектории в случае ее неоднозначности [13]. Процесс завершается, когда значение отклика схемы (действительная часть тока зонда) становится равным нулю.

Наиболее важная особенность предложенного метода с ПЧ — существенное улучшение сходимости по сравнению с методом без ПЧ. Сравнительный анализ методов [13], проведенный для набора из 11 тестовых генераторных схем, экспериментально подтвердил этот вывод. Метод позволяет находить решение в случае сложных траекторий и при условии существования множества решений [13].

На рис. 1, а представлена схема генератора на биполярных транзисторах, а на рис. 1, б — полученная траектория кривой отклика  $\text{Re}(I_{probe}(V_{probe}))$ . Этот пример иллюстрирует расчет сложной траектории с неоднозначностями с применением процедуры [12, 13] с подстройкой частоты и автоматическим изменением параметра продолжения.

## 2. Методы анализа генераторов с внешней синхронизацией

При воздействии внешнего периодического сигнала на схему автогенератора в зависимости от некоторых условий устанавливается режим колебаний с частотой, равной, кратной или дробно-кратной частоте внешнего возбуждения. Условия, при которых реализуется такой режим синхронизации, определяются амплитудой и частотой воздействия, а также свойствами самой схемы. В простейшем случае синхронизация на частоте воздействия происходит при достаточно больших значениях амплитуды и частотах внешнего воздействия, близких к собственной частоте колебаний. Спектр сигнала автогенератора содержит гармоники частоты внешнего воздействия. Если внешнее воздействие не удовлетворяет условиям синхронизации (захвата частоты), то реализуется квазипериодический режим, при котором спектр сигнала автогенератора содержит комбинационные компоненты двух фундаментальных частот, одна из которых — это частота внешнего воздействия, а вторая зависит от параметров генератора и сигнала возбуждения. Целью анализа является определение условий синхронизации генератора, т. е. характеристик внешнего сигнала, при которых реализуется чисто периодический режим работы генератора.

В приближении малости амплитуды внешнего возбуждения в классической работе Адлера получено дифференциальное уравнение для медленно меняющейся фазы  $\vartheta$  выходного сигнала [14]:

$$\frac{d\vartheta}{dt} = \omega_0 - \omega_{inj} - \frac{A_{inj}}{A_{osc}} \frac{\omega_0}{2Q} \sin(\vartheta), \quad (7)$$

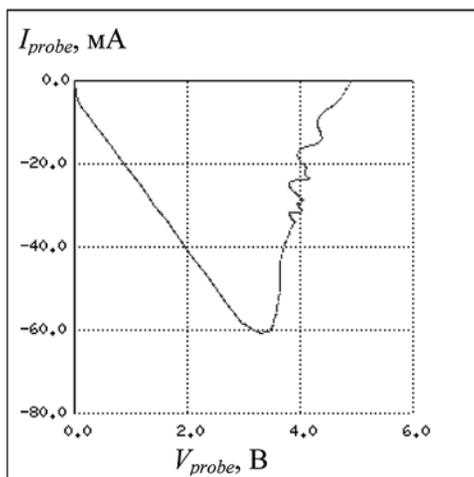
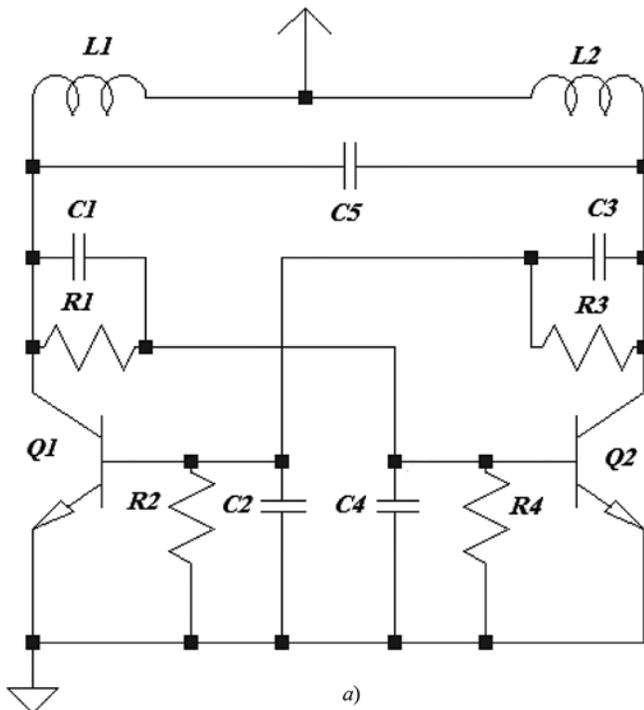


Рис. 1. Схема генератора (а) и траектория отклика (б)

где  $\omega_0$  — собственная частота автогенератора;  $A_{inj}$ ,  $A_{osc}$  — амплитуды входного сигнала и напряжения на емкости;  $Q$  — добротность  $LC$ -контура.

В режиме синхронизации фаза постоянна и  $d\theta/dt = 0$ , поэтому дифференциальное уравнение (7) сводится к алгебраическому уравнению относительно неизвестной фазы захвата. Решение этого уравнения может существовать в ограниченной области амплитуд и частот входного возбуждения, которая определяет диапазон захвата автогенератора:

$$|\omega_0 - \omega_{inj}| < \frac{A_{inj}}{A_{osc}} \frac{\omega_0}{2Q}. \quad (8)$$

Уравнение Адлера применимо к слабо нелинейным генераторам и синусоидальным внешним воздействиям. Уравнение включает параметр  $Q$ , что ограничивает его применимость только  $LC$ -автогенераторами.

Фазовое дифференциальное уравнение, полученное на основе теории Флоке, имеет вид [15]

$$d\alpha/dt = v^T(t + \alpha)b(t), \quad (9)$$

где  $\alpha$  — неизвестная фазовая переменная;  $b(t)$  — вектор внешних воздействий;  $v(t)$  — вектор проекции возмущений, введенный в работе [15], алгоритм вычисления которого представлен в работе [16].

В отличие от уравнения (7) фазовое дифференциальное уравнение (9) можно применять к произвольной автогенераторной схеме. Однако оно не позволяет получить условия захвата простой фиксацией нулевого значения фазовой производной по времени, так как в этом случае решение (9) с постоянной частотой не существует. В работе [17] предложено находить установившееся решение усредненного по времени уравнения (9). Данный подход позволил получить выражение для области захвата в случае синусоидального внешнего возбуждения с частотой, близкой к собственной частоте [18]. Случай произвольной формы периодического внешнего возбуждения рассмотрен в работах [19, 20].

Условие существования решения для линейной системы уравнений гармонического баланса с вырожденной матрицей позволило получить алгебраическое уравнение относительно неизвестного сдвига фаз  $\varphi$ :

$$(\omega_0 - \omega_{inj})/\omega_0 = W(B, \varphi), \quad (10)$$

$$\text{где } W(B, \varphi) = \sum_{k,l} B_{kl} V_{kl} e^{jk\varphi}. \quad (11)$$

Здесь  $B_{kl}$ ,  $V_{kl}$  — коэффициенты Фурье внешнего воздействия ( $b(t)$ ) и вектора проекции возмущений ( $v(t)$ );  $k, l$  — индексы гармоник и узла.

Решение уравнения (10) существует, если выполнено условие

$$\min_{\varphi} W(B, \varphi) \leq (\omega_0 - \omega_{inj})/\omega_0 \leq \max_{\varphi} W(B, \varphi). \quad (12)$$

Выражение (12) определяет диапазон захвата автогенератора. Для проверки его справедливости необходимо вычислить  $W(B, \varphi)$  на интервале  $0 \leq \varphi \leq 2\pi$  с достаточно малым шагом [19].

Большинство работ, посвященных анализу синхронизированных генераторов, ограничиваются рассмотрением случаев близости частот возбуждения к собственной частоте генератора. Однако эффект синхронизации может возникать не только при взаимодействии генераторов с близкими частотами, но и в случае, когда отношение частот генераторов близко к рациональной дроби [21], т. е.

$$\omega_{inj} = (p/q)\omega_0 + \Delta\omega, \quad (13)$$

где  $p, q$  — взаимно простые целые числа. В работе [22] получены выражения для анализа режима синхронизации произвольного автогенератора под воздействием слабого внешнего сигнала произвольной формы с частотой вида (13). Обобщение алгебраического уравнения (10) для фазы синхронизированного автогенератора имеет вид

$$\frac{q}{p} \frac{\Delta\omega}{\omega_0} = W_{pq}(B, \theta), \quad (14)$$

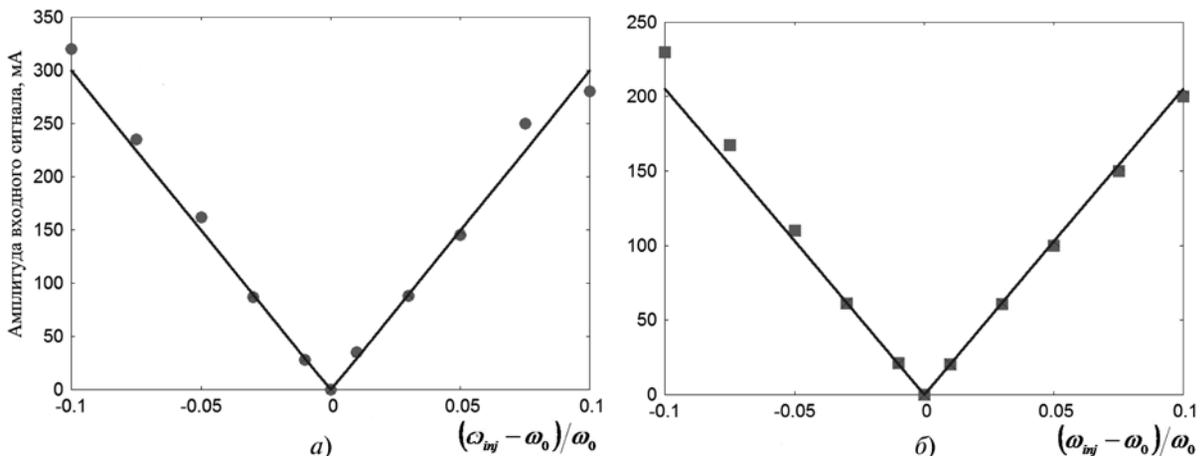


Рис. 2. Область захвата трехкаскадного кольцевого генератора при синусоидальном возбуждении (а), импульсном возбуждении (б)

где  $W_{pq}$  — индексированный вариант функции (11), определенный как

$$W_{pq}(B, \phi) = \sum_{k,l} V_{qk,l} B_{pk,l} \exp(-jk\phi). \quad (15)$$

Выражения (10), (11) являются частным случаем (14), (15) при  $p = q = 1$ .

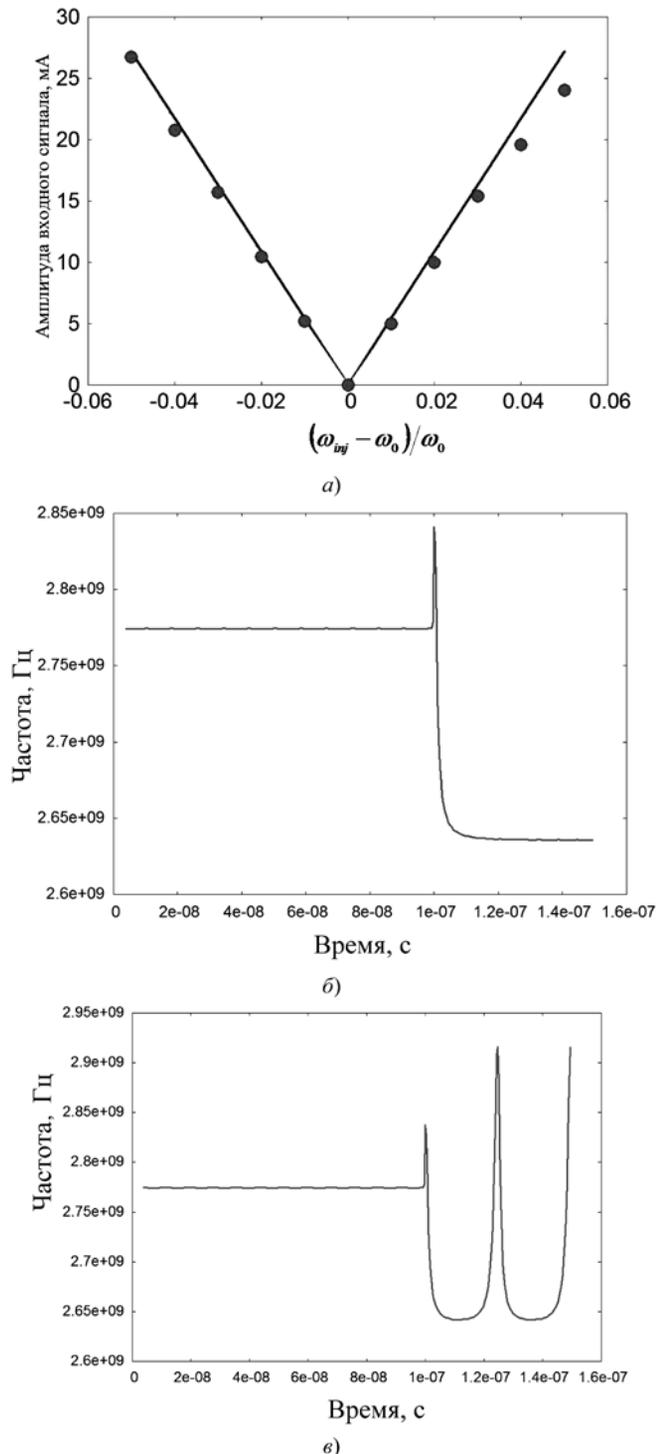


Рис. 3. Область захвата кольцевого БиКМОП-генератора (а), временные зависимости мгновенной частоты для амплитуды 27,5 мА (б) и 25,5 мА (в)

Частота синхронизированного автогенератора в дробном случае равна

$$\omega_{lock} = (q/p)\omega = \omega_0 + (q/p)\Delta\omega,$$

а диапазон захвата определяется из (14) в виде, аналогичном (12):

$$(p/q)\Delta\omega_{min} \leq \Delta\omega \leq (p/q)\Delta\omega_{max}, \quad (16)$$

где  $\Delta\omega_{min} = \omega_0(p/q) \min_{0 \leq \theta < 2\pi} W_{pq}(B, \theta)$ ,  $\Delta\omega_{max} = \omega_0(p/q) \times \max_{0 \leq \theta < 2\pi} W_{pq}(B, \theta)$  задают нижнюю и верхнюю границы отклонения частоты синхронизированного генератора от его собственной частоты:  $\Delta\omega_{min} \leq \omega_{lock} - \omega_0 \leq \Delta\omega_{max}$ .

Сравнение результатов расчетов диапазона захвата по выражению (12) с оценкой на основе моделирования переходных процессов (решением системы (1)) представлено на рис. 2 для схемы трехкаскадного кольцевого генератора. Эксперименты проводились для синусоидального и импульсного (меандр) возбуждений. По оси абсцисс отложено относительное отклонение входной частоты от собственной частоты генератора, по оси ординат — значения амплитуды входного сигнала в миллиамперах (мА). Сплошные линии показывают границы области захвата (язык Арнольда), полученные по формулам (14), (15). Результаты численного моделирования показаны отдельными точками. Проведенные эксперименты подтвердили правильность предложенного подхода для малых амплитуд.

На рис. 3 представлены результаты моделирования двухкаскадного кольцевого БиКМОП-генератора с частотой 2,7 ГГц. На рис. 3, а показана область синхронизации при синусоидальном возбуждении, рассчитанная с помощью полученных выражений для диапазона захвата. Можно заметить, что точка с отклонением по частоте — 5 % и амплитудой 27,5 мА находится в области захвата, а точка с амплитудой 25,5 мА лежит на границе этой области. На рис. 3, б, в представлены временные зависимости мгновенной частоты генератора при воздействии сигналов с амплитудами 27,5 мА и 25,5 мА соответственно, которые показывают что в первом случае генератор переходит в режим синхронизации, а во втором — в режим биений.

### 3. Методы анализа связанных генераторов в режиме взаимной синхронизации

Эффект синхронизации возникает не только при возбуждении отдельного генератора периодическим сигналом, но он может также проявиться в системе нескольких связанных генераторов, когда в результате взаимного влияния в системе возникают периодические колебания с общей частотой. Для проведения анализа взаимной синхронизации наиболее часто использовались методы на основе уравнения

Адлера (8). Такие методы применялись как в случае использования взаимной синхронизации для обеспечения требуемого функционирования проектируемой схемы [21, 23], так и при анализе нежелательной взаимной синхронизации, возникающей вследствие паразитных взаимосвязей в интегральных схемах [24, 25]. Однако имеются следующие ограничения при использовании этих методов:

- предложенные подходы применимы лишь к синусоидальным и слабонелинейным автогенераторам, в то время как паразитная синхронизация может возникать между автогенераторами любых типов;
- взаимодействие генераторов обычно задается постоянным коэффициентом передачи от выхода одного генератора к входу другого; это позволяет анализировать резистивно связанные генераторы [25], но не охватывает случай взаимодействия, представленного линейными цепями с частотно-зависимой передаточной функцией (например, взаимные индуктивности, проходные емкости, цепи питания и "земли").

Указанные ограничения сняты в работе [26], где рассматриваются  $n$  генераторов с близкими собственными частотами  $\omega_j$  и гармониками ВПВ  $V_j$ . Влияние сигналов напряжения  $X_j$   $j$ -го генератора на токи возбуждения  $i$ -го генератора задается частотно-зависимой матрицей комплексных проводимостей  $Y_{ij}(\omega)$ . Показано, что при полной синхронизации всех генераторов значения их фаз  $\varphi_i$  и общей частоты синхронизации  $\omega$  могут быть определены из системы нелинейных алгебраических уравнений вида

$$(\omega - \omega_i)/\omega_1 = \sum_j W^{(i)}(\tilde{Y}^{ij}(\omega_1)X^j, \varphi_i - \varphi_j) \quad (i = 1, \dots, n), \quad (17)$$

где  $\tilde{Y}^{ij}(\omega)$  — блочно-диагональная матрица с блоками  $Y^{ij}(k\omega)$ , а функция определяется выражением (15) с гармониками ВПВ  $V^i$ .

Фазы  $\varphi_j$  входят в систему (17) в виде разностей, поэтому одну из них можно задать произвольно, например  $\varphi_n = 0$ . Система (17) линейно зависит от частоты синхронизации, поэтому величину  $\omega$  из нее можно легко исключить и получить систему порядка  $n - 1$  относительно фаз  $\varphi_j$  ( $j = 1, \dots, n - 1$ ). В частности, для случая двух автогенераторов система (17) сводится к одному уравнению относительно разности фаз генераторов  $\varphi = \varphi_2 - \varphi_1$ :

$$\omega_2 - \omega_1 = \omega_1 W^{(1)}(\tilde{Y}^{1,2}(\omega_1)X^2, \varphi). \quad (18)$$

После нахождения  $\varphi$  из решения (18) значение частоты синхронизации определяется из явного выражения

$$\omega = \omega_1(1 + W^{(1)}(\tilde{Y}^{1,2}(\omega_1)X^2, \varphi)). \quad (19)$$

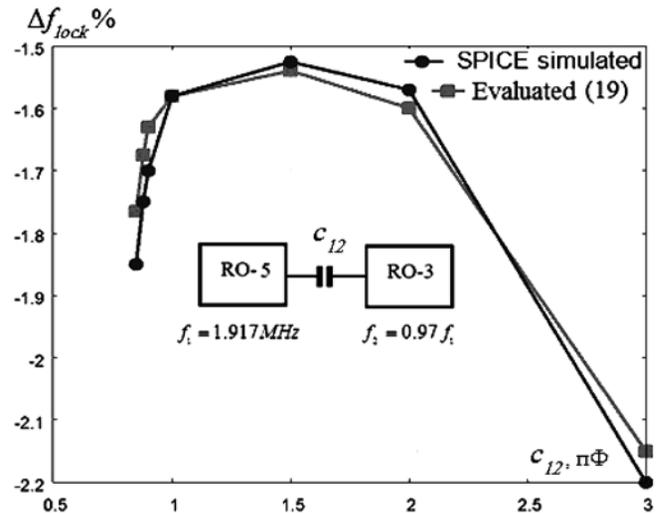


Рис. 4. Зависимости относительного отклонения частоты взаимной синхронизации от значения паразитной емкости, полученные прямым моделированием и оценкой (19)

Сравнение результатов расчетов по выражениям (18), (19) с результатами полного моделирования приведено ниже для случая трехкаскадного и пятикаскадного КМОП-генераторов, связанных паразитной емкостью. Собственные частоты генераторов различаются на 3 % и составляют  $f_1 = 1,917$  МГц и  $f_2 = 1,859$  МГц для пяти- и трехкаскадных генераторов соответственно. В результате взаимной синхронизации устанавливается режим колебаний с частотой  $f_{lock}$ , которая зависит от значения паразитной емкости. На рис. 4 показаны зависимости относительного отклонения частоты взаимной синхронизации  $\Delta f_{lock} = \frac{f_{lock} - f_1}{f_1} 100 \%$  от значения

паразитной емкости, полученные по выражениям (18), (19) и с помощью полного моделирования.

Большинство работ, посвященных анализу синхронизированных генераторов, ограничиваются рассмотрением случаев близости частоты возбуждения к собственной частоте генератора. Однако аналогично генераторам с внешней синхронизацией эффект взаимной синхронизации может возникать не только при взаимодействии генераторов с близкими частотами, но и при соотношении частот генераторов, близком к рациональной дроби [21]. Этот случай проанализирован в работе [27], где рассматривается синхронизация взаимосвязанных генераторов, для которых заданы дробные соотношения частот первого и остальных генераторов.

## Заключение

Анализ состояния разработок современных вычислительных методов расчета периодических процессов в автогенераторных электронных схемах позволяет определить наиболее перспективные вы-

числительные стратегии для систем автоматизации схемотехнического моделирования.

К наиболее универсальным методам моделирования автономного генератора в частотной области можно отнести метод продолжения на основе использования искусственного зонда с подстройкой частоты. Этот подход обеспечивает высокую надежность сходимости процесса к невырожденному решению при минимальных требованиях к информации, задаваемой пользователем.

Для определения периодических процессов синхронизации в автогенераторах при малом периодическом возбуждении к настоящему времени разработаны универсальные подходы, позволяющие анализировать поведение нелинейных генераторов в режиме синхронизации периодическим сигналом произвольной формы. При этом имеется возможность анализировать не только случаи близости частоты возбуждения к собственной частоте генератора, но и случаи соотношения этих частот, близких к рациональной дроби.

Вычислительный алгоритм для задачи моделирования процесса взаимной синхронизации группы связанных генераторов может быть основан на решении системы алгебраических уравнений относительно фаз генераторов и неизвестной частоты синхронизации. Алгоритм распространяется на случай дробного соотношения собственных частот генераторов.

#### Список литературы

1. **Kundert K. S., White J., Sangiovanni-Vincentelli A.** Steady-State Methods for Simulating Analog and Microwave Circuits. Boston: Kluwer Academic Publishers, 1990. 247 p.
2. **Актуальные** проблемы моделирования в системах автоматизации схемотехнического проектирования / Под ред. А. Л. Стемпковского. М.: Наука, 2003. 430 с.
3. **Rizzoli V., Neri A.** State of the art and present trends in nonlinear microwave CAD techniques // IEEE Trans. Microwave Theory and Technique. 1988. Vol. 36. P. 343–364.
4. **Rizzoli V., Costanzo A., Neri A.** Harmonic-balance analysis of microwave oscillators with automatic suppression of degenerate solution // Electronics Letters. 1992. Vol. 28, N. 3. P. 256–257.
5. **Ngoya E., Suarez A., Sommet R., Quer R.** Steady-state analysis of free or forced oscillators by harmonic balance and stability investigation of periodic and quasi-periodic regimes // Int. J. of Microwave and Millimeter-Wave Comp. Aided Eng. 1995. Vol. 5, N. 3. P. 210–223.
6. **Chang C.-R., Steer M. B.** Computer-Aided analysis of free-running microwave oscillators // IEEE Trans. on Microwave Theory and Techniques. 1991. Vol. 39, N. 10. P. 1735–1745.
7. **Lampe S., Laur R.** Global optimization applied to the oscillator problems // Proc. of Design Automation and Test in Europe Conf. 2002. P. 322–326.
8. **Duan X., Mayaram K.** Frequency domain simulation of high-Q oscillators with homotopy methods // Proc. of IEEE/ACM Int. Conf. on Computer-Aided Design. 2004. P. 683–686.

9. **Boianapally K., Mei T., Roychowdhury J.** A multi-harmonic probe technique for computing oscillator steady states // Proc. IEEE/ICM Int. Conf. Comput. Aided Des. 2005. P. 609–612.
10. **Brambilla A., Gruosso G., Gajani G.** Robust harmonic-probe method for the simulation of oscillators // IEEE Trans. on Circuits and Systems I: Regular Papers. 2010. Vol. 57, N. 9. P. 2531–2541.
11. **Gourary M., Ulyanov S., Zharov M., Rusakov S.** et al. Simulation of high-Q oscillators // Proc. of IEEE/ACM Int. Conf. on Computer-Aided Design. 1998. P. 162–169.
12. **Gourary M., Ulyanov S., Zharov M., Rusakov S.** et al. A robust and efficient oscillator analysis technique using harmonic balance // Computer Methods in Applied Mechanics and Engineering. 2000. Vol. 181. P. 451–466.
13. **Gourary M., Rusakov S., Ulyanov S., Zharov M.** et al. Frequency adjusting numerical technique for oscillator simulation // Proc. European Conf. on Circuit Theory and Design. Cork, Ireland. 2005. Vol. 3. P. 121–124.
14. **Adler R.** A study of locking phenomena in oscillators // Proc. of the IRE and Waves and Electrons. June 1946. N. 34. P. 351–357.
15. **Demir A., Mehrotra A., Roychowdhury J.** Phase noise in oscillators: a unifying theory and numerical methods for characterization // IEEE Trans. on Circuits and Systems — I. 2000. Vol. 47. P. 655–674.
16. **Demir A., Long P., Roychowdhury J.** Computing phase noise eigenfunctions directly from steady-state Jacobian matrices // IEEE/ACM Int. Conf. on CAD. San Jose, 2000. P. 283–288.
17. **Vanassche P., Gielen G., Sansen W.** On the difference between two widely publicized methods for analysing oscillator noise behavior // Proc. Int. Conf. Computer Aided Design. 2002. P. 229–233.
18. **Lai X., Roychowdhury J.** Automated oscillator macromodeling techniques for capturing amplitude variations and injection locking // Proc. Int. Conf. Computer Aided Design. 2004. P. 687–694.
19. **Gourary M. M., Rusakov S. G., Ulyanov S. L., Zharov M. M., Mulvaney B. J.** Analysis of oscillator injection locking by harmonic balance method // Proc. of Design Automation and Test in Europe Conf. 2008. P. 318–323.
20. **Гурарий М. М., Ульянов С. Л.** Анализ условий синхронизации автогенератора // Изв. вузов. Электроника. 2009. № 5 (79). С. 57–65.
21. **Боголюбов Н. Н., Митропольский Ю. А.** Асимптотические методы в теории нелинейных колебаний. Изд. 2-е испр. и доп. М.: Гос. изд-во физико-математ. лит., 1958. 408 с.
22. **Гурарий М. М., Жаров М. М., Русаков С. Г., Ульянов С. Л.** Метод анализа режимов синхронизации и биений автогенератора при паразитном возбуждении сигналом произвольной формы и частоты // Изв. вузов. Электроника. 2013. № 3 (101). С. 73–82.
23. **York R. A.** Injection- and phase-locking techniques for beam control // IEEE Trans. Microw. Tech. 1998. Vol. 46, N. 11. P. 1920–1929.
24. **Bonnin M., Corinto F.** Periodic oscillations in weakly connected cellular nonlinear networks // IEEE Trans. Circ. Syst. — I. 2008. Vol. 55, N. 6. P. 1671–1684.
25. **Maffezzoni P.** Synchronization analysis of two weakly coupled oscillators through a PPV macromodels // IEEE Trans. Circ. Syst. — I. 2010. Vol. 57, N. 3. P. 654–663.
26. **Gourary M. M., Rusakov S. G., Ulyanov S. L., Zharov M. M.** Mutual injection locking of oscillators under parasitic couplings // Scientific Computing in Electrical Engineering, Book series: Mathematics in Industry. 2012. Vol. 16. P. 303–312.
27. **Гурарий М. М., Русаков С. Г., Ульянов С. Л.** Разработка методов анализа режима взаимной синхронизации автогенераторов в интегральных схемах // Проблемы разработки перспективных микро- и наноэлектронных систем: Сб. трудов / под ред. акад. РАН А. Л. Стемпковского. М.: ИППМ РАН, 2010. С. 138–143.

## Frequency Domain Simulation Methods of Periodic States of Oscillators

The modern computational methods for determining periodic steady state and analyzing oscillators are considered. The methods are intended to be implemented in circuit simulators, enable to analyze arbitrary oscillators and provide the computation of characteristics under external excitation of arbitrary form.

**Keywords:** circuit simulation, oscillators, periodic steady state, harmonic balance, artificial probe method, injection locking

### References

1. **Kundert K. S., White J., Sangiovanni-Vincentelli A.** Steady-State Methods for Simulating Analog and Microwave Circuits — Boston: Kluwer Academic Publishers, 1990. 247 p.
2. **Aktual'nye problemy modelirovaniya v sistemah avtomatizacii shemotehnicheskogo proektirovaniya / Ed. A. L. Stempkovskij.** M.: Nauka, 2003. 430 p.
3. **Rizzoli V., Neri A.** State of the art and present trends in non-linear microwave CAD techniques. *IEEE Trans. Microwave Theory and Technique.* 1988. Vol. 36. N. 2. P. 343—364.
4. **Rizzoli V., Costanzo A., Neri A.** Harmonic-balance analysis of microwave oscillators with automatic suppression of degenerate solution. *Electronics Letters.* 1992. Vol. 28. N. 3. P. 256—257.
5. **Ngoya E., Suarez A., Sommet R., Quer R.** Steady-state analysis of free or forced oscillators by harmonic balance and stability investigation of periodic and quasi-periodic regimes. *Int. J. of Microwave and Millimeter-Wave Comp. — Aided Eng.* 1995. Vol. 5. N. 3. P. 210—223.
6. **Chang C.-R., Steer M. B.** Computer-Aided analysis of free-running microwave oscillators. *IEEE Trans. on Microwave Theory and Techniques.* 1991. Vol. 39, N. 10. P. 1735—1745.
7. **Lampe S., Laur R.** Global optimization applied to the oscillator problems. *Proc. of Design Automation and Test in Europe Conf.* 2002. P. 322—326.
8. **Duan X., Mayaram K.** Frequency domain simulation of high-Q oscillators with homotopy methods. *Proc. of IEEE/ACM Int. Conf on Computer-Aided Design.* 2004. P. 683—686.
9. **Boianapally K., Mei T., Roychowdhury J.** A multi-harmonic probe technique for computing oscillator steady states. *Proc. IEEE/ICM Int. Conf Comput. — Aided Des.* 2005. P. 609—612.
10. **Brambilla A., Grusso G., Gajani G.** Robust harmonic-probe method for the simulation of oscillators. *IEEE Trans. on Circuits and Systems I: Regular Papers.* 2010. Vol. 57, N. 9. P. 2531—2541.
11. **Gourary M., Ulyanov S., Zharov M., Rusakov S.** et al. Simulation of high-Q oscillators. *Proc. of IEEE/ACM Int. Conf. on Computer-Aided Design.* 1998. P. 162—169.
12. **Gourary M., Ulyanov S., Zharov M., Rusakov S.** et al. A robust and efficient oscillator analysis technique using harmonic balance. *Computer Methods in Applied Mechanics and Engineering.* 2000. Vol. 181. P. 451—466.
13. **Gourary M., Rusakov S., Ulyanov S., Zharov M.** et al., Frequency adjusting numerical technique for oscillator simulation. *Proc. European Conf on Circuit Theory and Design. Cork, Ireland.* 2005. V. 3. P. 121—124.
14. **Adler R.** A study of locking phenomena in oscillators. *Proc. of the IRE and Waves and Electrons.* June 1946. N. 34. P. 351—357.
15. **Demir A., Mehrotra A., Roychowdhury J.** Phase noise in oscillators: a unifying theory and numerical methods for characterization. *IEEE Trans. on Circuits and Systems — I.* 2000. Vol. 47. P. 655—674.
16. **Demir A., Long P., Roychowdhury J.** Computing phase noise eigenfunctions directly from steady-state Jacobian matrices. *IEEE/ACM Int. Conf on CAD.* San Jose, 2000. P. 283—288.
17. **Vanassche P., Gielen G., Sansen W.** On the difference between two widely publicized methods for analysing oscillator noise behavior. *Proc. Int. Conf. Computer Aided Design.* 2002. P. 229—233.
18. **Lai X., Roychowdhury J.** Automated oscillator macromodeling techniques for capturing amplitude variations and injection locking. *Proc. Int. Conf Computer Aided Design.* 2004. P. 687—694.
19. **Gourary M. M., Rusakov S. G., Ulyanov S. L., Zharov M. M., Mulvaney B. J.** Analysis of oscillator injection locking by harmonic balance method. *Proc. of Design Automation and Test in Europe Conf.* 2008. P. 318—323.
20. **Gourary M. M., Ulyanov S. L.** Analiz uslovij sinhronizacii avtogeneratora. *Izv. vuzov. Jelektronika.* 2009. N. 5 (79). P. 57—65.
21. **Bogoljubov N. N., Mitropol'skij Ju. A.** Asimptoticheskie metody v teorii nelinejnyh kolebanij. Izd. 2-e ispr. i dop. M.: Gos. izd-vo fiziko-matemat. lit, 1958. 408 p.
22. **Gourary M. M., Zharov M. M., Rusakov S. G., Ulyanov S. L.** Metod analiza rezhimov sinhronizacii i bienij avtogeneratora pri parazitnom vzbuzhdenii signalom proizvol'noj formy i chastoty. *Izv. vuzov. Jelektronika.* 2013. N. 3 (101). P. 73—82.
23. **York R. A.** Injection- and phase-locking techniques for beam control. *IEEE Trans. Microw. Tech.* 1998. Vol. 46, N. 11. P. 1920—1929.
24. **Bonnin M., Corinto F.** Periodic oscillations in weakly connected cellular nonlinear networks. *IEEE Trans. Circ. Syst. — I.* 2008. Vol. 55, N. 6. P. 1671—1684.
25. **Maffezzoni P.** Synchronization analysis of two weakly coupled oscillators through a PPV macromodels. *IEEE Trans. Circ. Syst. — I.* 2010. Vol. 57, N. 3. P. 654—663.
26. **Gourary M. M., Rusakov S. G., Ulyanov S. L., Zharov M. M.** Mutual injection locking of oscillators under parasitic couplings. *Scientific Computing in Electrical Engineering, Book series: Mathematics in Industry,* 2012. Vol. 16. P. 303—312.
27. **Gourary M. M., Rusakov S. G., Ulyanov S. L.** Razrabotka metodov analiza rezhima vzaimnoj sinhronizacii avtogeneratorov v integral'nyh shemah. *Problemy razrabotki perspektivnyh mikro- i nanojelektronnyh sistem / Ed. akad. RAN A. L. Stempkovskij.* M.: IPPM RAN, 2010. P. 138—143.

# ОБРАБОТКА ИЗОБРАЖЕНИЙ IMAGE PROCESSING

УДК 004.93

**К. С. Маркелов**, аспирант, e-mail: kosmar89@mail.ru,

**В. В. Нечаев**, д-р физ.-мат. наук, проф., зав. каф., e-mail: nechaev@mirea.ru,

Московский государственный технический университет радиотехники, электроники и автоматики

## Алгоритм повышения качества цифровых изображений дактилоскопической информации

*Рассматриваются вопросы обработки изображений в целях увеличения разрешающей способности цифрового изображения на базе метода суперразрешения. Разработан алгоритм повышения информативности цифрового изображения, который включает предобработку изображения с переводом в градации серого цвета, нахождение межкадрового движения методом Лукаса—Канаде, с последующим усреднением в один кадр высокого разрешения и постобработку цифрового изображения в целях устранения размытости и повышения резкости. Предлагаемый алгоритм инвариантен к конкретным задачам и может использоваться в различных областях знаний, в частности — в биометрии.*

**Ключевые слова:** цифровое изображение, разрешение, суперразрешение, метод Лукаса—Канаде, биометрия, дактилоскопия

### Введение

Задачи обработки цифровых изображений находят широкое практическое применение в различных областях инженерно-технической деятельности. Особую актуальность они приобретают в тех случаях, когда обрабатываемые изображения изначально имеют низкое качество. Такого рода изображения возникают из-за недостаточной разрешающей способности фото/видеокамер, сканеров или иных технических средств, а также из-за недостаточности корректного их применения. Повышение качества изображений зачастую связано с использованием дорогостоящего оборудования. Однако такой подход далеко не всегда оправдан. Например, использование высокоточных профессиональных дорогостоящих сканеров отпечатков пальцев в общедоступных местах без постоянного контроля за таким оборудованием не всегда оправдано. Так же, как и применение систем фото/видеофиксации с высоким разрешением в самолетах нецелесообразно в силу специфики оборудования таких камер, которые должны работать в сложных климатических условиях и при перегрузках при полете. Устанавливать дорогостоящее оборудование на самолеты не представляется возможным и по ряду других причин. Имеется достаточно много и других примеров целесообразности и необходимости использования цифровых изображений высокого разрешения.

Повышение уровня разрешения цифровых изображений возможно осуществлять аппаратными, программными и аппаратно-программными мето-

дами и средствами. Развитие современной микроэлектроники создает возможности успешно применять программные методы увеличения разрешения. В предлагаемой вниманию читателя статье рассматривается один из возможных алгоритмов решения такого рода задач.

### 1. Качественный анализ изображений как исходных данных для программной обработки

Для разработки алгоритма необходимо определить, какие исходные данные будут обрабатываться в программно-алгоритмическом модуле по улучшению и повышению информативности изображения. В нашем случае повышение информативности изображений — это улучшение и повышение разрешения и качества изображений, т. е. повышение видимости объектов или иной информации на цифровом изображении. В качестве обрабатываемых цифровых изображений могут выступать полученные различными способами кадры, например, сканируемые отпечатки пальцев человека, изображения фото/видеофиксации бортовыми системами взлетно-посадочной полосы и т. п. Степень важности корректного получения информации с таких цифровых изображений очевидна, так как неправильно или ошибочно распознанное изображение отпечатка пальца может повлиять на судьбу или даже жизнь человека, а незамеченные на цифровом изображении препятствия на взлетно-посадочной полосе могут быть причиной авиакатастрофы. Таким образом, даже если имеются исходные изображения

подобного рода объектов, но для принятия решений их качество недостаточно, возникает необходимость его повышения. Отметим, что для обработки таких изображений, в силу приведенных во введении причин, целесообразно использовать программные методы и средства.

Традиционно для решения подобного рода задач в качестве инструментальных средств используются проблемно ориентированные программные модули. На вход такого программного модуля поступает последовательность кадров (изображений) низкого разрешения, например, одного и то же отпечатка пальца, снятого со сканера несколько раз. Последовательности кадров могут иметь низкое разрешение вследствие разного рода причин: например, для цифровых изображений отпечатков пальцев из-за недостаточно хорошего сканера, плохих условий снятия отпечатков (влажность, температура); для цифровых изображений взлетно-посадочной полосы при использовании камер низкого разрешения, плохих погодных условий (низкая видимость, туман, дождь, ночное время суток). При проектировании программного модуля обработки изображений необходимо учитывать возможность работы с кадрами различных размеров, например,  $100 \times 100$  или  $720 \times 576$  и т. д.

Для визуальной иллюстрации изложенного выше, приведем примеры последовательностей исходных кадров низкого разрешения. Случай, представленный на рис. 1, — это последовательность изображений одного и того же отпечатка папиллярного узора пальца, снятого сканером.

Случай, представленный на рис. 2 (см. вторую сторону обложки), — это пример последовательности исходных кадров одной и той же сцены взлетно-посадочной полосы, снятой видеокамерой с самолета.

Принимая во внимание рассмотренные выше факторы, влияющие на качество изображений, определим ограничения, накладываемые на исходные видеокadres:

- кадры изображений должны иметь некоторые смещения по осям  $x$  и  $y$ . Такие смещения должны быть небольшими, т. е. учитываются вибрации и малые смещения самого прибора — сканера отпечатков пальцев;
- изображения в последовательности кадров, которые формируются в процессе фотографирования или сканирования отпечатка пальца, при приближении, смене сцены, появлении на картине новых объектов, не являются исходными в разрабатываемом программном модуле.



Рис. 1. Исходная последовательность кадров низкого разрешения одного отпечатка пальцев

## 2. Метод суперразрешения

Прежде всего отметим, что *суперразрешение* определяется как процесс реконструкции изображения высокого разрешения по нескольким изображениям низкого разрешения [1]. Использование нескольких изображений низкого разрешения дает возможность на основе метода суперразрешения сформировать изображения высокого разрешения, т. е. повысить их качество. Для реализации метода суперразрешения необходимо в качестве исходных данных для программной обработки использовать несколько кадров. Общий алгоритм работы программного модуля по повышению информативности изображения, а соответственно, и улучшения качества изображения, формируется согласно модели, описанной в работе [2]. Он включает следующие этапы (рис. 3):

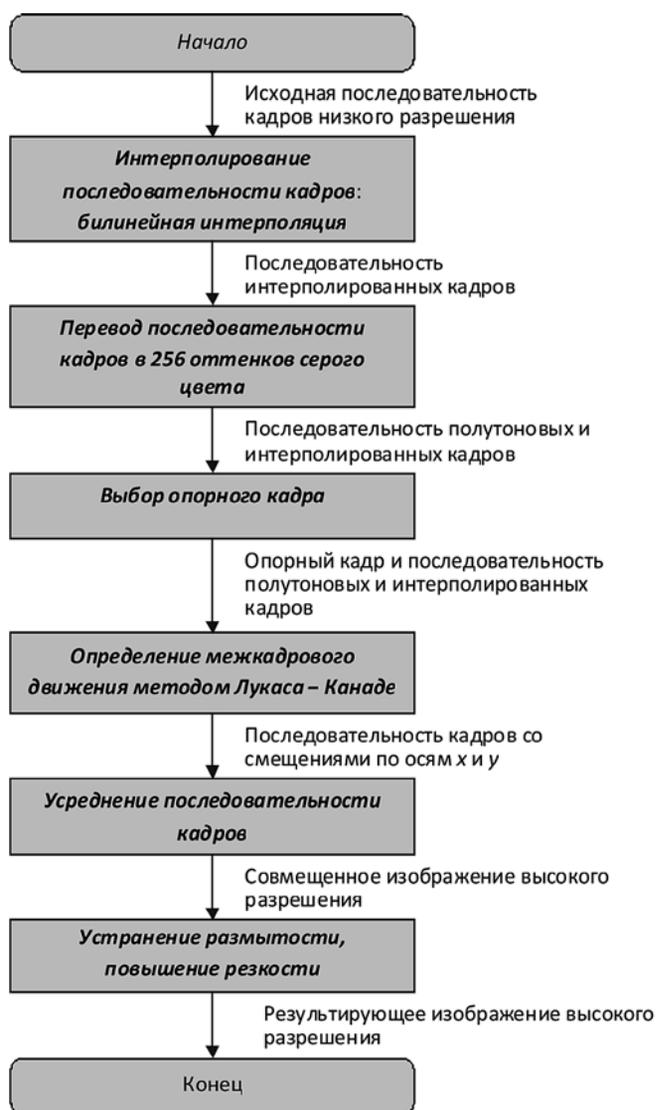


Рис. 3. Схема алгоритма повышения информативности изображений методом суперразрешения

1) интерполирование последовательности кадров в целях повышения их масштаба: билинейная интерполяция (на выходе — последовательность интерполированных кадров);

2) перевод последовательности кадров в 256 оттенков серого цвета, т. е. получение на выходе полутоновых изображений;

3) выбор опорного кадра (на выходе — последовательность кадров и опорный кадр); программа должна распознать, какой кадр является опорным — основным для сравнения и работы дальнейших этапов алгоритма. В программном модуле необходимо предусмотреть выбор опорного кадра вручную и автоматически программой;

4) определение методом Лукаса—Канаде межкадрового движения (на выходе — смещение кадров относительно опорного по осям  $x$  и  $y$ );

5) совмещение кадров или усреднение последовательности кадров (на выходе — одно совмещенное изображение высокого разрешения);

6) устранение размытости, повышение резкости (на выходе результирующее изображение высокого разрешения с применением фильтра или маски).

Таким образом, первые три этапа относятся к предобработке исходной последовательности кадров низкого разрешения. Этапы 4 и 5 являются ключевыми, результаты их выполнения дают возможность формировать результирующее изображение высокого разрешения. Шестой этап ориентирован на постобработку изображения.

## 2.1. Предобработка последовательности кадров

### 2.1.1. Интерполирование последовательности кадров: билинейная интерполяция

В компьютере изображение представлено конечным числом дискретных точек. Масштабирование цифровых изображений связано с приведением массива информации в соответствие с разрешением и размером иллюстрации. В зависимости от цели этот массив информации необходимо либо увеличить, либо сократить. Простое размножение или сокращение не является сложной задачей для компьютерных систем. Однако при этом возможны существенные искажения геометрии мелких деталей и появление ложных узоров на текстурах. Чтобы эти явления при трансформации были минимальны, необходимо использовать *интерполяционные алгоритмы*.

Для того чтобы сделать изображение больше при том же разрешении, в него нужно добавить новые пиксели. Задача состоит в том, чтобы рассчитать цвет новых пикселей, которые необходимо добавить между существующими.

*Интерполяция* — это пространственный (или временной) прогноз значений неизвестных значений пикселей среди истинных значений пикселей.

Возможна следующая интерполяция изображений:

- изображение представляется в виде функции;
- пиксели изображений являются точками, в которых значение функции известно.

Билинейная интерполяция рассматривает квадрат  $2 \times 2$  известных пикселей, окружающих неизвестный. В качестве интерполированного значения используется взвешенное усреднение этих четырех пикселей. В результате изображения выглядят значительно более гладко, чем результат работы метода ближайшего соседа.

### Алгоритм билинейной интерполяции

Для хранения кадров (изображений) необходимо использовать некоторый массив — хранилище интенсивностей каждого изображения.

*На вход* алгоритма поступает последовательность кадров (изображения отпечатка пальцев) и коэффициенты масштабирования по осям  $x$  и  $y$ . Оси  $x$  и  $y$  — это направления расположений интенсивности на изображении. Коэффициенты масштабирования могут быть как одинаковыми (для обеспечения масштабирования изображений в равных пропорциях), так и различаться по осям  $x$  и  $y$ .

Циклом будем называть проход по каждому пикселю на изображении по осям  $x$  и  $y$ : от  $i$  до  $h$  по оси  $x$ , от  $j$  до  $w$  по оси  $y$ , где

$i$  — начальное значение (по умолчанию равно 0) прохода по оси  $x$ ;

$h$  — конечное значение (по умолчанию равно высоте кадра) прохода по оси  $x$ ;

$j$  — начальное значение (по умолчанию равно 0) прохода по оси  $y$ ;

$w$  — конечное значение (по умолчанию равно ширине кадра) прохода по оси  $y$ .

Каждый кадр обрабатывается одним и тем же алгоритмом — алгоритмом билинейной интерполяции. Основные шаги алгоритма определены ниже.

**Шаг 1.** Выделение памяти для массива яркости исходного кадра.

**Шаг 2.** Цикл с заполнением массива яркости исходного кадра.

**Шаг 3.** Создание нового массива с новой высотой и шириной для нового масштабированного изображения.

**Шаг 4.** Цикл от  $i = 0$  до  $h - 1$  по оси  $x$ , от  $j = 0$  до  $w - 1$  по оси  $y$  нового массива с занесением в него новых яркостей (интенсивностей).

4.1. Цикл от  $i = 0$  до  $h - 2$  по оси  $x$ , от  $j = 0$  до  $w - 2$  по оси  $y$  с заполнением яркости по отдельным составляющим RGB четырех соседних пикселей в массив.

4.2. Цикл от  $i = 0$  до 2 по оси  $x$ , от  $j = 0$  до 2 по оси  $y$  с вычислением яркости в точке  $(x, y)$  по формуле интерполяции.

*На выходе* — новый кадр в увеличенном масштабе.

### 2.1.2. Перевод изображений в полутоновые изображения

Полутоновое изображение определяется имеющимся множеством значений тона, в данном случае — это 256 оттенков серого цвета и их непрерывное, плавное изменение. Как и все растровые изображения, полутоновое изображение кодируется в цифровом виде с помощью битовой карты (матрицы, хранящей значения элементов изображения (пикселей)). Каждый пиксель полутонового изображения может кодироваться различным числом битов, определяющим число возможных полутонов.

Проиллюстрируем данный подход на примере:  $2^2$  бит задает 4 полутона,  $2^3 = 8$ ,  $2^4 = 16$ , а  $2^8 = 256$  соответственно 256 полутонов.

При этом однобитовое бинарное изображение (1 бит на 1 пиксель) можно считать вырожденным полутоновым, способным передать лишь два полутона (например, черный и белый), или же его частным случаем.

Множество возможных полутонов называют уровнями серого (англ. *gray scale*), независимо от того, полутона какого цвета или его оттенка передаются. Таким образом, уровни серого не различаются по спектральному составу (оттенку цвета), но различаются по яркости. Число возможных полутонов в данном случае есть глубина цвета, которую часто передают не числом самих полутонов, а числом битов на пиксель. Какое из значений в допустимом диапазоне будет считаться самым ярким, а какое самым темным, не имеет значения, так как число, являющееся значением каждого пикселя — всего лишь условный код яркости. Например, могут существовать полутоновые растры, где на каждый пиксель отведено 8 бит, изображение имеет 256 полутонов, а пиксели со значением 0 или 255 будут являться черными, и наоборот, пиксели со значением 255 или 0 — белыми, а остальные полутона серого будут равномерно распределены между данными значениями цветового индекса. Любое изображение в цифровом формате представляет собой три двумерных массива по каналам RGB (где значения соответствуют яркости компонента) или может быть приведено к таковому [2]. Таким образом, возникает возможность обрабатывать не само изображение, а его представление одной двухмерной матрицей, в которой значения от 0 до 255 — яркость точки. Фактически это соответствует серому изображению с глубиной яркости 8 бит. Яркость точки определяется по формуле

$$Y = 0,3R + 0,59G + 0,11B. \quad (1)$$

#### Алгоритма перевода изображения в "серое"

Начало работы определяется поступлением на вход алгоритма цветного кадра (массива). Далее выполняются следующие шаги.

**Шаг 1.** Цикл от  $i = 0$  до  $h - 1$  по оси  $x$ , от  $j = 0$  до  $w - 1$  по оси  $y$  исходного кадра.

1.1. Выделение составляющих цвета RGB, суммирование и деление результата на 3.

1.2. Присваивание составляющим цвета RGB получившегося цвета.

На выходе — новый полутоновой кадр.

Каждый кадр последовательности обрабатывается одним и тем же алгоритмом.

### 2.1.3. Выбор опорного кадра

Для выбора опорного кадра необходимо на вход модуля подать все загруженные кадры, а на его выходе получить выбранный опорный кадр и последовательность остальных кадров.

В алгоритме выбора опорного кадра из их последовательности должна быть учтена возможность выбора кадра пользователем, т. е. выбор может быть сделан либо программой, либо пользователем. При программном методе выбирается самый четкий кадр с высокой степенью резкости.

## 2.2. Получение изображения высокого разрешения

### 2.2.1. Определение межкадрового движения методом Лукаса—Канаде

**Алгоритм Лукаса—Канаде** — дифференциальный локальный метод вычисления оптического потока, широко используемый в компьютерном зрении. Оптический поток — это изображение видимого движения объектов, поверхностей или краев сцены, получаемое в результате перемещения наблюдателя (глаз или камеры) относительно сцены. Основное уравнение оптического потока содержит два неизвестных и не может быть однозначно разрешено. Алгоритм Лукаса—Канаде обходит неоднозначность за счет использования информации о соседних пикселях в каждой точке. Метод основан на предположении, что в локальной окрестности каждого пикселя значение оптического потока одинаково.

**Описание алгоритма.** Предположим, что смещение пикселей между двумя кадрами невелико. Рассмотрим пиксель  $p$ , тогда по алгоритму Лукаса—Канаде оптический поток должен быть одинаков для всех пикселей, находящихся в окне с центром в  $p$ . А именно, вектор оптического потока  $(V_x, V_y)$  в точке  $p$  должен быть решением системы уравнений

$$\begin{aligned} I_x(q_1)V_x + I_y(q_1)V_y &= -I_t(q_1), \\ I_x(q_2)V_x + I_y(q_2)V_y &= -I_t(q_2), \\ &\vdots \\ I_x(q_n)V_x + I_y(q_n)V_y &= -I_t(q_n), \end{aligned} \quad (2)$$

где  $q_1, q_2, \dots, q_n$  — пиксели внутри окна;  $I_x(q_i), I_y(q_i), I_t(q_i)$  — частные производные изображения  $I$  по координатам  $x, y$  и времени  $t$ , вычисленные в

точке  $q_i$ . Вычисляют эти производные следующим образом [3, 4]:

$$\begin{aligned} I_x(x, y) &= \frac{\partial I(x, y)}{\partial x} = \frac{\partial I(x+1, y) - \partial I(x-1, y)}{2}; \\ I_y(x, y) &= \frac{\partial I(x, y)}{\partial y} = \frac{\partial I(x, y+1) - \partial I(x, y-1)}{2}; \\ I_t(x, y) &= \frac{\partial I(x, y) - \partial z J(x, y)}{2}, \end{aligned} \quad (3)$$

где  $I, J$  — исходный кадр и новый кадр соответственно. Это уравнение может быть записано в матричной форме  $Av = b$ , где

$$A = \begin{bmatrix} I_x(q_1) & I_y(q_1) \\ I_x(q_2) & I_y(q_2) \\ \vdots & \vdots \\ I_x(q_n) & I_y(q_n) \end{bmatrix}, \quad v = \begin{bmatrix} V_x \\ V_y \end{bmatrix}, \quad b = \begin{bmatrix} -I_t(q_1) \\ -I_t(q_2) \\ \vdots \\ -I_t(q_n) \end{bmatrix}. \quad (4)$$

Полученную переопределенную систему решаем с помощью метода наименьших квадратов. Таким образом, получается система уравнений  $2 \times 2$

$$A^T A v = A^T b$$

или

$$v = (A^T A)^{-1} A^T b, \quad (5)$$

где  $A^T$  — транспонированная матрица  $A$ . Таким образом, в соответствии с [3, 4] получаем:

$$\begin{aligned} \begin{bmatrix} V_x \\ V_y \end{bmatrix} &= \\ &= \begin{bmatrix} \sum_i I_x(q_i)^2 & \sum_i I_x(q_i) I_y(q_i) \\ \sum_i I_x(q_i) I_y(q_i) & \sum_i I_y(q_i)^2 \end{bmatrix}^{-1} \begin{bmatrix} -\sum_i I_x(q_i) I_t(q_i) \\ -\sum_i I_y(q_i) I_t(q_i) \end{bmatrix}. \end{aligned} \quad (6)$$

Следующий шаг — рассмотрение разрешимости задачи при поиске межкадрового движения. При этом необходимо выполнение следующих условий:

- матрица  $A^T A$  должна быть обратимой;
- матрица  $A^T A$  не должна стремиться к нулю.

Метод Лукаса—Канаде обладает высокой субпиксельной точностью.

Недостатками метода являются существенное возрастание ошибки при увеличении межкадровых движений и низкое быстродействие на целочисленных сигнальных процессорах. При выводах по применению и накладываемым ограничениям на алгоритм метода Лукаса—Канаде необходимо учитывать в разрабатываемом алгоритме следующие предположения:

- матрица  $A^T A$  обратима;
- имеет место низкая зашумленность кадровых изображений;
- яркость точки должна быть непостоянной (нет резкой смены освещения);
- движение между кадрами незначительное.

На вход алгоритма поступают опорный кадр и кадры-массивы. Основные этапы реализации алгоритма по методу Лукаса—Канаде следующие:

**Шаг 1.** Выделение памяти для вспомогательных массивов яркостей изображений, массивов вычисления производных.

**Шаг 2.** Цикл от  $i = 0$  до  $h - 1$  по оси  $x$ , от  $j = 0$  до  $w - 1$  по оси  $y$  опорного кадра.

2.1. Заполняем массив яркостями по всему исходному изображению.

**Шаг 3.** Цикл от  $i = 0$  до  $h - 1$  по оси  $x$ , от  $j = 0$  до  $w - 1$  по оси  $y$  опорного кадра.

3.1. Вычисление производной по  $y$  и занесение в массив в каждой точке.

**Шаг 4.** Цикл от  $i = 0$  до  $h - 1$  по оси  $x$ , от  $j = 0$  до  $w - 1$  по оси  $y$ .

4.1. Вычисление суммы квадратов производных по  $y$ .

**Шаг 5.** Цикл от  $i = 0$  до  $h - 1$  по оси  $x$ , от  $j = 0$  до  $w - 1$  по оси  $y$  опорного кадра.

5.1. Вычисление производной по  $x$  и занесение ее в массив в каждой точке.

**Шаг 6.** Цикл от  $i = 0$  до  $h - 1$  по оси  $x$ , от  $j = 0$  до  $w - 1$  по оси  $y$ .

6.1. Вычисление суммы квадратов производных по  $x$ .

**Шаг 7.** Цикл от  $i = 0$  до  $h - 1$  по оси  $x$ , от  $j = 0$  до  $w - 1$  по оси  $y$ .

7.1. Вычисление суммы производных по  $y$  и  $x$ .

**Шаг 8.** Цикл от  $i = 0$  до  $h - 1$  по оси  $x$ , от  $j = 0$  до  $w - 1$  по оси  $y$  исходного кадра.

8.1. Заполнение массива яркостями по всему исходному изображению.

**Шаг 9.** Цикл от  $i = 0$  до  $h - 1$  по оси  $x$ , от  $j = 0$  до  $w - 1$  по оси  $y$  исходного кадра.

9.1. Вычисление производной по  $y$  и занесение ее в массив в каждой точке.

**Шаг 10.** Цикл от  $i = 0$  до  $h - 1$  по оси  $x$ , от  $j = 0$  до  $w - 1$  по оси  $y$ .

10.1. Вычисление суммы производных по  $x$  и  $t$ .

**Шаг 11.** Цикл от  $i = 0$  до  $h - 1$  по оси  $x$ , от  $j = 0$  до  $w - 1$  по оси  $y$ .

11.1. Вычисление суммы производных по  $y$  и  $t$ .

**Шаг 12.** Вычисление определителя матрицы  $(A^T A)$ .

**Шаг 13.** Вычисление смещений исходного кадра от опорного по  $x$  и  $y$  ( $V_x, V_y$ ) — решение матричного уравнения (5).

На выходе — смещения исходного кадра относительно опорного по  $x$  и  $y$  ( $V_x, V_y$ ).

### 2.2.2. Усреднение последовательности кадров

С учетом найденного межкадрового движения интерполированные кадры исходной последовательности совмещаются усреднением. Результатом этого этапа является немного размытое совмещенное изображение высокого разрешения.

Усреднение проводится по всем кадрам последовательности с использованием опорного кадра, относительно которого и сравниваются пиксели

других кадров, и смещений, полученных на ранних этапах (рис. 4, см. вторую сторону обложки).

Усреднение проводится по следующей формуле:

$$z = \frac{1}{N} \sum_{k=1}^N W_k(x, y), \quad (7)$$

где  $z$  — результат усреднения;  $N$  — число кадров, которые учитываются при усреднении данного пикселя;  $W_k(x, y)$  — кадры, пиксели которых учитываются при усреднении.

На вход алгоритма поступают: последовательность кадров — массивы, массивы смещений кадров относительно опорного ( $mas\_V_x, mas\_V_y$ ), число кадров ( $N$ ). Этапы работы алгоритма усреднения следующие:

**Шаг 1.** Выделение памяти для вспомогательных массивов яркостей изображений, массивов вычислений.

**Шаг 2.** Цикл от  $j = 1$  до  $N - 1$  (по всем смещениям всех кадров).

2.1. Если модуль числа в массиве смещения по  $x$  на изображении  $> 0$ , присваиваем массив смещения — смещение на  $+1$  пиксель, иначе присваиваем массив смещения — смещение  $= 0$ .

2.2. Если модуль числа в массиве смещения по  $y$  на изображении  $> 0$ , присваиваем массив смещения — смещение на  $+1$  пиксель, иначе присваиваем массив смещения — смещение  $= 0$ .

**Шаг 3.** Цикл от  $i = 0$  до  $h - 1$  по оси  $x$ , от  $j = 0$  до  $w - 1$  по оси  $y$  опорного кадра.

3.1. Заполняем массив яркостями по всему исходному изображению.

**Шаг 4.** Цикл от  $j = 1$  до  $N - 1$  (проход по массиву, в котором сохранены все яркости кадров в отдельных массивах, т. е. массив массивов).

4.1. Цикл от  $i = 0$  до  $h - 1$  по оси  $x$ , от  $j = 0$  до  $w - 1$  по оси  $y$  опорного кадра.

4.1.1. Заполняем массив яркостями по всему исходному изображению.

**Шаг 5.** Цикл от  $i = 0$  до  $h - 1$  по оси  $x$ , от  $j = 0$  до  $w - 1$  по оси  $y$  опорного кадра.

5.1. Присваиваем переменной  $dx$  яркость конкретного пикселя.

5.2. Цикл от  $j = 1$  до  $N - 1$  (проход по массиву, в котором сохранены все яркости кадров в отдельных массивах, т. е. массив массивов).

5.2.1. Если смещения кадров в границах опорного кадра, то присваиваем переменной  $d$  (значение яркости в конкретной точке  $(x, y)$ ) яркость в конкретных пикселях и суммируем эти яркости, считаем число таких кадров, иначе присваиваем переменной  $d = 0$  яркость.

5.3. Суммируем яркости кадров и делим на число кадров.

**Шаг 6.** Заполняем изображение пикселями.

На выходе — совмещенное изображение высокого разрешения.

### 2.3. Устранение размытости и повышение резкости изображения высокого разрешения

Для устранения размытости и повышения резкости изображения необходимо использовать фильтр. Создавать и накладывать на изображения фильтры можно с помощью матрицы свертки (см. таблицу).

*Матрица свертки* — это преобразование одной матрицы с помощью другой, которая называется ядром.

В качестве ядра обычно используется матрица размером  $3 \times 3$ , но возможно и больше ( $5 \times 5, 7 \times 7$  и т. д.). Ядро содержит степени влияния окружающих значений элемента на сам элемент.

Преобразования происходят следующим образом. Каждый элемент исходной матрицы умножается на центральное значение матрицы ядра. Кроме этого, на соответствующие значения умножаются окружающие его элементы (при размере ядра  $3 \times 3$  их будет 8), после чего результаты суммируются и принимаются как преобразованное значение.

При обработке изображений необходима еще *нормализация*. Для нормализации результата ис-

Простейшие фильтры постобработки кадра для разрабатываемого алгоритма

№	Тип фильтра	Матрица свертки	Делитель (div)
1	Сглаживание (сильное)	$\begin{matrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{matrix}$	25
2	Сглаживание (среднее)	$\begin{matrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 3 & 1 & 0 \\ 0 & 3 & 9 & 3 & 0 \\ 0 & 1 & 3 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{matrix}$	25
3	Сглаживание (слабое)	$\begin{matrix} 0 & 1 & 2 & 1 & 0 \\ 1 & 3 & 10 & 3 & 1 \\ 2 & 10 & 90 & 10 & 2 \\ 1 & 3 & 10 & 3 & 1 \\ 0 & 1 & 2 & 1 & 0 \end{matrix}$	154
4	"Мягкая" резкость	$\begin{matrix} -1 & -1 & -1 & -1 & -1 \\ -1 & 3 & 4 & 3 & -1 \\ -1 & 4 & 13 & 4 & -1 \\ -1 & 3 & 4 & 3 & -1 \\ -1 & -1 & -1 & -1 & -1 \end{matrix}$	25 (*)
5	Резкость (средняя)	$\begin{matrix} -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & 49 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 \end{matrix}$	25
6	Резкость (слабая)	$\begin{matrix} 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & -3 & -1 & 0 \\ 0 & -3 & 41 & -3 & 0 \\ 0 & -1 & -3 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{matrix}$	25

\* — отрицательные значения повышают контраст.

пользуются дополнительные переменные:  $\text{div}$  (делитель) и  $\text{offset}$  (коэффициент). Они работают следующим образом: результат преобразования делится на  $\text{div}$  и к нему прибавляется  $\text{offset}$ .

При преобразованиях в качестве  $\text{div}$  обычно принимается сумма всех элементов матрицы скручивания. Это условие позволяет не допустить цветных искажений, если они не нужны.

Действительно, если преобразуемая область содержит один и тот же цвет, то результат получится как сумма элементов ядра, умноженная на этот цвет. Соответственно, чтобы оставить цвет без изменений, надо разделить результат преобразования на эту самую сумму.

На вход алгоритма поступает кадр высокого разрешения с нечеткими или размытыми границами, объектами (массивы) и числовой номер фильтра.

### **Алгоритм устранения размытости и повышения резкости**

Данный алгоритм следующий:

**Шаг 1.** Выделение памяти под массив яркостей изображения.

**Шаг 2.** Если выбран фильтр от 1 до 6, то выбрать соответствующую маску от 1 до 6 для свертки.

**Шаг 3.** Цикл от  $i = 0$  до  $h - 3$  по оси  $x$ , от  $j = 0$  до  $w - 3$  по оси  $y$  (заканчиваем проход в цикле на два пикселя раньше, так как маска  $5 \times 5$ ).

3.1. Циклы: от  $i = 0$  до 4 по оси  $x$ , от  $j = 0$  до 4 по оси  $y$  (от 0 до 4, так как маска  $5 \times 5$ ).

3.1.1. Умножение соответствующих соседних пикселей на маску, вычисление яркости в точке  $(x, y)$  — свертка.

3.2. Деление полученной яркости на делитель ( $\text{div}$ ), заполнение изображения яркостями.

На выходе — результирующее изображение высокого разрешения.

### **Заключение**

В работе изложено описание предложенной модели и разработан алгоритмический модуль повышения информативности изображений (повышения разрешения изображений). Дальнейшая реализация данного алгоритма позволит создать программно-алгоритмическое обеспечение, которое будет использоваться в рамках генерации цифровых изображений отпечатков пальцев. Рассматриваются два случая применения данного алгоритма для решения задач повышения информативности изображения с целью показать две возможные сферы, в которых использование цифровых изображений низкого разрешения может привести к негативным последствиям. Универсальность алгоритма по отношению к областям применения показывает его практическую значимость. Так, в рамках дактилоскопической идентификации повышение информативности изображений отпечатков позволяет использовать недорогое оборудование сканеров.

### **Список литературы**

1. **Насонов А. В., Крылов А. С.** Суперразрешение изображений и видео. [Электронный ресурс]. URL: <http://imaging.cs.msu.ru/ru/research/superresolution>
2. **Маркелов К. С.** Модель повышения информативности цифровых изображений на базе метода суперразрешения // Электронный научно-технический журнал Инженерный вестник. М.: ФГБОУ ВПО "МГТУ им. Н. Э. Баумана". 2013. № 3. 18 с.
3. **Lucas—Kanade** algorithm. Copterix. [Электронный ресурс]. URL: [http://copterix.perso.rezel.net/?page\\_id=58](http://copterix.perso.rezel.net/?page_id=58)
4. **Lucas B. D. and Kanade T.** An iterative image registration technique with an application to stereo vision // DARPA Image Understanding Workshop, April 1981.

**K. S. Markelov**, Graduate Student, e-mail: [kosmar89@mail.ru](mailto:kosmar89@mail.ru),

**V. V. Nechaev**, Dr. Head Chair, e-mail: [nechaev@mirea.ru](mailto:nechaev@mirea.ru),

The Moscow State Technical University of Radio Engineering, Electronics and Automatics

## **Algorithm to Improve the Quality of Digital Images of Fingerprint Importation**

*The issues of image processing in order to increase the resolution of the digital image based super resolution-method. The algorithm more informative digital image preprocessing which includes image transfer to grayscale, finding inter-frame motion by Lucas — Canada, and then averaged into a single frame and high-resolution digital image post-processing to eliminate blurring and sharpening. The proposed algorithm is invariant to specific problems and can be used in various fields, in particular — biometrics.*

**Keywords:** digital image, resolution, super resolution, Lucas—Canada method, Biometrics, fingerprinting

### **References**

1. **Nasonov A. V., Krylov A. S.** *Superrazreshenie izobrazhenii i video*. [Elektronnyi resurs]. URL: <http://imaging.cs.msu.ru/ru/research/super-resolution>
2. **Markelov K. S.** Model' povysheniya informativnosti cyfrovikh izobrazhenii na base metoda superrazresheniya. *Elektronnyi nauchno-*

*tehnicheskii jurnal Injenemyi vestnik*. MGTU. im. Baumana. 2013. N. 3. 18 p.

3. **Lucas—Kanade** algorithm. Copterix. [Elektronnyi resurs]. URL: [http://copterix.perso.rezel.net/?page\\_id=58](http://copterix.perso.rezel.net/?page_id=58)

4. **Lucas B. D. and Kanade T.** An iterative image registration technique with an application to stereo vision. *DARPA Image Understanding Workshop*. April 1981.

# ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В БИМЕДИЦИНСКИХ СИСТЕМАХ INFORMATION TECHNOLOGIES IN BIOMEDICAL SYSTEMS

УДК 004.932

**С. С. Садыков**, д-р техн. наук, проф., e-mail: sadykovss@yandex.ru,

**Ю. А. Буланова**, аспирант, e-mail: yuliyabulanova@yandex.ru,

**Е. Е. Канунова**, канд. техн. наук, доц., e-mail: kanunovae@list.ru,

Муромский институт (филиал) ФГБОУ ВПО "Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых", г. Муром,

**Е. А. Захарова**, врач-рентгенолог 1-й категории

НУЗ Отделенческая больница на станции Муром, ОАО "РЖД"

## Разработка концепций построения информационной системы диагностики новообразований на маммограммах

*Анализируются известные информационные системы обработки и анализа маммограмм как зарубежных, так и российских фирм-производителей. Поставлены цели и требования для разработки информационной системы диагностики новообразований на маммограммах. Приведена и подробно описана структура разработанной системы обработки и анализа маммограмм. Информационная система содержит в своем составе несколько программных модулей, которые в статье представлены в виде блок-схемы. Экспериментальные исследования с участием эксперта (врача-рентгенолога) и их результаты диагностики новообразований по маммографическим снимкам с использованием системы и в диалоговом режиме показали целесообразность применения разработанной системы для диагностики новообразований. Основные преимущества системы: высокая точность распознавания новообразований, сравнительно малый процент ложноположительных результатов, устранение субъективности при постановке диагноза, учет индивидуальных особенностей строения молочных желез пациентов.*

**Ключевые слова:** маммограмма, система обработки и анализа маммограмм, информационная система для обработки и анализа маммограмм, случайное марковское поле, выделение области молочной железы, предварительная обработка, текстурная сегментация, выделение области новообразования, выделение контуров, идентификация новообразований

### Введение

Заболевания молочной железы находятся на первом месте в списке заболеваний женщины. Основным методом неинвазивной диагностики патологий молочной железы является маммография [1].

На данный момент большинство лечебных учреждений Российской Федерации оснащены аналоговыми маммографическими комплексами, которые имеют недостатки [2]. Одним из них является ручная обработка и анализ маммограммы, которая приводит к субъективности диагноза, а следовательно, иногда и к пропуску новообразований, которые еще поддаются лечению. Дорогостоящие цифровые маммографы доступны только в частных клиниках. Таким образом, требуется автоматизация процесса обработки и анализа маммограмм.

Целью настоящей статьи является создание информационной системы для обработки и анализа маммограмм (ИСОАМ).

Для реализации поставленной цели необходимо решить следующие задачи:

- анализ существующих систем обработки/анализа маммограмм;
- формулирование основных требований к создаваемой информационной системе;
- разработка целесообразной структуры этой информационной системы (ИС);
- описание основных этапов реализации ИС;
- тестирование работы ИС на тестовых и реальных снимках для выяснения качества получаемых результатов.

### Состояние проблемы

Самыми известными фирмами-производителями программного обеспечения (ПО) для обработки и анализа маммограмм являются PARASCRIP (США) [3], Scanis (США) [4], iCAD (США) [5, 6], Siemens (Германия) [7], Fujifilm (Япония) [8]. Таким

образом, основной страной-производителем медицинского ПО для маммографии является США. В ходе проведенного анализа выяснилось, что в России коммерческих систем обработки маммограмм не существует вообще.

Малая часть из систем вышеуказанных производителей позволяет выделять области новообразований, но ни одна из них не дает возможности распознавания заболеваний по маммограммам. Большинство систем имеют базовый набор операций, который включает в себя только предварительную обработку маммограмм, однако узнать конкретный минимум функций можно только после составления договора на поставку программного обеспечения. Кроме всего прочего, стоимость зарубежного программного обеспечения [9] является за пределами для большинства лечебных учреждений России. Поэтому актуальным для России является разработка ИС диагностики новообразований в молочной железе по маммографическим снимкам.

### Особенности диагностики новообразований в молочной железе

В настоящее время врачи для определения заболеваний молочной железы придерживаются "Международной классификации болезней — 10" (МКБ-10) [10].

МКБ пересматривается один раз в 10 лет Всемирной организацией здравоохранения (ВОЗ). Целью МКБ является создание условий для систематизированной регистрации, анализа, интерпретации и сравнения данных о смертности и заболеваемости, полученных в разных странах или регионах и в разное время. МКБ используется для преобразования словесной формулировки диагнозов болезней и других проблем, связанных со здоровьем, в буквенно-цифровые коды, которые обеспечивают удобство хранения, извлечения и анализа данных.

Кроме МКБ, для диагностики новообразований врачи-рентгенологи используют протокол результатов лучевых диагностических исследований молочных желез [11], в котором указаны правила оформления маммографического исследования, схема изучения рентгенограммы молочной железы, структура и содержание протокола исследования молочной железы.

Основной сложностью диагностики новообразований на маммографическом снимке является наличие той или иной формы мастопатии, из-за которой плотная молочная железа имеет однородную структуру. В таком случае, иногда приходится прибегать к другим методам исследований (магнитно-резонансная томография, сонография и т. д.). Кроме того, возникает сложность при диагностике рака молочной железы, так как на ранней стадии это заболевание болезненных ощущений не вызывает, и опухоль имеет малый размер. Таким образом, в условиях наличия мастопатии в молочной железе диаг-

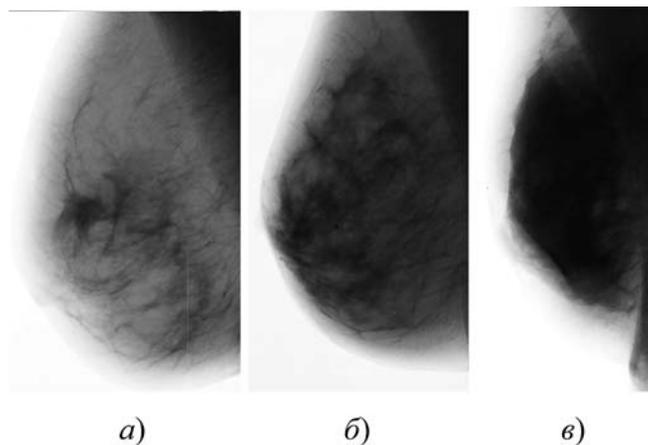


Рис. 1. Примеры структуры молочной железы: а — жировая инволюция; б — фиброзно-кистозная болезнь; в — аденоз

ностировать опухоль врачу-рентгенологу без какой-либо обработки маммограммы практически невозможно. В работе [12] указаны основные характеристики молочной железы с преобладанием железистого компонента (аденозом) и проблемы выделения новообразований в этом случае.

Как видно из рис. 1, жировая инволюция (рис. 1, а) — это самая простая структура для диагностики заболеваний, при фиброзно-кистозной болезни (рис. 1, б) и аденозе (рис. 1, в) выявить визуально новообразование достаточно сложно.

### Требования для разработки информационной системы

На первом этапе на маммограмме должен быть удален шум от маммографического аппарата и выровнены яркость и контрастность снимка. Однако обработка всего снимка не приведет к требуемому эффекту, так как область грудной мышцы и фона внесут помехи при обработке. Таким образом, необходимо предусмотреть выделение только области молочной железы и предварительную обработку именно этой области. Следовательно, должна быть разработана подсистема предварительной обработки.

Основываясь на статистические данные, собранные авторами, самыми распространенными заболеваниями молочной железы являются киста, фиброаденома и рак. Следовательно, разрабатываемая система должна содержать подсистему выделения областей для этих трех новообразований. После выделения областей для классификации объектов необходим расчет их текстурных характеристик.

Неотъемлемой частью системы должна быть подсистема распознавания (классификации) выделенных областей новообразований.

Кроме того, необходимо предусмотреть возможность в ручном режиме врачу-рентгенологу выделять подозрительные области, проводить обработку и расчет численных характеристик этих областей.

Составной частью ИС обработки и анализа маммограмм должен быть модуль работы с базой данных о пациентах. Эта база должна содержать основную информацию о пациенте, результаты медицинского обследования, рентгенограммы молочной железы.

Необходимым требованием врачей-рентгенологов является подсистема формирования отчетов по результатам обследования каждого пациента и статистических отчетов по всем исследованиям.

Для использования в аналоговых комплексах следует предусмотреть блок оцифровки аналоговых маммограмм. В этом блоке необходимо использовать функцию импорта оцифрованного изображения из различных форматов (bmp, png, tiff и т. д.) в формат DICOM-3.0 (это последняя версия протокола DICOM, используемого для передачи данных между медоборудованием и медицинскими информационными системами).

### Структура и основные алгоритмические модули

После анализа всех требований, предъявляемых к системе, авторами была разработана структура ИСОАМ, которая представлена на рис. 2 в виде блок-схемы. В блок-схеме черными рамками выделены основные блоки обработки, анализа и хранения маммограмм.

Функции, которыми должна обладать информационная система:

- регистрация маммографических снимков;
- автоматическое определение области молочной железы на снимке;
- предварительная обработка снимков, содержащих три типа новообразований;
- выделение областей новообразований трех типов (киста, фиброаденома, рак молочной железы);
- идентификация выделенных областей новообразований;
- визуальный контроль результатов распознавания с возможностью редактирования и выпол-

нения других этапов обработки на любом из шагов алгоритмов распознавания;

- работа с базой данных (БД) пациентов (выполнение стандартных операций над БД);
- хранение всех изменений, внесенных пользователем в процесс редактирования данных;
- генерация файла формата DICOM, содержащего основные данные об исследовании;
- формирование и печать отчета (протокола) маммографического исследования.

### Особенности реализации ИСОАМ

В ИСОАМ была предусмотрена возможность ее интеграции как с цифровым, так и с аналоговым маммографическим комплексом. Если результатом первого этапа является пленочная маммограмма, то с помощью блока оцифровки изображения переводится в цифровой формат. Для оцифровки снимков используется сканер, описанный в работах [12, 13].

Пленочная маммограмма из первого блока рис. 2 поступает на вход второго блока, где проводится оцифровка маммограммы.

Оцифрованный снимок молочной железы поступает в подсистему предварительной обработки. На первом этапе с использованием случайных марковских полей [14] формируется модель молочной железы (рис. 3, см. вторую сторону обложки).

Изображение разбивается на блоки  $7 \times 7$  пикселей. Формируются компоненты марковской модели для набора блоков  $D = \{1, \dots, y\}$ , где  $D$  — множество блоков,  $y$  — число блоков, на которое разбито изображение:

$$W_y = \{W_d | d \in D\}, \quad (1)$$

где  $W_y$  — окрестность блока  $y$ ;  $W_d$  — окрестность каждого пикселя  $d$ .

Для каждой окрестности рассчитываются два показателя: средняя яркость и среднеквадратическое отклонение (СКО). Определяются достоверные интервалы для каждой области, затем осуществляется объединение одинаковых блоков (рис. 3, а). Дальнейшая обработка проводится только внутри контура, который очерчивает молочную железу (рис. 3, б). На рис. 3, б видно, что молочная железа состоит из трех областей: белым цветом отображена область грудной мышцы, черным цветом — фон, а серым — область молочной железы.

Преимущество использования такой модели дает возможность автоматически ограничивать область для последующей обработки, кроме того, учитывает особенности строения молочных желез у каждого пациента индивидуально.

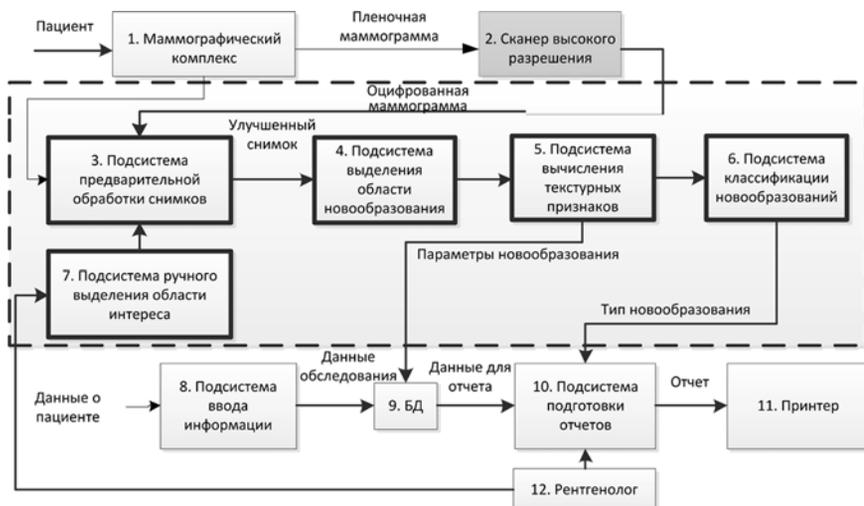


Рис. 2. Блок-схема программных модулей ИСОАМ

Далее врач, учитывая плотность молочной железы пациента (зависит от многих факторов, например от наличия мастопатии), применяет алгоритмы, улучшающие яркостные характеристики маммограммы, в основу положены алгоритмы гистограммного преобразования изображения и фильтрации [15].

Например, снимок, отмечающий рак молочной железы (РМЖ), требует вмешательства врача-рентгенолога, так как для диагностики данного типа новообразования необходимо учитывать характеристики ткани молочной железы пациентки (жировая инволюция, наличие мастопатии: фиброзно-кистозная болезнь (ФКБ), аденоз). Авторами были разработаны несколько алгоритмов предварительной обработки маммографических снимков [1, 12, 16], учитывающих свойства ткани молочной железы и позволяющие обнаруживать опухоли, находящиеся под мастопатией, которые врач-рентгенолог не способен увидеть (можно отметить выделение области рака молочной железы на фоне ФКБ или аденоза).

На следующем этапе улучшенный снимок поступает на вход подсистемы выделения областей новообразований, где изображение подвергается текстурной сегментации, зависящей от предыдущих алгоритмов предварительной обработки. Серия опытов показала, что для локализации всех трех типов патологий молочных желез беспроектным вариантом является использование алгоритма маркерного водораздела [17].

Подсистема вычисления признаков основана на вычислении признаков Харалика второго порядка по матрице смежности [18]. Общее число признаков равно 64. Вектор признаков выделенной области  $I[v_1, v_2, \dots, v_{64}]$  сравнивается с эталонными векторами  $V_3[v_3, 1, v_3, 2, \dots, v_3, 64]$  — признаками из базы знаний, при этом вычисляются минимумы средних сумм квадратов отклонений [19]. Тем самым определяется тип новообразования, т. е. осуществляется машинная диагностика болезни молочной железы (подсистема классификации новообразований). Для проверки принадлежности к тому или иному типу новообразования использовался также дискриминантный анализ Фишера [20].

Подсистемы работы с базой данных обеспечивают стандартные операции с БД (добавление, изменение, удаление исследований), формируют статистику по исследованиям.

### Исследование работы системы

На аналоговом маммографе Маммодиагност МД-РА [12] было получено приблизительно 500 снимков на пленке с размером  $18 \times 24$  см. Кроме того, в качестве обучающей выборки использовались снимки из БД маммографических исследований MIAS [14]. Таким образом, были сформированы две выборки: обучающая (в нее вошли снимки 322 снимка из базы MIAS), контрольная выборка (из 500

### Результаты диагностики-распознавания новообразований

Параметры		Число снимков	Заключение врача-рентгенолога		Заключенные системы	
Снимки с "нормой"	Жировая инволюция	50	50	100 %	50	100 %
	ФКБ	105	105	100 %	90	85,7 %
	Аденоз	28	28	100 %	22	78,5 %
Снимки с кистами	Жировая инволюция	39	39	100 %	35	89,7 %
	ФКБ	24	11	45,8 %	20	83 %
	Аденоз	6	3	50 %	4	67 %
Снимки с фиброаденомами	Жировая инволюция	7	7	100 %	6	85,7 %
	ФКБ	9	6	66,7 %	8	88,9 %
	Аденоз	6	3	50 %	5	83,3 %
Снимки с РМЖ	Жировая инволюция	23	23	100 %	22	95,6 %
	ФКБ	20	8	40 %	17	85 %
	Аденоз	5	2	40 %	4	80 %

снимков были отобраны 100 снимков, диагнозы на которых были подтверждены биопсией). Размер оцифрованных снимков равен  $1024 \times 1024$  пикселей.

Обучающая выборка под контролем рентгенолога была обработана рассмотренной выше технологией обнаружения новообразований. Для каждой области выделенного новообразования был сформирован вектор эталонных признаков и занесен в базу знаний.

Результаты распознавания новообразований обучающей выборки с помощью ИСОАМ, построенной на основе разработанной технологии, и врачом-рентгенологом были объединены в таблицу.

### Заключение

В данной статье была предложена разработанная информационная система обработки и анализа маммограмм, которая представляет собой аналог иностранных информационных систем обработки маммограмм, однако имеет перед ними ряд существенных преимуществ: низкая стоимость, высокая скорость постановки предварительного диагноза, выделение областей новообразований, не видимых глазу рентгенолога, возможность широкого использования в силу распространенности аналоговых маммографических комплексов в России, учет типов тканей молочной железы, возможность обработки как аналоговых, так и цифровых маммограмм.

Результаты экспериментальных данных показали, что и система, и врач с высокой точностью выделяют все новообразования на фоне жировой инволюции, однако на фоне ФКБ и аденоза у врача возникают сложности с постановкой диагноза. Так, врач выявил не более 50 % как доброкачественных (киста, фиброаденома), так и злокачественных новообразований на фоне ФКБ. Система же правильно

выделяет более 80 % новообразований. Таким образом, наличие мастопатии любого типа уменьшает точность постановки правильного диагноза, так как накладывает тень практически на всю область молочной железы, чем увеличивает вероятность пропуска рентгенологом новообразований на ранней стадии при скрининге. Указанный недостаток устраняет основанная на предложенной технологии комплексная система, благодаря которой при обработке маммограмм, осложненных мастопатией, убирается верхний слой ткани и становится видно новообразование. Кроме того, учитывая возможность выделения контуров подозрительных областей, можно определить, к какому виду они относятся (доброкачественные или злокачественные). С помощью этой системы удастся выявить новообразования диаметром менее 7 мм, в то время как существующие САД-системы способны выявить новообразование в стадии, не поддающейся лечению.

#### Список литературы

1. Садыков С. С., Буланова Ю. А., Захарова Е. А. Алгоритм выделения области кисты на малоконтрастных маммограммах // Информационные технологии. 2013. № 8. С. 53—57.
2. Садыков С. С., Буланова Ю. А., Захарова Е. А. Диалоговая система анализа маммографических снимков // Алгоритмы, методы и системы обработки данных. 2012. № 19. С. 155—187.
3. Parascript. Medical Imaging. AccuDetect. URL: <http://www.parascript.com/medical-imaging/> (дата обр.: 25.11.2013).
4. MedWOWGlobal. Компании медицинского оборудования. Scanis. URL: <http://ru.medwoglobal.com/company/scanis/91190> (дата обр.: 25.11.2013).
5. iCAD. Mammography CAD. TotalLook MammoAdvantage. URL: <http://www.icadmed.com/products/mammography/total-look.htm> (дата обр.: 25.11.2013).
6. iCAD. Mammography CAD. SecondLook Digital. URL: <http://www.icadmed.com/products/mammography/secondlookdigital.htm> (дата обр.: 25.11.2013).
7. Computer-Aided Detection for Digital Mammography syngo MammCAD. Siemens. 2008. P. 1—8. URL: [http://www.medical.siemens.com/siemens/en\\_INT/gg\\_sps\\_FBAs/files/brochures/cad/finalpdfMammCAD.pdf](http://www.medical.siemens.com/siemens/en_INT/gg_sps_FBAs/files/brochures/cad/finalpdfMammCAD.pdf) (дата обр.: 25.11.2013).
8. FUJIFILM Digital Mammography CAD. URL: <http://www.fujifilm.eu/eu/products/medical-systems/products/p/fujifilm-digital-mammography-cad/> (дата обр.: 25.11.2013).
9. New Technologies to Help Improve Mammography. URL: <http://www.imaginis.com/mammography/new-technologies-to-help-improve-mammography> (дата обр.: 20.01.2014).
10. Международная классификация болезней 10-го пересмотра. URL: <http://mkb-10.com/> (дата обр.: 20.01.2014).
11. Линденбрaten Л. Д. Рекомендации по протоколированию результатов лучевых диагностических исследований молочных желез // Радиология — практика. 2001. № 3. С. 36—39.
12. Буланова Ю. А. Использование информационных технологий для локализации области рака молочной железы на маммограммах с преобладанием железистого компонента // Прикаспийский журнал: управление и высокие технологии. 2013. № 3 (23). С. 100—111.
13. Шербаков С. На стыке двух эпох: Планшетный сканер Epson Perfection V750 Pro // Foto & Video, 2006. № 7. С. 60—65.
14. Ахметшин А. М., Федоренко А. Е. Сегментация низкоконтрастных изображений с применением иерархической модели Марковского случайного поля // Искусственный интеллект. 2002. № 4. С. 414—421.
15. Садыков С. С., Захарова Е. А., Буланова Ю. А. Технология выделения кисты на маммограмме // Вестник РГРТУ. Рязань. 2013. № 1. С. 7—11.
16. Sadykov S. S., Bulanova Y. A. Algorithm of localization of breast cancer in the background of mastopathy // 11<sup>th</sup> Internat. Conf. on Pattern Recognition and Image Analysis: New Information Technologies (PRIA-11—2013). 2013. V. 2. P. 717—721.
17. Садыков С. С., Буланова Ю. А., Захарова Е. А., Яшков В. С. Исследование маркерного водораздела для выделения области рака молочной железы // Алгоритмы, методы и системы обработки данных. 2013. № 23. С. 56—64.
18. Haralick R. M., Shanmugan K., Dinstein I. Textural Features for Image Classification // IEEE Transactions on systems, man and cybernetics. 1973. V. SMC-3, N. 6. P. 610—621.
19. Садыков С. С., Буланова Ю. А., Захарова Е. А., Каряев В. В. Автоматизация обнаружения новообразований на маммографических снимках // Тр. Междунар. симпозиума "Надежность и качество". 2012. Т. 2. С. 376—379.
20. Фишер Рональд Э. В сб.: Современные проблемы кибернетики. М.: Знание, 1979. С. 6—20.

S. S. Sadykov<sup>1</sup>, Professor, e-mail: sadykovss@yandex.ru, Yu. A. Bulanova<sup>1</sup>, Graduate Student, E. E. Kanunova<sup>1</sup>, Associate Professor, E. A. Zaharova<sup>2</sup>, Radiologist

<sup>1</sup> Vladimir State University named after Alexander Grigoryevich and Nickolay Grigoryevich Stoletovs

<sup>2</sup> Departmental Hospital Murom station JSC "Russian Railways"

## Development of Concepts for Building Information Systems Diagnosis of Tumors on Mammograms

*This article analyzes the known information systems processing and analysis of mammograms, both foreign and Russian manufacturers. It set goals and requirements for the development of an information system diagnosis of tumors on mammograms. Presented and described in detail the structure of the developed system of processing and analysis of mammograms. EC comprises in its structure several software modules which are presented in the article in the form of a flowchart. Experimental studies involving expert (radiologist) and their results diagnosis of tumors by mammographic images using the system and interactively demonstrated the feasibility of the developed system for the diagnosis of tumors. The main advantages of the system: the high recognition accuracy tumors, a relatively small percentage of false positive results, the elimination of subjectivity in the diagnosis, taking into account individual characteristics of the structure of the mammary glands of patients.*

**Keywords:** mammogram, the system processing and analysis of mammograms, ISOAM, Markov random field, selection area of the breast, pretreatment, texture segmentation, region selection neoplasms, edge detection, identification of tumors

## References

1. Sadykov S. S., Bulanova Ju. A., Zaharova E. A. Algoritm vydelenija oblasti kisty na malokontrastnyh mammogrammah. *Informacionnye tehnologii*. 2013. N. 8. P. 53–57.
2. Sadykov S. S., Bulanova Ju. A., Zaharova E. A. Dialogovaja sistema analiza mammograficheskikh snimkov. *Algoritmy, metody i sistemy obrabotki dannyh*. 2012. N. 19. P. 155–187.
3. Parascript. *Medical Imaging*. AccuDetect. URL: <http://www.parascript.com/medical-imaging/> (data obrashhenija: 25.11.2013).
4. MedWOWGlobal. *Kompanii medicinskogo oborudovaniya*. Scanis. URL: <http://ru.medwowglobal.com/company/scanis/91190> (data obrashhenija: 25.11.2013).
5. iCAD. *Mammography CAD. TotalLook MammoAdvantage*. URL: <http://www.icadmed.com/products/mammography/totallook.htm> (data obrashhenija: 25.11.2013).
6. iCAD. *Mammography CAD. SecondLook Digital*. URL: <http://www.icadmed.com/products/mammography/secondlookdigital.htm> (data obrashhenija: 25.11.2013).
7. Computer-Aided Detection for Digital Mammography syngo MammoCAD. Siemens. 2008. P. 1–8. URL: [http://www.medical.siemens.com/siemens/en\\_INT/gg\\_sps\\_FBAs/files/brochures/cad/fmalpdfMammoCAD.pdf](http://www.medical.siemens.com/siemens/en_INT/gg_sps_FBAs/files/brochures/cad/fmalpdfMammoCAD.pdf) (data obrashhenija: 25.11.2013).
8. FUJIFILM *Digital Mammography CAD*. URL: <http://www.fujifilm.eu/eu/products/medical-systems/products/p/fujifilm-digital-mammography-cad/> (data obrashhenija: 25.11.2013).
9. *New Technologies to Help Improve Mammography*. URL: <http://www.imaginis.com/mammography/new-technologies-to-help-improve-mammography> (data obrashhenija: 20.01.2014).
10. *Mezhdunarodnaja klassifikacija boleznej 10-go peresmotra*. URL: <http://mkb-10.com/> (data obrashhenija: 20.01.2014).
11. Lindenbraten L. D. Rekomendacii po protokolirovaniyu rezul'tatov luchevyh diagnosticheskikh issledovanij molochnyh zhelez. *Radiologija — praktika*. 2001. N. 3. P. 36–39.
12. Bulanova Ju. A. Ispol'zovanie informacionnyh tehnologij dlja lokalizacii oblasti raka molochnoj zhelezy na mammogrammah s preobladaniem zhelezistogo komponenta. *Prikaspijskij zhurnal: upravlenie i vysokie tehnologii*. 2013. N. 3 (23). P. 100–111.
13. Shherbakov S. Na styke dvuh jepoh: Planshetnyj skaner Epson Perfection V750 Pro. *Foto & Video*. 2006. N. 7. P. 60–65.
14. Ahmetshin A. M., Fedorenko A. E. Segmentacija nizkokontrastnyh izobrazhenij s primeneniem ierarhicheskoj modeli Markovskogo sluchajnogo polja. *Iskusstvennyj intellekt*. 2002. N. 4. P. 414–421.
15. Sadykov S. S., Zaharova E. A., Bulanova Ju. A. Tehnologija vydelenija kisty na mammogramme. *Vestnik RGRTU. Rjazan'*. 2013. N. 1. P. 7–11.
16. Sadykov S. S., Bulanova Y. A. Algorithm of localization of breast cancer in the background of mastopathy. *11<sup>th</sup> International Conference on Pattern Recognition and Image Analysis: New Information Technologies (PRIA-11–2013)*, 2013. V. 2. P. 717–721.
17. Sadykov S. S., Bulanova Ju. A., Zaharova E. A., Jashkov V. S. Issledovanie markernogo vodorazdela dlja vydelenija oblasti raka molochnoj zhelezy. *Algoritmy, metody i sistemy obrabotki dannyh*. 2013. N. 23. P. 56–64.
18. Haralick R. M., Shanmugan K., Dinstein I. Textural Features for mage Classification. *IEEE Transactions on systems, man and cybernetics*. 1973. V. SMC-3, N. 6. P. 610–621.
19. Sadykov S. S., Bulanova Ju. A., Zaharova E. A., Karjaev V. V. Avtomatizacija obnaruzhenija novoobrazovamj na mammograficheskikh snimkah. *Trudy mezhdunarodnogo simpoziuma "Nadezhnost' i kachestvo"*. 2012. V. 2. P. 376–379.
20. Fisher Ronald' d Je. V sb.: *Sovremennye problemy kibernetiki*. M.: Znanie, 1979. P. 6–20.

УДК 615.47-114:616-07-08

Н. Т. Абдуллаев<sup>1</sup>, канд. техн. наук, доц., e-mail: a.namik49@mail.ru,  
О. А. Дышин<sup>2</sup>, канд. физ.-мат. наук, ст. науч. сотр., Г. Э. Абдуллаева<sup>2</sup>, магистр,

<sup>1</sup> Азербайджанский технический университет

<sup>2</sup> Азербайджанская государственная нефтяная академия, г. Баку

## Интервальные байесовские механизмы вывода при дифференциальной диагностике нервно-мышечных заболеваний

На основе построенных доверительных интервалов для априорных вероятностей рассматриваемых заболеваний нервно-мышечной системы показана возможность использования байесовских механизмов вывода для принятия диагностических решений. Итеративное применение обобщенных интегральных байесовских схем приводит к уточненным оценкам достоверности заболевания исследуемого пациента по заданному симптомокомплексу значений диагностических признаков и позволяет рассмотреть различные возможные случаи подтверждения или отрицания гипотезы о наличии у пациента того или иного заболевания из рассматриваемой совокупности нервно-мышечных заболеваний.

**Ключевые слова:** электромиография, дифференциальная диагностика, диагностические признаки, байесовский механизм, интервальная оценка

### Введение

Внедрение инновационных технологий в биологию и медицину предполагает применение автоматизированных информационно-измерительных систем, программного обеспечения, методологии обработки и анализа информации, адекватно заложенной в биоэлектрических сигналах. Биоэлектрические сигналы в виде электроэнцефалограмм,

электрокардиограмм и электромиограмм являются отображением функционирования таких систем организма, как центральная нервная, сердечно-сосудистая и нейромышечная.

Существенное значение в функционировании организма человека занимает нейромышечная система. Наиболее эффективным методом оценки функционального состояния нейромышечной сис-

темы является регистрация ее биоэлектрической активности в электромиограммах (ЭМГ) и электронейрограммах [1, 2].

В отличие от других электрофизиологических методов функциональной диагностики (ЭЭГ, ЭКГ, ЭОГ и др.), следует отметить ряд особенностей ЭМГ-исследования при диагностике состояния нервно-мышечной системы:

- большое число объектов исследования (наличие в организме более нескольких сотен мышц) различного типа;
- возможность исследования проводящей функции моторных, сенсорных и вегетативных аксонов периферических нервов;
- возможность определения функционального состояния нервно-мышечной передачи [3–5].

Наличие этих и ряда других особенностей ЭМГ-методов исследования создает сложности для реализации неинвазивных электромиографических компьютерных систем по сравнению с другими диагностическими методами. Существующие классификации заболеваний мышц во многом, а порой и полностью, противоречивы [3].

В статистической теории распознавания классификацию образов проводят по известным значениям (информативным) диагностических признаков объектов обучающего множества, выделяя статистические свойства этих объектов и группируя сходные входные векторы значений их признаков в классы [6]. В случае отсутствия статистических данных о значениях признаков объектов в медицинской диагностике приходится с некоторой степенью уверенности пользоваться базой знаний врачей-экспертов в виде высказываний "если — то" так называемых "правил нечетких продукций" (ПНП). На основе этой базы формируются функции принадлежности входных и выходных признаков [7, 8].

В настоящей работе для дифференциальной диагностики нервно-мышечных заболеваний (демиелинизирующая полиневропатия (PNP), карпальный туннельный синдром (KTS) и кубитальный туннельный синдром (KubTS)) электромиографические методы исследования которых изложены в работах [1–5], применяются обобщенные интервальные байесовские механизмы вывода [9–11], отличные от традиционных (с априорной вероятностью заболевания, равной распространенности данного заболевания) [12–14] и нетрадиционных (итерационных) формул Байеса [15, 16], дающих точечную оценку апостериорных вероят-

ностей заболевания у исследуемого пациента. Интервальные байесовские подходы применимы как к точечным, так и к интервальным оценкам условных вероятностей значений диагностических признаков представленного симптомокомплекса исследуемого пациента и дают интервальные оценки апостериорных вероятностей каждого заболевания из рассматриваемой совокупности заболеваний.

### Оценка диагностических признаков исследуемых болезней с помощью функций принадлежности

Под диагностическими признаками будем понимать симптомы заболевания, данные инструментальных обследований пациента, клинические анализы. Всего использовалось 27 диагностических признаков (табл. 1), а если включить пол и возраст пациента, то их число возрастет до 29.

Первые 11 из них являются симптоматическими факторами (СФ), а остальные 16 — патогенетическими (ПФ).

В табл. 2 приведены диагностические признаки из истории болезни, полученные в результате электромиографического обследования конкретного пациента.

Из 27 используемых при компьютерной диагностике диагностических признаков в представленных данных содержатся 11, однако с накоплением

Таблица 1

Совокупность диагностических признаков

№	Название	Обозначение	Четкость	Тип фактора	Классы признаков	
1	Мышечная слабость	МСлаб	–	СФ	Дис.П	
2	Тонус мышцы	ТМ	–	СФ	Дис.П	
3	Локализация боли	ЛокБ	–	СФ	Дис.П	
4	Парез	Пар	–	СФ	Дис.П	
5	Сосудистые нарушения	СН	–	СФ	Дис.П	
6	Атрофия мышц	АтрМ	–	СФ	Дис.П	
7	Гипер- и гипотрихоз	ГТЗ	–	СФ	Дис.П	
8	Гипер- и гипогидроз	ГГЗ	–	СФ	Дис.П	
9	Чувствительные нарушения	ЧН	–	СФ	Дис.П	
10	Скованность конечностей	Сков.Кон	–	СФ	Дис.П	
11	Онемение	О	–	СФ	Дис.П	
12	Поражения двигательных волокон	ПДВ	–	ПФ	Дис.П	
13	Трофика ткани	ТрТ	–	ПФ	Дис.П	
14	Поражения нервов	ПНер	–	ПФ	Дис.П	
15	Локализация поражения нервов	ЛПНер	–	ПФ	Дис.П	
16	Наследственность	Н	+	ПФ	Дис.П	
17	Уровень миелина	УМ	+	ПФ	Дис.П	
18	Проводящие функции сенсорных нервов	Амплитуда	Амп <sub>С</sub>	+	ПФ	Непр.П
		Латентность	Лат <sub>С</sub>	+	ПФ	Непр.П
		Скорость	СРВ <sub>С</sub>	+	ПФ	Непр.П
21	Проводящие функции моторных нервов	Амплитуда	Амп <sub>М</sub>	+	ПФ	Непр.П
		Латентность	Лат <sub>М</sub>	+	ПФ	Непр.П
		Скорость	СРВ <sub>М</sub>	+	ПФ	Непр.П
24	Потенциал двигательных единиц	Длительность	ДПДЕ	+	ПФ	Непр.П
Амплитуда		АмпПДЕ	+	ПФ	Непр.П	
Форма		ФПДЕ	+	ПФ	Дис.П	
27	Функциональное состояние мышцы	ФСМ	+	ПФ	Дис.П	

Таблица 2

**Диагностические признаки пациента, полученные  
в результате электромиографического обследования**

№	Название	Обозначение	Значение
1	Форма потенциала двигательных единиц	ФПДЕ	Полифаз
2	Функциональное состояние мышцы	ФСМ	2 балла
3	Уровень миелина	УМ	110 МОм/см <sup>2</sup> (демиелинизация)
4	Длительность потенциала двигательных единиц	ДПДЕ	25 мс ( $\geq N$ )
5	Амплитуда потенциала двигательных единиц	АПДЕ	50 мкВ ( $> N$ )
6	Проводящие функции сенсорных нервов	Амплитуда	Амп <sub>С</sub> 17,7 мкВ ( $\ll N$ )
7		Скорость	СРВ <sub>С</sub> 44,4 м/с ( $< N$ )
8	Проводящие функции моторных нервов	Латентность	Лат <sub>С</sub> 2,7 мс ( $\ll N$ )
9		Амплитуда	Амп <sub>М</sub> 10 мкВ ( $N$ )
10		Скорость	СРВ <sub>М</sub> 53,9 м/с ( $N$ )
11		Латентность	Лат <sub>М</sub> 3,8 мс ( $< N$ )

*Примечание:*  $N$  показывает "степень нормы", как это принято в медицинской практике.

статистики планируется ввести в диагностику все имеющиеся данные, что уточнит конечный результат — диагноз.

Диагностические признаки разделяются на два класса: дискретные (Дис.П) и непрерывные (Непр.П).

Дискретные признаки (мышечная слабость, парез, атрофия мышцы, пол и т. д.) принимают одно из четко определенных значений. В рамках диагностируемых состояний — одно из двух значений: наличие патогенного признака или его отсутствие, норма или не норма. Например, наличие мышечной слабости или ее отсутствие, парез или нет, форма потенциала двигательных единиц — норма или не норма.

Состоянию "нет" (отсутствие) ставилось в соответствие число 0, состоянию "да" (наличие) — число 1. Для признака "пол": 0 — мужской пол, 1 — женский пол.

Медицинские данные иногда позволяют разделять дискретные признаки на три и более категории. Например, тонус мышцы: гипотонус, норматонус, гипертонус. Сосудистые нарушения (СН): побледнение, покраснение, "мраморный рисунок". При этом надо учесть, что увеличение возможных состояний дискретных признаков может привести к увеличению ошибок диагноза (в силу субъективности

Таблица 3

**Значения дискретных и непрерывных диагностических признаков**

Значения диагностического признака	Зн1	Зн2	Зн3	Зн4	Зн5	Зн6
МСлаб	Нет	Есть				
ТМ	Гипотонус	Норматонус	Гипертонус			
Наличие боли	Кисть	Локти	Шея			
Пар	Нет	Есть				
СН	Побледнение	Покраснение	"Мраморный рисунок"			
АтрМ	Полностью	По частям				
ГТЗ	Гипотрихоз	Норма	Гипертрихоз			
ГГЗ	Гипогидроз	Норма	Гипергидроз			
ЧН	Гипостезия	Норма	Гиперстезия	Глубокая гиперстезия		
Сков.Кон	Акинезия	Гипокинезия	Норма			
О	Слабо, переходящий	Сильно, сопровождающий болью	Полный	На ногах	На руках (от 1 до 3 пальцев кисти)	На руках (от 4 до 5 пальцев кисти)
ПДВ	Полностью	По частям				
ТрТ	Ниже	Норма				
ПНер	Денервация	Иннервация	Реиннервация			
ЛПНер	Срединный	Локтевой	Малый берцовый	Большой берцовый	Икроножный	Лучевой
Н	Неотягощенная	Слабо отягощенная	Сильно отягощенная			
УМ	Демиелинизация по частям	Демиелинизация				
Амп <sub>С</sub>	$\ll N$	$< N$	$N$	$> N$	$\gg N$	
Лат <sub>С</sub>	$\ll N$	$< N$	$N$	$> N$	$\gg N$	
СРВ <sub>С</sub>	$\ll N$	$< N$	$N$	$> N$	$\gg N$	
Амп <sub>М</sub>	$\ll N$	$< N$	$N$	$> N$	$\gg N$	
Лат <sub>М</sub>	$\ll N$	$< N$	$N$	$> N$	$\gg N$	
СРВ <sub>М</sub>	$\ll N$	$< N$	$N$	$> N$	$\gg N$	
ДПДЕ	$\ll N$	$< N$	$N$	$> N$	$\gg N$	
АмпПДЕ	$\ll N$	$< N$	$N$	$> N$	$\gg N$	
ФПДЕ	Полифаз	Норма				
ФСМ	Сила мышцы 1—2 балла	Сила мышцы 3 балла	Сила мышцы 4 балла	Сила мышцы 5 баллов		

обследования лечащим врачом), поэтому принято, что дискретный признак может иметь два значения. При этом в ущерб детальности повышается надежность исходных данных.

Непрерывный признак (амплитуда, латентность, скорость моторных и сенсорных нервов, потенциалы двигательных единиц) характеризуются вещественным числом в определенном интервале: от минимального до максимального значений признака. Анализ врачом значений ( $Z_n$ ) непрерывных признаков связан с медицинскими интервалами — нормами для каждого признака (кроме возраста). Каждое значение признака любого пациента чаще всего характеризуется так: норма, выше нормы, ниже нормы (табл. 3). Иногда используются дополнительные категории: много выше нормы, много ниже нормы и т. п.

### Теоретические основы байесовского подхода к диагностике

По классической формуле Байеса вероятность болезни каждого пациента вычисляется в виде

$$p_j = \frac{\pi_j \prod_i q_{ij}(a_{0i})}{\sum_j \pi_j \prod_i q_{ij}(a_{0i})}, \quad (1)$$

где  $p_j$  — вероятность  $j$ -й болезни пациента;  $\pi_j$  — доля больных болезнью  $j$  среди больных с диагностируемыми болезнями;  $q_{ij}(a_{0i})$  — плотность распределения (или вероятность)  $i$ -го диагностического признака (ДП)  $a_{0i}$  при  $j$ -й болезни (имеется в виду плотность распределения вероятности непрерывного ДП или вероятность дискретного ДП); "0" означает выявленное при обследовании рассматриваемого пациента фиксированное значение ( $a_{0i}^{(1)}, \dots, a_{0i}^{(k)}, \dots, a_{0i}^{(k_i)}$ )

вектора  $a_i = (a_i^{(1)}, \dots, a_i^{(k)}, \dots, a_i^{(k_i)})$  значений  $i$ -го ДП ( $k_i$  — общее число значений  $i$ -го ДП);  $\prod_i$  — произведение

всех  $i$ -х признаков от 1 до  $m$ . В знаменателе сумма таких произведений для всех диагностируемых болезней  $j$  (от  $j = 1$  до  $j = n$ ). В нашем случае  $n = 4$ :  $j = 1$  — РНР,  $j = 2$  — КТС,  $j = 3$  — КубТС,  $j = 4$  — прочие заболевания нервно-мышечной системы.

Важнейшим элементом байесовского подхода является теорема Байеса, которая для непрерывных случайных процессов формулируется следующим образом [17].

Обозначим  $p(y, \theta)$  совместную функцию плотности распределения вероятностей (ФПВ) для вектора случайных наблюдений  $y$  и вектора параметров  $\theta$ , который тоже считается случайным. Тогда в соответствии с обычными операциями над ФПВ можно записать

$$p(y, \theta) = p(y|\theta)p(\theta) = p(\theta|y)p(y) \quad (2)$$

и, таким образом,

$$p(\theta|y) = \frac{p(\theta)p(y|\theta)}{p(y)}, \quad (3)$$

где полагается, что  $p(y) \neq 0$ . В формуле (3)  $p(y|\theta)$  есть апостериорная ФПВ вектора параметров  $\theta$  при условии заданной выборочной информации  $y$ , причем

$$p(y) = \sum_{\theta} p(\theta)p(y, \theta), \quad (4)$$

где  $p(\theta)$  — априорная ФПВ для вектора параметра  $\theta$ , а  $p(y|\theta)$ , рассмотренная как функция от  $\theta$ , есть хорошо известная функция правдоподобия. Выражение (3) есть запись теоремы Байеса. При этом "принцип правдоподобия" утверждает, что  $p(y|\theta)$ , рассмотренная как функция от  $\theta$ , "полностью представляет все свидетельство эксперимента, т. е. рассказывает обо всем, о чем может рассказать эксперимент" [17].

Принимая за  $\theta$  скаляр, означающий номер  $j$ -й болезни,  $y = (y_1, \dots, y_m)$  — вектор диагностических признаков  $i$  ( $i = 1, \dots, m$ ) со значениями  $a_{0i}$ , выявленными у рассматриваемого пациента;  $p(\theta) = \pi_j$ ,  $p(y|\theta) = p(a_{0i}|j)$ ,  $p(\theta|y) = p(j|a_{0i})$  и  $p(y) = p(a_{0i})$ , формулу Байеса можно записать в виде

$$p(j|a_{0i}) = \frac{\pi_j q(a_{0i}|j)}{p(a_{0i})}, \quad (5)$$

где  $\pi_j$  — априорная ФПВ параметра  $j$ ;  $a_{0i}$  — вектор ( $a_{0i1}, \dots, a_{0im}$ ). По формуле (5) можно диагностировать у пациента наличие болезни  $j$  из некоторой совокупности диагностируемых болезней  $J = \{j|1, \dots, n\}$ .

В отличие от методов выборочной теории, байесовский подход допускает естественное объединение данных из разных источников для получения оценок параметров. Если при классическом подходе параметры функции распределения считаются константами, которые нужно оценить, то при байесовском подходе параметры рассматриваются как случайные величины, которые описываются некоторой априорной функцией распределения. На практике, как правило, имеющейся априорной информации не достаточно для точного определения априорной функции распределения. В этих условиях проблема выбора априорной функции распределения и вычисления байесовских оценок становится нетривиальной.

В случаях, когда полностью отсутствует априорная информация относительно оцениваемого параметра  $\theta$  или имеется незначительное количество такой информации, используют неинформативную априорную функцию распределения [18].

При выборе априорной функции распределения очень часто руководствуются соображениями простоты математических выкладок, с помощью которых вычисляются байесовские оценки. Такие свойства присущи сопряженным априорным функ-

циям распределения. Сопряженной априорной плотностью распределения  $h(\theta)$  для данной выборочной плотности  $f(x|\theta)$  является такая плотность распределения, для которой как априорная плотность распределения  $h(\theta)$ , так и апостериорная плотность распределения  $g(\theta|x)$  принадлежат одному и тому же классу распределения [19].

Будем полагать, что априорная функция плотности вероятностей для непрерывной случайной величины  $P = p_j$  имеет форму бета-распределения  $B(\alpha, \beta)$  с подходящим выбором параметров  $\alpha$  и  $\beta$ :

$$f_P(P) = \begin{cases} p^{\alpha-1}(1-p)^{\beta-1}, & 0 \leq p \leq 1; \\ 0, & \text{в других точках.} \end{cases} \quad (6)$$

Известно [18], что если априорное распределение для  $P$  есть  $B(\alpha, \beta)$  и если случайным образом выбраны  $n$  элементов совокупности, из которых  $a$  элементов подчиняется бета-распределению с некоторыми значениями параметров  $\alpha$  и  $\beta$ , а  $b$  элементов не обладают таким свойством ( $a + b = n$ ), то апостериорное распределение для  $P$  есть  $B(\alpha + a, \beta + b)$ . Это означает, что  $B(\alpha, \beta)$  является сопряженным априорным распределением.

В случае отсутствия априорной информации будем считать, что априорное распределение для  $P$  есть  $B(\alpha, \beta)$  с  $\alpha = 0$  и  $\beta = 0$ , и, следовательно, апостериорное распределение будет  $B(\alpha, \beta)$ . Среднее значение  $EB(\alpha, \beta)$  и дисперсия  $DB(\alpha, \beta)$  вычисляются по формулам [20]:

$$EB(\alpha, \beta) = \frac{\alpha}{\alpha + \beta}, \quad DB(\alpha, \beta) = \frac{\alpha\beta}{(\alpha + \beta)^2(\alpha + \beta + 1)}. \quad (7)$$

При отсутствии априорной оценки (что равносильно предположению, что априорное распределение есть  $B(0, 0)$ ) апостериорное распределение для  $P$  приближенно является нормальным распределением  $N[\hat{p}, \hat{p}(1-\hat{p})/(n+1)]$ , где  $\hat{p}$  — выборочная доля, а  $n$  — объем выборки [18]. Это позволяет указать (приблизительно)  $(1-\alpha)$ -процентный вероятностный интервал для переменной  $P$ :

$$\hat{p} \pm A_{1-\alpha} \sqrt{\frac{\hat{p}(1-\hat{p})}{n+1}}, \quad (8)$$

где константа  $A_{1-\alpha}$  равна 1,96; 2,58; 3,29 при  $\alpha = 0,05; 0,01; 0,001$  соответственно.

Таким образом, при отсутствии априорной оценки за  $\pi_j$  можно принять (с достаточно хорошей степенью точности при больших  $n$ ) долю больных  $j$ -й болезнью  $\hat{p}_j$  в общем числе  $n$  больных с диагностируемыми болезнями при достаточно большом числе  $n$ . При этом для  $\pi_j$  получаются нижняя  $\hat{\pi}_j^L$  и верхняя  $\hat{\pi}_j^R$  оценки

$$\hat{\pi}_j^L = \hat{p}_j - A_{1-\alpha} \sqrt{\frac{\hat{p}_j(1-\hat{p}_j)}{n+1}}; \quad \hat{\pi}_j^R = \hat{p}_j + A_{1-\alpha} \sqrt{\frac{\hat{p}_j(1-\hat{p}_j)}{n+1}}. \quad (9)$$

Основная трудность применения диагностической формулы (5) заключается в вычислении значений  $q(a_0, i|j) = p(a_0, i|j)$ , т. е. вероятности того,

что случайные величины  $a_1, \dots, a_m$  принимают соответственно значения  $a_{0,1}, \dots, a_{0,m}$ , выявленные у исследуемого пациента, при условии наличия у него болезни  $j$ . При недостаточности или отсутствии статистической базы знаний эту проблему можно решить на основе экспертных мнений, выраженных в виде правил нечетких продукций (ПНП) с определенной степенью уверенности. Из формулы (3) имеем

$$p(y|\theta) = \frac{p(y)p(\theta|y)}{p(\theta)}, \quad (10)$$

откуда в введенных нами обозначениях получим

$$q(a_0, i) = \frac{p(a_0, i)p(j|a_0, i)}{\pi_j} \quad (j = 1, \dots, n). \quad (11)$$

Для вычисления условных вероятностей  $p(j|a_0, i)$  используется система нечеткого логического вывода (*fuzzy inferences system — FIS*), предложенная в работах [7, 8] для эндоскопической диагностики и реализованная в работе [16] для рассматриваемой нами совокупности нервно-мышечных заболеваний. Для симптомокомплекса пациента  $a_0, i$ , заданного табл. 2, с помощью системы *FIS* получены следующие оценки [16]:

$$\begin{aligned} \hat{p}(1|a_0, i) &= 0,74 \hat{p}(2|a_0, i) = 0 \hat{p}(3|a_0, i) = \\ &= 0 \hat{p}(4|a_0, i) = 0,26. \end{aligned} \quad (12)$$

### Интервальный байесовский механизм принятия диагностических решений

Подставляя в формулу (11) для условной вероятности  $p(a_0, i|j) = q_j(a_0, i)$  вместо  $\pi_j$  нижнюю оценку  $\hat{\pi}_j^L = \hat{P}^L$  и верхнюю оценку  $\hat{\pi}_j^R = \hat{P}^R$ , где  $\hat{\pi}_j^L$  и  $\hat{\pi}_j^R$  определены формулой (9), получим соответственно нижнюю и верхнюю оценки для  $p(a_0, i|j)$

$$\begin{aligned} \hat{P}^L(a_0, i|j) &= \frac{p(a_0, i)\hat{p}(j|a_0, i)}{\hat{\pi}_j^R}; \\ \hat{P}^R(a_0, i|j) &= \frac{p(a_0, i)\hat{p}(j|a_0, i)}{\hat{\pi}_j^L}. \end{aligned} \quad (13)$$

Это дает возможность использовать байесовские механизмы вывода [9—11] для уточнения апостериорных оценок для условных вероятностей  $\hat{p}(j|a_0, i)$ ,  $j \in J = \{1, 2, 3, 4\}$ .

Следуя обозначениям и терминологии из работы [10], назовем гипотезой  $h_j$  болезнь  $j$  из рассматриваемого нами списка болезней  $J = \{1, 2, 3, 4\}$ , а симптомокомплекс  $a_0, i$  представленный табл. 2 — свидетельством  $e$ . Связь между гипотезами  $h$  из  $H = \{h_j\}_{j=1}^N$  и свидетельством  $e$  устанавливается следующим образом: каждой паре  $(e, h)$  ставится в соответствие вектор  $\delta(e, h) = (\hat{P}(e|h), \hat{P}(e|\bar{h}))$ . Здесь  $\hat{P}(e|h)$  и  $\hat{P}(e|\bar{h})$  — принятые в теории вероят-

ностей стандартные обозначения условных вероятностей, которые получаются экспертным путем или на основе статистических данных. В нашем случае  $\hat{P}(e|h_j) = q_j(j|a_{0,j})$  — это оценки, определяемые формулой (12), где  $p(j|a_{0,j})$  — условные вероятности, вычисляемые на основе правил нечетких продукций с помощью операций нечеткой логики [16].

Свидетельство  $e$  называют подтверждением для гипотезы  $h$  при выполнении неравенства

$$\hat{P}(e|h) \geq \hat{P}(e|\bar{h}) \quad (14)$$

и отвергающим гипотезу  $h$  — в противном случае.

Далее, через  $B_0(h) = \hat{P}(h)$  обозначим априорную оценку достоверности гипотезы  $h$  (в нашем случае это вероятность  $\pi_j$  распространения болезни  $j$ ) и через  $B(h) = \hat{P}(h|e)$  — оценку достоверности  $h$ , вычисленную с учетом представленного свидетельства  $e = a_{0,j}$  (т. е. оценку условной вероятности  $\hat{p}(j|a_{0,j})$ , вычисленную по методике, представленной в работе [16]).

На основании формул (13) имеем

$$\hat{P}^L(e|h) = \frac{p(e)\hat{p}(h|e)}{\hat{p}^R(h)}, \quad \hat{P}^R(e|h) = \frac{p(e)\hat{p}(h|e)}{\hat{p}^L(h)}, \quad (15)$$

$$\hat{P}^L(e|\bar{h}) = \frac{p(e)\hat{p}(\bar{h}|e)}{\hat{p}^R(h)}, \quad \hat{P}^R(e|\bar{h}) = \frac{p(e)\hat{p}(\bar{h}|e)}{\hat{p}^L(h)}, \quad (16)$$

где

$$\hat{p}^L(\bar{h}) = 1 - \hat{p}^R(h), \quad \hat{p}^R(\bar{h}) = 1 - \hat{p}^L(h), \quad (17)$$

$$\hat{p}^L(h) = \pi_j^L, \quad \hat{p}^R(h) = \pi_j^R \text{ при } h = j. \quad (18)$$

Через  $u(e)$  будем обозначать оценку справедливости свидетельства  $e$ :  $0 \leq u(e) \leq 1$ .

Для пересчета оценки достоверности  $B(h)$  гипотезы  $h$  с учетом оценки справедливости  $u(e)$  свидетельства  $e$  положим

$$B(h) = Q(\delta(e, h), B(h), u(e)), \quad (19)$$

где  $Q$  — итеративная функция, в качестве которой используется кусочно-линейная аппроксимация

функции  $P(h|e)$ , принимающая значение  $\hat{P}(h|e)$  в случае полного подтверждения свидетельства  $e$  (при  $u(e) = 1$ ), линейно интерполирующая значения

$B(h)$  и  $\hat{P}(h|e)$  в случае неполного подтверждения свидетельства  $e$  ( $0 < u(e) < 1$ ) и не изменяющая уже имеющееся значение  $B(h)$  в случае полной неопределенности (при  $u(e) = 0$ ):

$$Q(\delta(e, h), B(h), u(e)) = B(h) + u(e)[\hat{P}(h|e) - B(h)], \quad (20)$$

где оценка  $\hat{P}(h|e)$  для условной вероятности  $P(h|e)$  вычисляется по формуле Байеса

$$\hat{P}(h|e) = \frac{B(h)\hat{P}(e|h)}{B(h)\hat{P}(e|h) + (1 - B(h))\hat{P}(e|\bar{h})}. \quad (21)$$

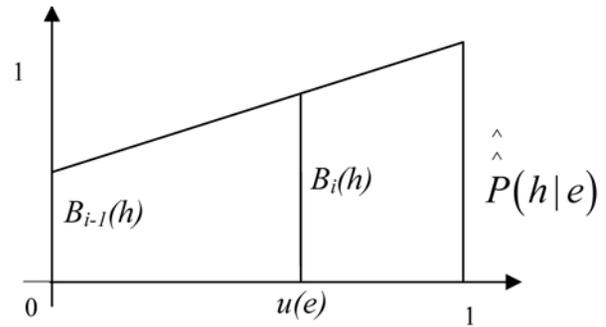


Рис. 1

Принципиальная схема интерполяции оценки для  $B(h)$  (т. е. оценки  $B_i(h)$  на итерации  $i$  по известной оценке  $B_{i-1}(h)$  на предыдущей итерации) в случае  $0 < u(e) < 1$  изображена на рис. 1 [11].

В случае интервального задания вероятностей  $\hat{P}(e|h)$  и  $\hat{P}(e|\bar{h})$

$$\hat{P}^L(e|h) \leq \hat{P}(e|h) \leq \hat{P}^R(e|h),$$

$$\hat{P}^L(e|\bar{h}) \leq \hat{P}(e|\bar{h}) \leq \hat{P}^R(e|\bar{h}) \quad (22)$$

можно провести интервальное оценивание достоверности проверяемой гипотезы  $h$ , т. е. вычислить границы  $B^L(h)$  и  $B^R(h)$  такие, что  $B^L(h) \leq \hat{P}(h) \leq B^R(h)$  по представленному свидетельству  $e$  с учетом силы связи  $e$  с  $h$ :

$$\delta^L(e, h) = (\hat{P}^L(e|h), \hat{P}^L(e|\bar{h})),$$

$$\delta^R(e, h) = (\hat{P}^R(e|h), \hat{P}^R(e|\bar{h})). \quad (23)$$

В качестве информации относительно свидетельства  $e$  могут быть указаны границы  $u^L(e)$  и  $u^R(e)$ , т. е.

$$0 \leq u^L(e) \leq u(e) \leq u^R(e) \leq 1. \quad (24)$$

Для получения формул пересчета  $B^L(h)$  и  $B^R(h)$  перепишем (20) с учетом (21) в виде

$$Q(\delta(e, h), B(h), u(e)) = B(h) + B(h)(1 - B(h))(\hat{P}(e|h) - \hat{P}(e|\bar{h}))u(e)K(\delta(e, h), B(h)), \quad (25)$$

где

$$K(\delta(e, h), B(h)) = \frac{1}{B(h)(\hat{P}(e|h) + (1 - B(h))\hat{P}(e|\bar{h}))}.$$

Для того чтобы в случае задания интервальных оценок (19) выписать соотношения типа (17), выделим три случая.

1. Случай  $\hat{P}^R(e|h) \leq \hat{P}^L(e|\bar{h})$ , т. е. свидетельство  $e$  является отвергающим для гипотезы  $h$  для любых оценок  $\hat{P}(e|h)$ ,  $\hat{P}(e|\bar{h})$  из допустимых интервалов (рис. 2).

Тогда нужные соотношения имеют вид

$$\begin{aligned} B^L(h) &:= Q((\hat{P}^L(e|h), \hat{P}^R(e|\bar{h})), B^L(h), u^R(e)); \\ B^R(h) &:= Q((\hat{P}^R(e|h), \hat{P}^L(e|\bar{h})), B^R(h), u^L(e)). \end{aligned} \quad (26)$$

2. Случай  $\hat{P}^L(e|h) \geq \hat{P}^R(e|\bar{h})$ , т. е. свидетельство  $e$  является подтверждением для гипотезы  $h$  в допустимых интервалах для оценок условных вероятностей (рис. 3).

Формулы для пересчета оценок  $B^L(h)$  и  $B^R(h)$  в этом случае принимают следующий вид:

$$\begin{aligned} B^L(h) &:= Q((\hat{P}^L(e|h), \hat{P}^R(e|\bar{h})), B^L(h), u^L(e)); \\ B^R(h) &:= Q((\hat{P}^R(e|h), \hat{P}^L(e|\bar{h})), B^R(h), u^R(e)). \end{aligned} \quad (27)$$

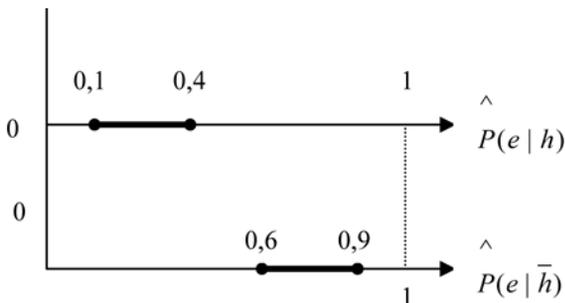


Рис. 2

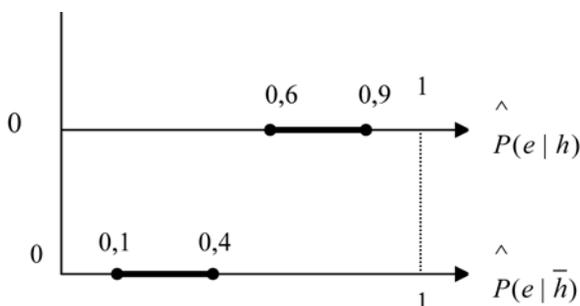


Рис. 3

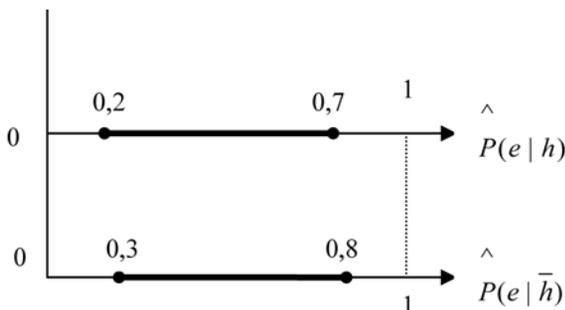


Рис. 4

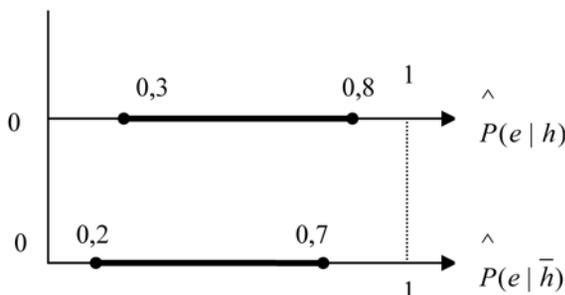


Рис. 5

3. Смешанный случай  $\hat{P}^L(e|h) \leq \hat{P}^R(e|\bar{h})$  и одновременно  $\hat{P}^R(e|h) \geq \hat{P}^L(e|\bar{h})$  включает варианты, представленные на рис. 4, 5.

Формулы для пересчета оценок  $B^L(h)$  и  $B^R(h)$  в случае 3 запишутся в виде

$$\begin{aligned} B^L(h) &:= Q((\hat{P}^L(e|h), \hat{P}^R(e|\bar{h})), B^L(h), u^R(e)); \\ B^R(h) &:= Q((\hat{P}^R(e|h), \hat{P}^L(e|\bar{h})), B^R(h), u^R(e)). \end{aligned} \quad (28)$$

В случае, когда какой-либо из рассмотренных показателей задан точечной оценкой, получаем нужные формулы для пересчета из формул (26)–(28), убрав индексы  $L$  и  $R$  при соответствующих этому показателю левой и правой границ интервала допустимых изменений.

Определим для  $k = 0, 1, 2, \dots$  условные вероятности

$$\begin{aligned} \hat{P}_k^L(e|h) &= \frac{p(e)\hat{p}(h|e)}{B_k^R(h)}, \quad \hat{P}_k^R(e|h) = \frac{p(e)\hat{p}(h|e)}{B_k^L(h)}, \\ \hat{P}_k^L(e|\bar{h}) &= \frac{p(e)\hat{p}(\bar{h}|e)}{B_k^R(\bar{h})}, \quad \hat{P}_k^R(e|\bar{h}) = \frac{p(e)\hat{p}(\bar{h}|e)}{B_k^L(\bar{h})}, \end{aligned} \quad (29)$$

где  $B_k^L(h)$  и  $B_k^R(h)$  при  $k = 0$  есть нижняя и верхняя границы априорной вероятности  $B_0(h)$  гипотезы  $h$ , а при  $k \geq 1$  определяются указанным ниже итеративным способом. Будем считать, что

$$\hat{p}(\bar{h}|e) = 1 - \hat{p}(h|e) \quad (30)$$

и

$$\begin{aligned} B_k^L(\bar{h}) &= 1 - B_k^R(h) B_k^R(\bar{h}) = 1 - B_k^L(h), \\ k &= 0, 1, 2, \dots \end{aligned} \quad (31)$$

Для пересчета оценок  $B_k^L(h)$  и  $B_k^R(h)$  в случае 1 будем использовать итеративную схему, соответствующую формулам (26):

$$\begin{aligned} B_k^L(h) &= B_{k-1}^L(h) + B_{k-1}^L(h) \times \\ &\times (1 - B_{k-1}^L(h)) (\hat{P}_{k-1}^L(e|h) - \hat{P}_{k-1}^R(e|\bar{h})) u^R(e) \times \\ &\times \frac{1}{B_{k-1}^L(h) \hat{P}_{k-1}^L(e|h) + (1 - B_{k-1}^L(h)) \hat{P}_{k-1}^R(e|\bar{h})}; \\ B_k^R(h) &= B_{k-1}^R(h) + B_{k-1}^R(h) \times \\ &\times (1 - B_{k-1}^R(h)) (\hat{P}_{k-1}^R(e|h) - \hat{P}_{k-1}^L(e|\bar{h})) u^L(e) \times \\ &\times \frac{1}{B_{k-1}^R(h) \hat{P}_{k-1}^R(e|h) + (1 - B_{k-1}^R(h)) \hat{P}_{k-1}^L(e|\bar{h})}; \end{aligned} \quad (32)$$

в случае 2 — итеративную схему, соответствующую формулам (27):

$$\begin{aligned}
 B_k^L(h) &= B_{k-1}^L(h) + B_{k-1}^L(h) \times \\
 &\times (1 - B_{k-1}^L(h)) (\hat{P}_{k-1}^L(e|h) - \hat{P}_{k-1}^R(e|\bar{h})) u^L(e) \times \\
 &\times \frac{1}{B_{k-1}^L(h) \hat{P}_{k-1}^L(e|h) + (1 - B_{k-1}^L(h)) \hat{P}_{k-1}^R(e|\bar{h})}; \\
 B_k^R(h) &= B_{k-1}^R(h) + B_{k-1}^R(h) \times \\
 &\times (1 - B_{k-1}^R(h)) (\hat{P}_{k-1}^R(e|h) - \hat{P}_{k-1}^L(e|\bar{h})) u^R(e) \times \\
 &\times \frac{1}{B_{k-1}^R(h) \hat{P}_{k-1}^R(e|h) + (1 - B_{k-1}^R(h)) \hat{P}_{k-1}^L(e|\bar{h})};
 \end{aligned} \quad (33)$$

в случае 3 — итеративную схему, соответствующую формулам (28):

$$\begin{aligned}
 B_k^L(h) &= B_{k-1}^L(h) + B_{k-1}^L(h) \times \\
 &\times (1 - B_{k-1}^L(h)) (\hat{P}_{k-1}^L(e|h) - \hat{P}_{k-1}^R(e|\bar{h})) u^R(e) \times \\
 &\times \frac{1}{B_{k-1}^L(h) \hat{P}_{k-1}^L(e|h) + (1 - B_{k-1}^L(h)) \hat{P}_{k-1}^R(e|\bar{h})}; \\
 B_k^R(h) &= B_{k-1}^R(h) + B_{k-1}^R(h) \times \\
 &\times (1 - B_{k-1}^R(h)) (\hat{P}_{k-1}^R(e|h) - \hat{P}_{k-1}^L(e|\bar{h})) u^R(e) \times \\
 &\times \frac{1}{B_{k-1}^R(h) \hat{P}_{k-1}^R(e|h) + (1 - B_{k-1}^R(h)) \hat{P}_{k-1}^L(e|\bar{h})}.
 \end{aligned} \quad (34)$$

При  $h = j$ ,  $e = a_0$  имеем

$$\hat{p}(h|e) = \hat{p}(j|a_0, j), \quad B_0^L(h) = \pi_j^L, \quad B_0^R(h) = \pi_j^R,$$

так что при  $j = 1$  и уровне значимости  $\alpha = 0,05$  в (9) находим

$$\hat{p}(h|e) = 0,74, \quad \hat{p}(h|\bar{e}) = 0,24, \quad \pi_j^L = 0,165, \quad \pi_j^R = 0,335,$$

и, следовательно, согласно (29):

$$\begin{aligned}
 \hat{P}_0^L(e|h) &= \frac{0,74}{0,335} p(e) = 2,21p(e); \\
 \hat{P}_0^R(e|h) &= \frac{0,74}{0,165} p(e) = 4,48p(e); \\
 \hat{P}_0^L(e|\bar{h}) &= p(e) \frac{0,26}{0,835} = 0,311p(e); \\
 \hat{P}_0^R(e|\bar{h}) &= p(e) \frac{0,26}{0,665} = 0,391p(e).
 \end{aligned} \quad (35)$$

Из (35) при  $p(e) > 0$  находим, что  $\hat{P}_0^R(e|h) > \hat{P}_0^L(e|\bar{h})$  и  $\hat{P}_0^R(e|\bar{h}) > \hat{P}_0^L(e|h)$ , т. е. при  $k = 0$  имеет место случай 3 и следует воспользоваться схемой (34). В расчетах в качестве нижней и верхней границ справедливости  $u(e)$  свидетельства  $e$  нами принялись:  $u^L(e) = 0,7$  и  $u^R(e) = 0,8$ .

При переходе к следующему шагу итераций необходимо предварительно проверить, какой из

случаев 1—3 имеет место, и применять затем соответствующую итеративную схему.

После конечного числа итераций мы получим установившиеся значения величин  $B^L(h)$  и  $B^R(h)$ , которые с вероятностью  $P = 0,95$  определяют доверительный интервал ( $B^L(h)$ ,  $B^R(h)$ ) для достоверности гипотезы  $h$ .

## Заключение

На основе предельных значений априорных вероятностей заболеваний нервно-мышечной системы, полученных в результате многочисленных клинических наблюдений врачей-экспертов относительно распространенности данных заболеваний, установлены с большой доверительной вероятностью (более 95 %) нижние и верхние границы их априорных вероятностей.

С помощью найденных доверительных интервалов для априорных вероятностей заболеваний вычисляются интервальные оценки условных вероятностей значений диагностических признаков, представленных симптомокомплексом исследуемого пациента при условии наличия у него конкретного заболевания из рассматриваемой совокупности.

Путем итеративного применения интервальных байесовских механизмов вывода получают уточненные оценки апостериорных вероятностей заболеваний у исследуемого пациента по заданному симптомокомплексу значений диагностических признаков. Рассмотрены различные случаи подтверждения или отрицания гипотезы о наличии у пациента каждого из рассматриваемых заболеваний при заданном индивидуальном наборе (симптомокомплексе) значений диагностических признаков.

## Список литературы

1. **Команцев В. Н., Заболотных В. А.** Методические основы клинической электронейромиографии. СПб.: Лан, 2001. 460 с.
2. **Николаев С. Г.** Практикум по клинической электромиографии. Иваново: Изд. Иванов. ГосМедАкадемия, 2003. 246 с.
3. **Гехт Б. М.** Теоретическая и клиническая электромиография. Л.: Наука, 1990. 203 с.
4. **Зенков Л. Р., Ронкин М. А.** Функциональная диагностика нервных болезней. М.: Медицина, 1991. 640 с.
5. **Касаткина Л. Ф., Гильванова О. В.** Электромиографические методы исследования в диагностике нервно-мышечных заболеваний. Игольчатая электромиография. М.: Медицина, 2010. 416 с.
6. **Фомин Я. А., Тарловский Г. Р.** Статистическая теория распознавания образов. М.: Радио и связь, 1986. 264 с.
7. **Локтюхин В. Н., Мальченко С. И., Черепнин А. А.** Основы математического обеспечения поддержки диагностических решений в биотехнических системах с использованием нечеткой логики. Рязань. Изд-во Рязан. гос. радиотех. унта, 2009. 64 с.
8. **Черепнин А. А.** Модели, алгоритмы и средства поддержки принятия диагностических решений при эндоскопическом обследовании на основе технологии нечеткой логики. Дис. ... канд. техн. наук. Рязань, 2010. 169 с.
9. **Заложенкова И. А., Парасюк И. Н.** Интегрированная система проверки гипотез: программно-алгоритмические средства // Кибернетика и системный анализ. 1997. № 5. С. 50—58.
10. **Верева О. В., Заложенкова И. А., Парасюк И. Н.** Обобщение интервальных байесовских механизмов вывода и перспективы их использования // Кибернетика и системный анализ. 1998. № 6. С. 3—13.

11. **Верева О. В., Заложенкова И. А., Парасюк И. Н.** Интервальные байесовские механизмы вывода и их приложения // Проблемы программирования. 2000. № 1–2. С. 467–470.

12. **Lewis R. J., Wears R. L.** An introduction to the Bayesian analysis of clinical trials // *Ann. Emerg. Med.* 1993. V. 22. P. 1328–1336.

13. **Abrams K., Ashby D., Errington D.** Simple Bayesian analysis of clinical trials: a tutorial // *Control Clin. Trials.* 1994. V. 15. P. 349–359.

14. **Spiegelhalter D. J., Myles J. P., Jones D. R., Abrams K. R.** Bayesian methods in health technology assessment: a review // *Health Technol. Assess.* 2000. V. 4. P. 1–30.

15. **Жмудяк М. Л., Поваляхин А. Н., Стребуков А. В.** и др. Диагностика заболеваний методами теории вероятностей. Барнаул: Изд-во АлтГТУ, 2006. 168 с.

16. **Абдуллаев Н. Т., Дышин О. А., Абдуллаева Г. Э.** Нетрадиционное использование байесовских формул для дифферен-

циальной диагностики нервно-мышечных заболеваний // *Биомедицинская радиоэлектроника.* 2014. № 1. С. 50–58.

17. **Зельнер А.** Байесовские методы в эконометрии / Пер. с англ. М.: Статистика, 1980. 438 с.

18. **Хей Дж.** Введение в методы байесовского статистического вывода / Пер. с англ. М.: Финансы и статистика, 1987. 335 с.

19. **Голодников А. Н., Кнопов П. С., Пепеляев В. А.** Оценивание параметров надежности при наличии неполной первичной информации // *Компьютерная математика.* 2003. № 1. С. 36–47.

20. **Айвазян С. А., Енюков И. С., Мешалкин Л. Д.** Прикладная статистика: Основы моделирования и первичная обработка информации. Справочное изд. М.: Финансы и статистика, 1983. 471 с.

**N. T. Abdullaev<sup>1</sup>**, Associate Professor, e-mail: a.namik49@mail.ru,  
**O. A. Dyshin<sup>2</sup>**, Senior Researcher, **G. E. Abdullaeva<sup>2</sup>**, Magister  
<sup>1</sup> Azerbaijan Technical University, <sup>2</sup> Azerbaijan State Oil Academy

## Interval Bayesian Inference in the Differential Diagnosis of Neuromuscular Diseases

*In practice rather difficult to point estimates diagnostic features that affect the presence of a particular type of disease. Much easier with sufficient accuracy to indicate intervals containing the desired evaluation. This determines the usefulness of the Bayesian approach for cases with interval uncertainty. Based on thresholds priors considered neuro — muscular diseases with a large installed a confidence level of the lower and upper limits of their priors. Using found confidence intervals for the a priori probabilities are calculated diseases interval estimates of conditional probabilities diagnostic features.*

*Iterative application of generalized integral Bayesian schemes leads to updated estimates of confidence diseases studied patient for a given complex of symptoms and diagnostic features allows us to consider different possible cases to confirm or deny the hypothesis that the patient of a disease of the given set of neuromuscular diseases.*

*In this paper, this approach is used for the differential diagnosis of common neuromuscular diseases such as demyelinating polyneuropathy, carpal tunnel syndrome and cubital tunnel syndrome.*

**Keywords:** differential diagnostics, diagnostic signs, electromyography, Bayesian formula, interval assessment

### References

1. **Komantsev V. N., Zabolotnykh V. A.** *Metodicheskiye osnovy klinicheskoy elektroneyromiografii.* SPb: Lan, 2001. 460 p.

2. **Nikolayev S. G.** *Praktikum po klinicheskoy elektroniografii.* Ivanovo: Ivanov. GosMedAkademiya, 2003. 246 p.

3. **Gekht B. M.** *Teoreticheskaya i klinicheskaya elektroniografiya.* L.: Nauka, 1990. 203 p.

4. **Zenkov L. R., Ronkin M. A.** *Funktsionalnaya diagnostika nervnykh bolezney.* M.: Meditsina, 1991. 640 p.

5. **Kasatkina L. F., Gilvanova O. V.** *Elektromiograficheskiye metody issledovaniya v diagnostike nervno-myshechnykh zabolevaniy. Igolchataya elektroniografiya.* M.: Meditsina, 2010. 416 p.

6. **Fomin Ya. A., Tarlovsky G. R.** *Statisticheskaya teoriya raspoznavaniya obrazov.* M.: Radio i svyaz, 1986. 264 p.

7. **Loktyukhin V. N., Malchenko S. I., Cherepnin A. A.** *Osnovy matematicheskogo obespecheniya podderzhki diagnosticheskikh resheniy v biotekhnicheskikh sistemakh s ispolzovaniyem nechetkoy logiki.* Ryazan: Izd-vo Ryazan. gos. radiotekh. un-ta, 2009. 64 p.

8. **Cherepnin A. A.** *Modeli, algoritmy i sredstva podderzhki prinyatiya diagnosticheskikh resheniy pri endoskopicheskom obsledovanii na osnove tekhnologii nechetkoy logiki.* Dis. ... kand. tekhn. nauk. Ryazan, 2010. 169 p.

9. **Zalozhenkova I. A., Parasyuk I. N.** Integriruyannaya sistema proverki gipotez: programmno-algoritmicheskiye sredstva. *Kibernetika i sistemnyy analiz.* 1997. N 5. P. 50–58.

10. **Vereva O. V., Zalozhenkova I. A., Parasyuk I. N.** Obobshcheniye intervalnykh bayesovskikh mekhanizmov vyvoda i perspektivy ikh ispolzovaniya. *Kibernetika i sistemnyy analiz.* 1998. N. 6. P. 3–13.

11. **Vereva O. V., Zalozhenkova I. A., Parasyuk I. N.** Intervalnye bayesovskiy mekhanizmy vyvoda i ikh prilozheniya. *Problemy programmirovaniya.* 2000. N. 1–2. P. 467–470.

12. **Lewis R. J., Wears R. L.** An introduction to the Bayesian analysis of clinical trals. *Ann. Emerg. Med.* 1993. V. 22. P. 1328–1336.

13. **Abrams K., Ashby D., Errington D.** Simple Bayesian analysis of clinical trials: a tutorial. *Control. Clin. Trals.* 1994. V. 15. P. 349–359.

14. **Spiegelhalter D. J., Myles J. P., Jones D. R., Abrams K. R.** Bayesian methods in health technology assessment: a review. *Health Technol. Assess.* 2000. V. 4. P. 1–30.

15. **Zhmudyak M. L., Povalikhin A. N., Strebukov A. V.** i dr. *Diagnostika zabolevaniy metodami teorii veroyatnostey.* Barnaul, Izd-vo AltGTU, 2006. 168 p.

16. **Abdullayev N. T., Dyshin O. A., Abdullayeva G. E.** Netraditsionnoye ispolzovaniye bayesovskikh formul dlya differentsialnoy diagnostiki nervno-myshechnykh zabolevaniy. *Biomeditsinskaya radioelektronika.* 2014. N. 1. P. 50–58.

17. **Zelner A.** *Bayesovskiy metody v ekonometrii.* Per s angl. M.: Statistika, 1980. 438 p.

18. **Khey Dzh.** *Vvedeniye v metody bayesovskogo statisticheskogo vyvoda.* Per. s angl. M.: Finansy i statistika, 1987. 335 p.

19. **Golodnikov A. N., Knopov P. S., Pepelyaev V. A.** Otsenivaniye parametrov nadezhnosti pri nalichii nepolnoy pervichnoy informatsii. *Kompyuternaya matematika.* 2003. N. 1. P. 36–47.

20. **Ayvazyan S. A., Yenyukov I. S., Meshalkin L. D.** *Prikladnaya statistika: Osnovy modelirovaniya i pervichnaya obrabotka informatsii.* Spravochnoye izd. M.: Finansy i statistika, 1983. 471 p.

УДК 004.023

А. М. Марченко<sup>1</sup>, д-р техн. наук, проф., Е. А. Попов<sup>1</sup>, инженер,  
В. Ю. Савченко<sup>2</sup>, аспирант, e-mail: valentine.savchenko@gmail.com,  
<sup>1</sup> Московский государственный университет им. М. В. Ломоносова  
<sup>2</sup> Национальный исследовательский университет "МИЭТ"

## Трассировка соединений в нанометровых стандартных ячейках с помощью шаблонов

*Метод стандартных ячеек широко применяется для проектирования цифровых интегральных схем специального назначения (ASIC). Стандартная ячейка представляет собой набор специальным образом соединенных транзисторов, выполняющий булеву функцию или функцию хранения. Этап топологического проектирования ASIC является неотъемлемой частью метода стандартных ячеек и состоит в получении геометрического представления (топологии) схемы на основе геометрического представления библиотечных элементов. Рассмотрены вопросы, связанные с синтезом топологии стандартных ячеек. Предложен метод трассировки соединений в библиотечном элементе, учитывающий трудно формализуемый критерий качества топологии — доступность портов для подключения на уровне схемы. Предложены модели для трассировки, позволяющие учитывать конструкторско-технологические ограничения 32-нанометрового семейства технологий.*

**Ключевые слова:** синтез топологии, трассировка, стандартные ячейки

### Введение

Типовой маршрут, используемый в САПР синтеза библиотек стандартных ячеек, состоит из нескольких этапов, позволяющих перейти от представления ячейки в виде списка соединений (*netlist*) к представлению в виде топологии (*layout*). Этими этапами являются [1]: определение размеров транзисторов (*transistor folding*), размещение транзисторов, трассировка соединений и сжатие топологии с учетом всех правил проектирования.

Для успешного синтеза топологии на технологиях до 45 нм достаточно размещать и трассировать транзисторы, представляя топологию с помощью штрих-диаграммы (*stick diagram*) [2]. Штрих-диаграмма — это упрощенное представление топологии, показывающее взаимное расположение транзисторов и соединений. При таком подходе учет подавляющего числа правил проектирования начинается на этапе сжатия, а штрих-диаграмма учитывает только суммарное число трассировочных треков. Модуль сжатия получает на вход штрих-диаграмму, преобразует ее в топологию путем масштабирования, сдвигает элементы топологии таким образом, чтобы высота ячейки удовлетворяла требованиям шаблона библиотеки, модифицирует элементы топологии в целях минимизации ширины ячейки и, наконец, создает порты.

Начиная с технологии 32 нм, имеются критические правила, игнорирование которых на этапе трассировки делает невозможным успешное сжатие топологии. Так как штрих-диаграмма не учитывает

правила проектирования, необходима модель, которая, с одной стороны, является упрощением топологии, а с другой — позволяет вести учет критических правил проектирования. Модель, отвечающая данным требованиям, предложена в разделе 1.1.

Важным критерием качества трассировки ячейки является доступность портов для подключения на уровне микросхемы [3]. Этот критерий противоречит суммарной длине соединений — другому важному критерию, используемому при трассировке [4]. В разделе 3 предложены методы, учитывающие доступность портов при трассировке стандартной ячейки.

### 1. Модели для трассировки

Технология 32 нм характерна тем, что на слое, в котором лежат затворы транзисторов, определен шаг (*pitch*) между соединениями. Данное ограничение приводит к тому, что размещение транзисторов полностью определяет ширину ячейки.

#### 1.1. Модель детальной трассировки

В качестве модели детальной трассировки используется **решетчатый граф**. Граф строится на основе размещения транзисторов. Для построения графа каждому трассировочному слою ставится в соответствие прямоугольная сетка. Сетка моделируется подмножеством вершин и ребер графа. Вершины соответствуют пересечениям вертикальных и горизонтальных линий сетки, ребра — сегментам линий, заключенным между соседними вершинами. Каждая вершина хранит слой и координату пе-

ресека. Если вершина находится на слое затворов или слое одной из диффузий и лежит на середине терминала транзистора, то цепь этого терминала также хранится в вершине. Межслойные переходы моделируются с помощью ребер. Межслойные ребра создаются между вершинами из соседних слоев, координаты которых совпадают.

На рис. 1 (см. третью сторону обложки) приведен разрез графа для некоторого размещения транзисторов. Граф построен для четырех слоев:  $N$ -диффузии, затворов, первого и второго металлов. Вершины представлены окружностями, ребра — линиями. При построении графа горизонтальный шаг сетки на слоях  $N$ -диффузии, затворов и второго металла выбран равным расстоянию между центрами затворов, а горизонтальный шаг на слое первого металла равен половине расстояния между затворами.

### 1.2. Модель глобальной трассировки

Планирование соединений осуществляется в трех **регионах**:  $P$ -области,  $N$ -области и области затворов. На рис. 2 (см. третью сторону обложки) показаны регионы на примере некоторого размещения.

Трассировочный ресурс диффузионной области моделируется с помощью **символьной таблицы**. Таблица строится на основе графа. Слой, сетка которого имеет наибольшее число вертикальных линий, определяет число столбцов таблицы. Каждой вертикальной линии такого слоя ставится в соответствие столбец. Со столбцом ассоциируются индекс (порядковый номер),  $X$ -координата линии, цепь и тип терминала транзистора. Строка таблицы показывает, как используются горизонтальный и вертикальный трассировочные ресурсы в области на слое.

Ячейка таблицы содержит следующую информацию:

- максимальный вертикальный ресурс;
- вертикальные ресурсы каждой цепи, спланированной через область;
- цепь, чей вертикальный сегмент проходит над областью;
- пометку о том, покрывает область портовую сетку (параметр шаблона библиотеки) или нет.

На рис. 3 (см. третью сторону обложки) приведен фрагмент таблицы для  $N$ -области некоторой ячейки. Шаблон библиотеки допускает трассировку в слоях  $P$ - и  $N$ -диффузий, затворов, первого и второго металлов. Согласно шаблону, вертикальные линии портовой сетки проходят между вертикальными соединениями слоя затворов на всех слоях. Например, со столбцом с индексом 3 ассоциированы абсцисса 150, цепь  $VSS$  и тип  $d$  (сток). Ячейка  $S_{22}$ , показывающая использование ресурса на первом металле в области столбца с индексом 2, содержит максимальный вертикальный ресурс — 140, вертикальный ресурс для цепи  $VSS$  — 55, вертикальный ресурс для цепи  $QN$  — 45, цепь  $QN$  и пометку о том, что область лежит вне портовой сетки:

$$S_{22} = \{140, ((VSS, 55), (QN, 45)), QN, \text{off grid}\}.$$

## 2. Трассировка соединений

Метод трассировки основан на практическом наблюдении о возможности отдельного планирования соединений в диффузионных областях и области затворов. Трассировка отдельной цепи осуществляется с помощью нескольких эвристических алгоритмов. Каждый алгоритм реализует способ трассировки (шаблон), используемый при "ручном" проектировании. Шаблон применим для цепи, если она имеет определенный набор терминалов.

### 2.1. Классификация цепей

Соединения, создаваемые в ходе трассировки, подключаются к терминалам трех типов: истоку, затвору, а также стоку. Транзисторы бывают двух видов:  $p$  и  $n$ . Различают пять типов цепей [5]:

- $P$  — цепь, соединяющая только стоки или истоки  $p$ -транзисторов;
- $N$  — цепь, соединяющая только стоки или истоки  $n$ -транзисторов;
- $G$  — цепь, соединяющая только затворы транзисторов;
- $PN$  — цепь, соединяющая стоки или истоки  $p$ - и  $n$ -транзисторов;
- $PNG$  — цепь, соединяющая стоки или истоки  $p$ - и  $n$ -транзисторов и затворы.

Цепи  $P$ ,  $N$ ,  $PN$  и  $PNG$  планируют в диффузионных областях,  $G$  и  $PNG$  — в затворной области. Например, цепи, принадлежащие топологии, показанной на рис. 4 (см. третью сторону обложки), классифицируются следующим образом:  $G$ -цепи —  $S$ ,  $A$  и  $B$ ;  $P$ -цепи —  $P1$  и  $P2$ ;  $N$ -цепи —  $N1$  и  $N2$ ;  $PN$ -цепь —  $O$ ;  $PNG$ -цепь —  $I1$ .

### 2.2. Алгоритм трассировки соединений

Алгоритм трассировки соединений является итеративным (рис. 5). В каждой итерации выбирают набор шаблонов и порядок трассировки цепей в диффузионных областях (строки 03—06). Итерация (строки 02—30) состоит из двух этапов. Каждый этап содержит две фазы. На первой фазе осуществляется отдельное планирование соединений в затворном регионе и диффузионных областях. Решения для регионов проверяются на совместность с помощью детальной трассировки на второй фазе. На первом этапе (строки 07—17) проверяется возможность трассировки соединений без создания портов согласно выбранному порядку и набору шаблонов. На втором этапе (строки 18—30) планируются входные порты.

В качестве финального решения (строки 31—35) выбирается план, позволяющий использовать наименьшее число слоев и имеющий наибольшее число портов.

Возможны случаи, когда не удастся завершить трассировку соединений, в этом случае полезно предоставить частично трассированную топологию. Частично трассированная топология может повысить производительность пользователя, если время, необходимое для завершения трассировки вручную, меньше времени, необходимого для ручной трассировки.

**Алгоритм** ТрассироватьСоединения  
**Вход:**  $T$  – размещение  
**Выход:**  $T'$  – топология  
00:  $S_p = \emptyset$  – множество годных решений  
01:  $S_p - \emptyset$  – множество несовместных решений  
02: **выполнять до прерывания**  
03: сгенерировать порядок трассировки диффузионных цепей  $O$   
04: сгенерировать набор шаблонов  $P$  для порядка  $O$   
05: **если** пара  $(O, P)$  уже исследовалась  
06: **прервать итерации**  
07: спланировать затворные сегменты  $S_o$ , учесть шаблоны  $P$   
08: **если**  $S_p = \emptyset$   
09: **перейти к следующей итерации**  
10: оценить ресурс  $R_p$ , необходимый для сегментов  $S_p$   
11: спланировать диффузионные сегменты  $S_d$  согласно порядку  $O$  и шаблону  $P$ , учесть сегменты  $S_p$  и ресурс  $R_p$   
12: **если**  $S_d = \emptyset$   
13: **перейти к следующей итерации**  
14: провести детальную трассировку сегментов  $S_p$  и  $S_d$   
15: **если** детальная трассировка не удалась  
16:  $S_p = S_p \cup (S_d, S_p)$   
17: **перейти к следующей итерации**  
18: оценить ресурс  $R_d$ , необходимый для сегментов  $S_d$   
19: спланировать затворные сегменты  $S_p$  и входные порты  $P_i$ , учесть шаблоны  $P$  и ресурс  $R_d$   
20: **если**  $S_p = \emptyset$   
21: **перейти к следующей итерации**  
22: оценить ресурс  $R_p$ , необходимый для сегментов  $S_p$   
23: спланировать диффузионные сегменты  $S_d$  согласно порядку  $O$  и шаблону  $P$ , учесть сегменты  $S_p$ , ресурс  $R_p$  и порты  $P_i$   
24: **если**  $S_d = \emptyset$   
25: **перейти к следующей итерации**  
26: провести детальную трассировку сегментов  $S_p$  и  $S_d$ , а также портов  $P_i$   
27: **если** детальная трассировка не удалась  
28:  $S_p = S_p \cup (S_d, S_p, P_i)$   
29: **перейти к следующей итерации**  
30:  $S_o = S_o \cup (S_d, S_p, P_i)$   
31: **если**  $S_p = \emptyset$   
32: выбрать из решений  $S_p$  какое-нибудь решение  $S$   
33: **иначе**  
34: выбрать из решений  $S_p$  решение  $S$  с наибольшим числом портов и наименьшим числом сегментов в высоких слоях  
35: провести детальную трассировку размещения  $T$  в соответствии с решением  $S$ , тем самым получив топологию  $T'$

Рис. 5. Алгоритм трассировки соединений в стандартной ячейке

Для сокращения числа итераций используют двухуровневое ранжирование цепей. На первом уровне цепи упорядочиваются по типу двумя способами:  $\{P, N, PNG, PN\}$  и  $\{P, N, PN, PNG\}$ . На втором уровне цепи одного типа упорядочиваются относительно друг друга. В худшем случае планировщик исследует  $2(|P| + |N| + |PN| + |PNG|)$  порядков цепей. В силу того, что только  $PNG$  цепи имеют больше одного шаблона, для каждого порядка необходимо сгенерировать  $3|PNG|$  наборов шаблонов.

### 2.3. План трассировки соединений

План трассировки цепи описывается множеством сегментов. Сегмент характеризуется набором терминалов и слоем. Терминал имеет три параметра: регион (см. рис. 2), слой и индекс столбца символической таблицы (см. раздел 1.2). В результате планирования затворных (рис. 5, строки 07 и 19) и диффузионных (рис. 5, строки 11 и 23) сегментов для каждой цепи определяется план и порядок детальной трассировки сегментов в каждом регионе.

На рис. 6 (см. четвертую сторону обложки) приведен план трассировки соединений для топологии, показанной на рис. 4. Порядок детальной трассировки сегментов в  $P$ -регионе:  $\{O, P2, P1, L\}$ . Сегмент цепи  $O$ , принадлежащий  $P$ -региону, имеет три терминала:  $(p, 13, P \text{ диф.})$ ,  $(p, 17, \text{мет. 2})$  и  $(p, 21, P \text{ диф.})$ .

Горизонтальный сегмент  $N$ -региона, принадлежащий цепи  $L$ :  $\{(n, 1, N \text{ диф.}), (n, 8, \text{мет. 1})\}$ , мет. 1). Вертикальный сегмент  $P$ -региона, принадлежащий цепи  $L$ :  $\{(p, 8, \text{мет. 1}), (g, 8, \text{мет. 1})\}$ , мет. 1). Межрегиональный вертикальный сегмент цепи  $O$ :  $\{(p, 17, \text{мет. 2}), (n, 17, \text{мет. 2})\}$ , мет. 2).

### 2.4. Планирование соединений в области затворов

Входными данными для задачи планирования затворных сегментов являются упорядоченный набор транзисторов и множество шаблонов для  $PNG$  цепей. Цель задачи состоит в построении упорядоченного множества затворных сегментов, использующего минимальный вертикальный ресурс. Ресурс, использованный для диффузионных сегментов, дополняет (см. рис. 5, строка 19) исходные данные на второй фазе итерации. На рис. 7 (см. четвертую сторону обложки) показан вертикальный ресурс, необходимый для затворных сегментов на слое первого металла.

## 3. Планирование соединений в диффузионных регионах

### 3.1. Алгоритм планирования диффузионных сегментов

Процедура планирования диффузионных сегментов (рис. 8), используемая в алгоритме трассировки соединений (см. рис. 5, строки 11 и 23), состоит из двух частей. В первой части (см. рис. 8, строки 00–09) цепи обрабатываются согласно их порядку. Для планирования цепи применяется алгоритм, реализующий выбранный шаблон. Во второй части (см. рис. 8, строка 10) определяется порядок детальной трассировки сегментов в каждой области.

**Алгоритм** СпланироватьДиффузионныеСегменты  
**Вход:**  $H_p$  – символическая таблица  $p$ -региона,  $H_n$  – символическая таблица  $n$ -региона,  $O$  – упорядоченное множество цепей,  $P$  – шаблоны трассировки для каждой цепи,  $S_p$  – упорядоченное множество затворных сегментов,  $R_p$  – вертикальный ресурс затворных сегментов и  $P_i$  – множество входных портов  
**Выход:**  $S_d = (S_d^p, S_d^n, S_d^{pn})$  – упорядоченное множество горизонтальных сегментов в  $p$ -регионе, упорядоченное множество горизонтальных сегментов в  $n$ -регионе, вертикальные сегменты  $k$  затворам и вертикальные сегменты между диффузиями  
00:  $S_d = \emptyset$   
01: зарезервировать ресурс  $R_p$  в таблицах  $H_p$  и  $H_n$   
02: **для каждой** цепи  $o \in O$   
03: извлечь шаблон  $p$  для цепи  $o$  из множества  $P$   
04: спланировать горизонтальные  $S_h$  и вертикальные  $S_v$  сегменты цепи  $o$  согласно шаблону  $p$ , учесть сегменты  $S_p$  и порты  $P_i$   
05: **если**  $S_h = \emptyset$   
06:  $S_d = \emptyset$   
07: **завершить алгоритм**  
08: добавить сегменты  $S_h$  в множества  $S_d^p$  и  $S_d^n$   
09: добавить сегменты  $S_v$  в множества  $S_d^p$  и  $S_d^{pn}$   
10: упорядочить множества  $S_d^p$  и  $S_d^n$

Рис. 8. Алгоритм планирования диффузионных сегментов

### 3.2. Шаблоны трассировки $P$ - и $N$ -цепей

Для трассировки  $P$ - и  $N$ -цепей применяется два шаблона.

Первый шаблон предназначен для цепей земли и питания (рис. 9). Согласно шаблону каждый терминал цепи подключается к шине через ближайший слой (обычно это слой первого металла). В случае, когда шаблоном библиотеки разрешена трассировка в диффузии, подключение осуществляется через нее.

**Алгоритм ПрименитьШаблонШины**  
**Вход:**  $H$  – символьная таблица региона и  $N$  – цепь  
**Выход:**  $S_v$  – множество вертикальных сегментов  
00:  $S_v = \emptyset$   
01:  $r$  – регион, соответствующий таблице  $H$   
02:  $l_{rail}$  – слой шины, соответствующей региону  $r$   
03:  $l_{diffusion}$  – слой диффузии, соответствующей региону  $r$   
04:  $l_{lower}$  – слой, ближайший к слою  $l_{diffusion}$   
05: для каждого столбца  $c \in H$   
06:  $n = \text{net}(c)$  – определить цепь столбца  $c$   
07: если  $n = N$   
08:  $i = \text{index}(c)$  – определить индекс столбца  $c$   
09:  $s = \{(r, l_{diffusion}), (r, l_{rail}), l_{lower}\}$   
10: оценить ресурс  $R_s$ , необходимый для сегмента  $s$   
11: если в таблице  $H$  нет свободного ресурса  $R_s$   
12:  $S_v = \emptyset$   
13: завершить алгоритм  
14: зарезервировать ресурс  $R_s$  в таблице  $H$   
15:  $S_v = S_v \cup s$

Рис. 9. Алгоритм трассировки цепей земли и питания

На рис. 10 (см. четвертую сторону обложки) приведен фрагмент топологии, полученный с помощью алгоритма трассировки цепей земли и питания (см. рис. 9). Шина питания расположена во втором металле, ближайший к диффузии слой – первый металл, поэтому созданы переходы из диффузии в первый металл, вертикальные провода в первом металле и переходы из первого металла во второй.

Второй шаблон применяется ко всем остальным  $P$ - и  $N$ -цепям. Шаблон предписывает (рис. 11) соединить все терминалы цепи одним горизонтальным сегментом в первом металле от самого левого терминала до самого правого. Соединения  $P1$ ,  $P2$ ,  $M1$  и  $N2$  на топологии, показанной на рис. 4, созданы с помощью алгоритма трассировки  $P$ - и  $N$ -цепей (рис. 12).

**Алгоритм ОпределитьПараметрыГоризонтальногоСегмента**  
**Вход:**  $H$  – символьная таблица региона и  $N$  – цепь  
**Выход:**  $s$  – горизонтальный сегмент  
00:  $T = \emptyset$  – множество терминалов  
01:  $r$  – регион, соответствующий таблице  $H$   
02:  $l_{diffusion}$  – слой диффузии, соответствующей региону  $r$   
03: для каждого столбца  $c \in H$   
04: если  $\text{net}(c) = N$   
05:  $T = T \cup (r, \text{index}(c), l_{diffusion})$   
06:  $l_{m1}$  – слой первого металла  
07:  $s = (T, l_{m1})$

Рис. 11. Алгоритм определения параметров для горизонтального сегмента

**Алгоритм ПрименитьОбщийШаблонPNЦепей**  
**Вход:**  $H$  – символьная таблица региона и  $N$  – цепь  
**Выход:**  $S_h$  – множество горизонтальных сегментов  
00:  $S_h = \emptyset$   
01:  $s$  = ОпределитьПараметрыГоризонтальногоСегмента( $H, N$ )  
02: оценить ресурс  $R_s$ , необходимый для сегмента  $s$   
03: если в таблице  $H$  имеется свободный ресурс  $R_s$   
04: зарезервировать ресурс  $R_s$  в таблице  $H$   
05:  $S_h = S_h \cup s$

Рис. 12. Алгоритм трассировки  $P$ - и  $N$ -цепей

### 3.5. Шаблон трассировки $PN$ -цепей

Для трассировки  $PN$ -цепей используется один шаблон (рис. 13). Шаблон применяется в два этапа.

Цель первого этапа (рис. 13, строки 00–09) состоит в создании горизонтальных сегментов. На этом этапе каждый регион обрабатывается незави-

**Алгоритм ПрименитьШаблонPNЦепей**  
**Вход:**  $H_p$  – символьная таблица -региона,  $H_n$  – символьная таблица  $N$ -региона,  $S_g$  – упорядоченное множество затворных сегментов,  $P_i$  – множество входных портов и  $N$  – цепь  
**Выход:**  $S_h^p$  – множество горизонтальных сегментов -региона,  $S_h^n$  – множество горизонтальных сегментов  $N$ -региона и  $S_v^{pn}$  – множество вертикальных сегментов  
00:  $S_h^p = S_h^n = S_v^{pn} = \emptyset$   
01:  $s$  = ОпределитьПараметрыГоризонтальногоСегмента( $H_p, N$ )  
02:  $S_h^p$  = СпланироватьГоризонтальныеСегменты( $H_p, N, s$ )  
03: если  $S_h^p = \emptyset$   
04: завершить алгоритм  
05:  $s$  = ОпределитьПараметрыГоризонтальногоСегмента( $H_n, N$ )  
06:  $S_h^n$  = СпланироватьГоризонтальныеСегменты( $H_n, N, s$ )  
07: если  $S_h^n = \emptyset$   
08:  $S_v^{pn} = \emptyset$   
09: завершить алгоритм  
10:  $S_v^{pn}$  = СпланироватьВертикальныеСегментыPNЦепи( $H_p, H_n, N, S_g, P_i, S_h^p, S_h^n$ )  
11: если  $S_v^{pn} = \emptyset$   
12:  $S_v^{pn} = \emptyset$   
13:  $S_v^{pn} = \emptyset$

Рис. 13. Алгоритм трассировки  $PN$ -цепей

**Алгоритм СпланироватьГоризонтальныеСегменты**  
**Вход:**  $H$  – символьная таблица региона,  $N$  – цепь и  $s$  – начальный сегмент  
**Выход:**  $S_h$  – множество горизонтальных сегментов  
00:  $S_h = \emptyset$   
01:  $S' = s$  – множество сегментов-кандидатов, требующих проверки наличия ресурса  
02: пока  $S' \neq \emptyset$   
03: извлечь первый сегмент  $s' = (T', l')$  из множества  $S'$   
04: оценить ресурс  $R_{s'}$ , требуемый для сегмента  $s'$   
05: с помощью таблицы  $H$  выявить индексы столбцов  $O$ , блокирующих создание сегмента  $s'$   
06: если  $O = \emptyset$   
07: зарезервировать ресурс  $R_{s'}$  в таблице  $H$   
08:  $S_h = S_h \cup s'$   
09: перейти к следующей итерации  
10: иначе если  $\exists (t^i, l^i, l^i) \in T^i: l^i \in O$   
11:  $S_h = \emptyset$   
12: завершить алгоритм  
13: для каждого индекса  $a \in O$   
14:  $T^{i+1} = \{(t^i, l^i, l^i) \in T^i: l^i < a\}$   
15: если  $T^{i+1} \neq \emptyset$   
16:  $T^i = T^i \setminus T^{i+1}$   
17:  $S^i = S^i \cup (T^{i+1}, l)$

Рис. 14. Алгоритм создания горизонтальных сегментов

симо. Сегменты планируются в первом металле. Терминалы цепи соединяются наименьшим числом горизонтальных сегментов. В простейшем случае достаточно одного сегмента. Дополнительные сегменты возникают, если множество терминалов не может быть соединено одним сегментом из-за отсутствия свободного ресурса в ячейках, лежащих между терминалами (рис. 14, строки 13–17).

На втором этапе (см. рис. 13, строки 10–13) горизонтальные сегменты, принадлежащие противоположным областям, соединяются с помощью вертикальных сегментов. Горизонтальный сегмент подключается к максимальному числу смежных сегментов противоположной области. Для повышения доступности портов вертикальному сегменту выбирается как можно более низкий слой (рис. 15, строка 01) и столбец, лежащий на портовой сетке и не содержащий входных портов (рис. 15, строка 03). Для сокращения длины соединения подбирается столбец, расположенный ближе к горизонтальным сегментам (рис. 15, строка 02). Для улучшения электрических характеристик соединений (рис. 15, строка 04) отдается предпочтение столбцам, лежащим ближе к центрам соединяемых сегментов.

В результате соединения пары горизонтальных сегментов в каждый элемент этой пары добавляется по одному вспомогательному терминалу (рис. 15, строки 09 и 11). Вспомогательный терминал показывает столбец и слой, в котором горизонтальный сегмент соединяется с вертикальным.

Пример использования алгоритма создания вертикальных сегментов  $PN$ -цепей (рис. 16) приведен на рис. 17. Горизонтальные сегменты показаны с помощью отрезков без стрелок.  $P$ -сегменты распо-

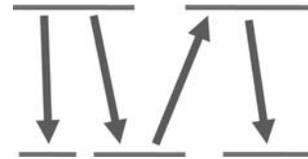


Рис. 17. Пример использования алгоритма создания вертикальных сегментов  $PN$ -цепей (рис. 16)

жены сверху, а  $N$  — снизу. Вертикальные сегменты изображены отрезками со стрелкой, направленной на сегмент, к которому производилось подключение.

### 3.4. Шаблоны трассировки $PNG$ -цепей

Для трассировки  $PNG$ -цепей применяется три шаблона. Шаблоны описываются общим алгоритмом, представленным на рис. 18. Алгоритм состоит из трех фаз.

На первой фазе (рис. 18, строки 00—01) создаются горизонтальные сегменты в каждой из диффузионных областей. Области обрабатываются независимо. Сегменты планируются в первом металле. Терминалы цепи соединяются наименьшим числом сегментов. Особенность  $PNG$ -цепей состоит в том, что они имеют горизонтальные сегменты как в затворном регионе, так и в диффузионных областях. Для улучшения электрических характеристик желательно, чтобы проекции этих сегментов не пересекались. Для этого осуществляется предварительное определение параметров сегментов (рис. 19), в ходе которого выявляются последовательности идущих подряд диффузионных терминалов. Каждая последовательность обрабатывается (рис. 20,

```

Алгоритм СпланироватьВертикальныйСегментМежду
ДиффузионнымиРегионами
Вход:  $H_p$  — символьная таблица  $p$ -региона,  $H_n$  — символьная таблица  $N$ -региона,  $N$  —
цепь,  $S_p$  — упорядоченное множество затворных сегментов,  $P_i$  — множество входных
портов,  $p = (r_p, l_p)$  — горизонтальный  $P$  сегмент и  $n = (r_n, l_n)$  — горизонтальный  $N$ 
сегмент
Выход:  $pn$  — вертикальный сегмент
00:  $Q = \emptyset$  — очередь пар индексов  $l$  для проведения вертикального сегмента
01: для каждого слоя  $l$  от слоя  $l_{min}(l_p, l_n)$ 
02: для региона пересечения, объединения и дополнения сегментов  $p$  и  $n$ 
03: определить индексы столбцов  $i$  для вертикального сегмента цепи  $N$  на слое  $l$ ,
учтя таблицы  $H_p$  и  $H_n$ , сегменты  $S_p$  и порты  $P_i$ 
04: отсортировать индексы  $i$  в порядке удаленности от центров сегментов  $p$  и  $n$ 
05: добавить пары  $\{(l, i) | i \in I\}$  в конец очереди  $Q$ 
06:  $r_p$  — регион, соответствующий сегменту  $p$ 
07:  $r_n$  — регион, соответствующий сегменту  $n$ 
08: для каждой пары  $(l, i) \in Q$ 
09:  $p' = (r_p \cup (r_p, l, i), l_p)$ 
10: оценить ресурс  $R_{p'}$ , требуемый для сегмента  $p'$ 
11:  $n' = (r_n \cup (r_n, l, i), l_n)$ 
12: оценить ресурс  $R_{n'}$ , требуемый для сегмента  $n'$ 
13:  $pn = ((r_p, l, i), (r_n, l, i), i)$ 
14: оценить ресурс  $R_{pn}$ , требуемый для сегмента  $pn$ 
15: если в таблицах  $H_p$  и  $H_n$  имеется свободный ресурс  $R_{p'}$ ,  $R_{n'}$  и  $R_{pn}$ 
16: зарезервировать ресурс  $R_{pn}$  в таблице  $H_p$ 
17:  $p = p'$ 
18: зарезервировать ресурс  $R_{n'}$  в таблице  $H_n$ 
19:  $n = n'$ 
20: зарезервировать ресурс  $R_{pn}$  в таблицах  $H_p$  и  $H_n$ 
21: завершить алгоритм
22:  $pn = (\emptyset, l_{undefined})$  — сегмент не удалось спланировать
    
```

Рис. 15. Алгоритм создания вертикального сегмента между диффузионными регионами

```

Алгоритм СпланироватьВертикальныеСегментыPNЦепи
Вход:  $H_p$  — символьная таблица  $p$ -региона,  $H_n$  — символьная таблица  $N$ -региона,  $N$  —
цепь,  $S_p$  — упорядоченное множество затворных сегментов,  $P_i$  — множество входных
портов,  $S_p^p$  — горизонтальные  $P$  сегменты и  $S_n^n$  — горизонтальные  $N$  сегменты
Выход:  $S_p^{pn}$  — множество вертикальных сегментов цепи  $N$ 
00:  $S_p^{pn} = \emptyset$ 
01: установить итератор  $i$  на первый сегмент из списка  $S_p^p$ 
02: установить итератор  $m$  на первый сегмент из списка  $S_n^n$ 
03:  $S = \emptyset$  — множество вертикальных сегментов, подключенных к сегменту итератора  $i$ 
04: пока оба итератора  $m$  и  $i$  находятся в границах списков
05: выбрать из итераторов  $m$  и  $i$   $P$  сегмент  $p$ 
06: выбрать из итераторов  $m$  и  $i$   $N$  сегмент  $n$ 
07:  $pn =$  СпланироватьВертикальныйСегментМеждуДиффузионными
Регионами( $H_p, H_n, N, S_p, P_i, p, n$ )
08: если между сегментами  $p$  и  $n$  создан сегмент  $pn$ 
09:  $S = S \cup pn$ 
11: переместить итератор  $m$  на следующий сегмент
12: если итератор  $m$  оказался вне границ списка
13: если  $i$  указывает не на последний сегмент в списке
14:  $i' = i$ 
15: установить итератор  $i$  на последний сегмент из списка итератора  $m$ 
16:  $m = i' + 1$ 
17:  $S_p^{pn} = S_p^{pn} \cup S$ 
18:  $S = \emptyset$ 
19: иначе если  $S \neq \emptyset$ 
20: если  $i$  установлен на последний сегмент в списке
21:  $S_p^{pn} = \emptyset$ 
22: завершить алгоритм
23: иначе
24:  $i' = i$ 
25:  $i = m - 1$ 
26:  $m = i' + 1$ 
27:  $S_p^{pn} = S_p^{pn} \cup S$ 
28:  $S = \emptyset$ 
29: иначе
30:  $S_p^{pn} = \emptyset$ 
31: завершить алгоритм
    
```

Рис. 16. Алгоритм создания вертикальных сегментов  $PN$ -цепей

```

Алгоритм ПрименитьШаблонPNGЦепей
Вход:  $H_p$  — символьная таблица  $p$ -региона,  $H_n$  — символьная таблица  $N$ -региона,  $N$  —
цепь,  $S_p$  — упорядоченное множество затворных сегментов и  $P_i$  — множество входных
портов
Выход:  $S_d = (S_n^p, S_n^n, S_p^p, S_p^n)$  — упорядоченное множество горизонтальных сегментов в
 $P$  регионе,  $N$  регионе, вертикальные сегменты к затворам и вертикальные сегменты
между диффузий
00:  $S_n^p =$  СпланироватьГоризонтальныеСегментыPNGЦепи( $H_p, N$ )
01:  $S_n^n =$  СпланироватьГоризонтальныеСегментыPNGЦепи( $H_n, N$ )
02:  $(S_p^p, S_p^n) =$  СпланироватьВертикальныеСегментыPNGЦепи( $H_p, H_n, N, S_p, P_i$ )
03: если  $S_d$  имеет более одной компоненты связности
04: попытаться связать компоненты  $S_d$ 
05: если  $S_d$  имеет более одной компоненты связности
06:  $S_d = \emptyset$ 
    
```

Рис. 18. Общий алгоритм трассировки  $PNG$ -цепей

```

Алгоритм ОпределитьПараметрыГоризонтальных
СегментовPNGЦепи
Вход:  $H$  — символьная таблица региона и  $N$  — цепь
Выход:  $S_n$  — множество горизонтальных сегментов
00:  $S_n = \emptyset$ 
01:  $T \neq \emptyset$  — последовательность подряд идущих диффузионных терминалов
02:  $r$  — регион, соответствующий таблице  $H$ 
03:  $l_{diffusion}$  — слой диффузии, соответствующий региону  $r$ 
04:  $l_{m1}$  — слой первого металла
05: для каждого столбца  $c \in H$ 
06: если  $type(c) = N$ 
07: если  $type(c) = g$ 
08: если  $T \neq \emptyset$ 
09:  $S_n = S_n \cup (T, l_{m1})$ 
10: иначе если  $type(c) = s$  или  $type(c) = d$ 
11:  $T = T \cup (r, index(c), l_{diffusion})$ 
12: если  $T \neq \emptyset$ 
13:  $S_n = S_n \cup (T, l_{m1})$ 
    
```

Рис. 19. Алгоритм определения параметров горизонтальных сегментов  $PNG$ -цепей

```

Алгоритм СпланироватьГоризонтальныеСегментыPNG
Цепи
Вход:  $H$  – символьная таблица региона и  $N$  – цепь
Выход:  $S_h$  – множество горизонтальных сегментов
00:  $S_h = \emptyset$ 
01:  $S_h^i =$  ОпределитьПараметрыГоризонтальных
СегментовPNGЦепи( $H, N$ )
02: для каждого сегмента  $s^i \in S_h^i$ 
03:  $S_h^{it} =$  СпланироватьГоризонтальныеСегменты( $H, N, s^i$ )
04: если  $S_h^{it} = \emptyset$ 
05:  $S_h = \emptyset$ 
06: завершить алгоритм
07:  $S_h = S_h \cup S_h^{it}$ 

```

Рис. 20. Алгоритм создания горизонтальных сегментов PNG-цепей

```

Алгоритм СпланироватьВертикальныйСегментМежду
ДиффузионнымИЗатворнымРегионом
Вход:  $H$  – символьная таблица региона,  $N$  – цепь,  $S_g$  – упорядоченное множество
затворных сегментов,  $P_l$  – множество входных портов и  $s = (r, l)$  – горизонтальный
сегмент
Выход:  $s_v$  – вертикальный сегмент
00:  $Q = \emptyset$  – очередь пар: индекс и слой для проведения
вертикального сегмента
01:  $r$  – регион, соответствующий сегменту  $s$ 
02: для каждого слоя  $l_j$  от слоя  $l$ 
03:  $s_g^l = \{s_g \in S_g | \text{адрес}(s_g) = l_j\}$  – сегменты  $S_g$ , лежащие на слое  $l_j$ 
04: определить индексы столбцов  $l$  для вертикального сегмента цепи  $N$  на слое  $l_j$ 
между регионом  $r$  и затворной областью, учтя таблицу  $H$ , сегменты  $s_g^l$  и порты  $P_l$ 
05: отсортировать индексы  $l$  в порядке удаленности от центра сегмента  $s$ 
06: добавить пары  $\{(r, l_j) | l_j \in l\}$  в конец очереди  $Q$ 
07: для каждой пары  $(r, l_j) \in Q$ 
08:  $s' = (r \cup (r, l_j), l_j)$ 
09: оценить ресурс  $R_{s'}$ , требуемый для сегмента  $s'$ 
10:  $s_v = \{(r, l_j), (g, l_j), l_j\}$ 
11: оценить ресурс  $R_{s_v}$ , требуемый для сегмента  $s_v$ 
12: если в таблице  $H$  имеется свободный ресурс  $R_{s'}$  и  $R_{s_v}$ 
13: зарезервировать ресурс  $R_{s'}$  в таблице  $H$ 
14:  $s = s'$ 
15: зарезервировать ресурс  $R_{s_v}$  в таблице  $H$ 
16: завершить алгоритм
17:  $s_v = (\emptyset, l_{undefines})$  – сегмент не удалось спланировать

```

Рис. 21. Алгоритм создания вертикального сегмента между диффузионным и затворным регионами

строка 03) алгоритмом, используемым для создания горизонтальных сегментов PN-цепей (см. рис. 14).

На второй фазе (см. рис. 18, строка 02) создаются вертикальные сегменты. Для планирования вертикальных сегментов используются два алгоритма: алгоритм создания вертикального сегмента между диффузионными регионами (см. рис. 15), а также его модификация, предназначенная для создания вертикального сегмента между горизонтальным диффузионным сегментом и сегментами в области затворов (рис. 21). Модифицированный алгоритм имеет некоторые отличия. Затворные сегменты не



Рис. 22. Пример использования симметричного шаблона



Рис. 23. Пример использования P-асимметричного шаблона



Рис. 24. Пример дотрассировки PNG-цепи

удлиняются, поэтому на слое рассматриваются только те столбцы, в которых они лежат. В отличие от PN-цепей, применение PNG-шаблона не прерывается, если не удастся создать очередной вертикальный сегмент.

Шаблоны PNG-цепей различаются способом создания вертикальных сегментов. Симметричный шаблон предписывает соединять каждый диффузионный сегмент с одним из затворных сегментов. На рис. 22 показан пример использования шаблона для цепи, имеющей по одному горизонтальному сегменту в каждом регионе.

Асимметричные шаблоны предписывают подключать каждый сегмент первой диффузионной области к сегментам затворной области, а сегменты второй диффузионной области — к любому из сегментов первой. Шаблон, в котором за вторую область принимается P-регион, называется P-асимметричным. Второй асимметричный шаблон называется N-асимметричным. На рис. 23 приведен пример использования P-асимметричного шаблона.

В ходе третьей фазы алгоритма цепь проверяется на связность (см. рис. 18, строки 03). Для незавершенной цепи инициируется дотрассировка (см. рис. 18, строки 04). Дотрассировка состоит в выявлении групп соединенных горизонтальных сегментов (компонент связности), определении ближайших сегментов между соседними группами и их соединении вертикальным сегментом с помощью алгоритма для диффузионных регионов (см. рис. 15).

На рис. 24 приведен пример дотрассировки PNG-цепи, имеющей две компоненты связности. Ближайшие сегменты компонент соединены с помощью вертикального сегмента, который показан отрезком со стрелкой.

### 3.5. Упорядочивание горизонтальных диффузионных сегментов

При упорядочивании сегментов (см. рис. 8, строка 10) учитываются:

- расстояния между переходами из диффузии в первый металл и шинами;
- используемый вертикальный ресурс в диффузионных регионах.

Первый показатель обусловлен тем, что для слоев, обеспечивающих переходы из диффузий и из слоя затворов в слой первого металла, определены сложные условные правила, которые могут препятствовать достижению заданной высоты ячейки. Увеличение расстояний (рис. 25, см. четвертую сторону обложки) между переходами позволяет снизить число областей, в которых имеются условия для применения этих правил. Каждый диффузионный регион обрабатывается по отдельности. В результате упорядочивания определяется последовательность детальной трассировки горизонтальных сегментов, принадлежащих региону.

Вертикальные сегменты вносят ограничения на порядок трассировки горизонтальных сегментов.

Например, в  $P$ -области на рис. 25 из-за наличия вертикального сегмента цепи  $NEN$  в первом металле горизонтальный сегмент цепи  $Y$  должен быть оттрассирован раньше горизонтального сегмента цепи  $NEN$ . Для учета таких ограничений используются графы вертикальных и горизонтальных ограничений [4]. Сначала строится граф горизонтальных ограничений. На основе него и вертикальных сегментов строится граф вертикальных ограничений. Наконец, граф вертикальных ограничений проверяется на наличие циклов. В случае, когда граф оказывается циклическим, текущий порядок цепей и набор шаблонов признаются непригодными для трассировки.

С помощью графа вертикальных ограничений генерируется множество перестановок горизонтальных сегментов. В силу того, что не все сегменты связаны горизонтальными ограничениями, в полученном множестве имеются эквивалентные, в смысле детальной трассировки, перестановки. Например, сегменты  $Z$ ,  $Y$  и  $NEN$  в  $N$ -области на рис. 25 могут трассироваться в любом порядке, и результат всегда будет одним и тем же.

На следующем шаге определяются подмножества эквивалентных перестановок и оставляется по одному представителю каждого класса во множестве. Таким образом, удается сократить число элементов множества.

Далее каждая перестановка оценивается по критерию удаленности диффузионных переходов от шины. Например, все переходы из диффузий на рис. 25 расположены на минимальном расстоянии от шин. Если в  $N$ -области изменить порядок, так что сегмент  $X$  будет трассироваться после сегмента  $Y$ , то три перехода ( $Z$ ,  $X$  и  $NEN$ ) будут расположены на минимальном удалении от шины, и один переход ( $Y$ ) будет расположен на расстоянии одного соединения от шины. Перестановка оценивается следующим образом: чем больше переходов, удаленных от шины, тем оценка выше.

Наконец, перестановки, отсортированные по критерию удаленности переходов, оцениваются по используемому вертикальному ресурсу с помощью детальной трассировки. В ходе детальной трассировки, каждый сегмент максимально прижимается к шине. Первым рассчитывается ресурс перестановки, имеющей самую лучшую оценку по критерию удаленности. Расчет продолжается до тех пор, пока удастся уменьшать ресурс.

#### 4. Экспериментальные результаты

Представленный метод использовался в модуле трассировки промышленной системы синтеза стандартных библиотек NanGate Library Creator [6]. Была подготовлена тестовая библиотека из нескольких сотен ячеек следующих групп:

- группа 1 (3 % от общего числа ячеек) — элементарные ячейки типа  $INV$  и  $BUF$ ;

- группа 2 (92 %) — комбинационные ячейки типа  $NAND$ ,  $NOR$ ,  $AND$ ,  $OR$ ,  $AOI$ ,  $OAI$ ,  $AO$ ,  $OA$  и  $MUX$ ;
- группа 3 (5 %) — ячейки типа  $XOR$ ,  $XNOR$ ,  $HA$  и  $FA$ .

Каждая группа содержала варианты ячеек, рассчитанные на разное число входных сигналов (например,  $MUX2$ ,  $MUX3$  и т. д.) и несколько нагрузочных способностей (например,  $X1$ ,  $X2$ ,  $X4$  и т. д.). Для трассировки использовались два слоя металлизации в обоих направлениях. Шины земли и питания были расположены над нижней и верхней границами в первом металле.

Табл. 1 показывает процент успешно трассированных ячеек, а также процент ячеек со всеми требуемыми портами.

В табл. 2 указано среднее время трассировки ячейки для каждой из групп.

Таблица 1

Результаты трассировки

Группа	% оттрассированных	% с портами
Группа 1	100	100
Группа 2	99	99
Группа 3	65	65

Таблица 2

Время трассировки

Группа	Среднее время трассировки, мин
Группа 1	3
Группа 2	11
Группа 3	16

#### Заключение

В работе предложен метод трассировки стандартной ячейки, учитывающий приемы, используемые при ручном проектировании топологий, и доступность портов на уровне микросхемы.

Для улучшения метода ведется разработка эвристики использования двух слоев металлизации для горизонтальных диффузионных сегментов, а также эвристики быстрой обработки ячеек с большим числом  $PNG$ -цепей.

#### Список литературы

1. Guruswamy M., Maziasz R., Dulitz P., Raman S., Chiluvuri V., Fernandez A. and Jones L. CELLERITY: A Fully Automatic Layout Synthesis System for Standard Cell Libraries // Proceedings of the 34th Design Automation Conference. 1997. P. 327–332.
2. Saint C. and Saint J. IC Layout Basics, New York: McGraw-Hill, 2002. P. 120–126.
3. Lefebvre M., Marple D. and Sechen C. The Future of Custom Cell Generation in Physical Synthesis // Proceedings of the 34th Design Automation Conference. 1997. P. 446–451.
4. Kahng A., Lienig J., Markov I. and Hu J. VLSI Physical Design: From Graph Partitioning to Timing Closure, Dordrecht: Springer, 2011.
5. Борисов В. А., Кривошеин Д. Ю., Марченко А. М., Попов Е. А. и Савченко В. Ю. Подсистема планировки цепей в стандартных ячейках нанометровых СБИС // Сб. тр. Всероссийских научно-техн. конф. "Проблемы разработки перспективных микро- и нанoeлектронных систем (МЭС)", Москва, 2012.
6. Library Creator, Nangate Inc. URL: [http://www.nangate.com/?page\\_id=263](http://www.nangate.com/?page_id=263). [1/12/2013].

A. M. Marchenko<sup>1</sup>, Sc. D., Professor, E. A. Popov<sup>1</sup>, Engineer,  
V. Yu. Savchenko<sup>2</sup>, Graduate Student, e-mail: valentine.savchenko@gmail.com

<sup>1</sup> Lomonosov Moscow State University, <sup>2</sup> National Research University of Electronic Technology

## Pattern Based Standard Cell Routing

*Standard cell methodology is widely used for designing digital application specific integrated circuits (ASIC). A standard cell is a group of interconnected transistors that provides a Boolean or storage function. Physical design is an important step in standard cell based design flow, during this step geometrical representation of ASIC (layout) is obtained from geometrical representation of standard cells. In this article some issues related to standard cell layout synthesis are considered. A routing algorithm for standard cells that considers I/O ports accessibility, vertical cell porosity, blockages on higher metallization layers and layout regularity is proposed. Routing models that takes constraints of 32 nm technology node into account are proposed as well. Presented algorithm was utilized in an industrial tool for standard cell library synthesis. It showed acceptable runtime and high coverage of cells.*

**Keywords:** layout synthesis, routing, standard cells

### Reference

1. Guruswamy M., Maziasz R., Dulitz P., Raman S., Chiluvuri V., Fernandez A., Jones L. CELLERITY: A Fully Automatic Layout Synthesis System for Standard Cell Libraries. *Proceeding of the 34<sup>th</sup> Design Automation Conference*. 1997. P. 327–332.

2. Saint C. и Saint J. *IC Layout Basics*, New York: McGraw-Hill, 2002. P. 120–126.

3. Lefebvre M., Marple D. and Sechen C. The Future of Custom Cell Generation in Physical Synthesis. *Proceedings of the 34<sup>th</sup> Design Automation Conference*. 1997. P. 446–451.

4. Kahng A., Lienig J., Markov I. and Hu J. *VLSI Physical Design: From Graph Partitioning to Timing Closure*. Dordrecht: Springer, 2011.

5. Borisov V. A., Krivoshein D. Iu., Marchenko A. M., Popov E. A. i Savchenko V. Iu. Podsystema planirovki tcepei' v standartny'kh iachei'kakh nanometrovy'kh SBIS. *Sb. tr. Vserossii' skikh nauchno-tekhnicheskikh konf. "Problemy' razrabotki perspektivny'kh mikro i nnoe 'lektronny'kh sistem (ME'S)"*, Moskva, 2012.

6. Library Creator, Nangate Inc. URL: [http://www.nangate.com/?page\\_id=263](http://www.nangate.com/?page_id=263). [1/12/2013].

УДК 004.056:004.738

**В. Ю. Михайлов**, д-р техн. наук, проф., Московский авиационный институт  
(Национальный исследовательский университет), e-mail: mihvj@yandex.ru,

**В. Н. Гридин**, д-р техн. наук, проф.,

Центр информационных технологий в проектировании РАН, e-mail: info2@ditc.ras.ru,

**Р. Б. Мазепа**, канд. техн. наук, проф., Московский авиационный институт  
(Национальный исследовательский университет), e-mail: mrb402@mail.ru

## Безопасное информационное взаимодействие. Проблемы и решения

*Рассмотрены вопросы безопасности систем информационного взаимодействия. Делается попытка системного анализа основных проблем безопасности, в частности, причин и источников уязвимостей, способов проникновения в систему вредоносного программного обеспечения (ВПО) и методы борьбы с ним. Такой подход позволяет повысить эффективность решения задачи по сравнению с традиционным утилитарным стилем, использующим отдельные, слабо связанные приемы, методы и средства обеспечения информационной безопасности. В качестве главных объектов анализа предложены и использованы логические каналы, среда исполнения и сервис среды исполнения. Отдельно рассмотрены также проблемы использования существующих средств борьбы с ВПО, показана актуальность разработки комплексных средств борьбы с ВПО для использования специалистами различной квалификации, в том числе и для некоммерческого применения.*

**Ключевые слова:** информационные системы, информационное взаимодействие, логический канал, протокол, уязвимость, информационная безопасность

### Введение

Под системами информационного взаимодействия (СИВ) будем понимать любые информационные системы, выполняющие свою задачу путем формирования и передачи разнообразных сообщений служебного (управляющего, уведомляющего) и целевого (фактографического) типа.

Массовое использование информационных технологий во всех сферах деятельности порождает

необходимость создания и использования разнообразных, призванных повысить эффективность этой деятельности СИВ. Логика развития СИВ такова, что ее программные компоненты не только интенсивно заменяют и дополняют аппаратные компоненты для достижения высокой гибкости и управляемости, обеспечивающие значительное продление "жизненного цикла" подсистем, но и становятся основой СИВ как сложных систем. Отрицательным

фактором этого процесса является то, что программное обеспечение по своей "информационной" природе в значительно большей степени, чем оборудование подвержено внешним, разрушительным информационным воздействиям. Следовательно, достигаемая более высокая гибкость и управляемость СИВ распространяется и на нелегальные активные информационные воздействия (например, неправильное использование открытых протокольных конструкций), подавляющие работу СИВ или снижающие их эффективность [1].

## 1. Общая характеристика проблемы безопасности СИВ

Проектирование новых и модернизация существующих СИВ должны учитывать современную опасность информационной среды и тенденцию к постоянному нарастанию ее в ближайшей перспективе. Основными причинами столь неприятного вывода являются:

- вынужденное, в целях обеспечения совместности, применение стандартных базовых протоколов, реализующих все уровни модели OSI, но уже не отвечающих в полной мере требованиям безопасности;
- открытость логической среды Интернет, в основном обеспечивающей глобальное информационное взаимодействие;
- постоянный рост количества и функциональной сложности СИВ, использующих ограниченный общий ресурс (например, полосу частот), что порождает разнообразные явления утечки информации в физических каналах передачи данных, а также воздействия на них внешних естественных возмущений и искусственных помех;
- интенсивное развитие и применение беспроводных технологий информационного взаимодействия, использующих, в силу своей природы, открытые физические радиоканалы;
- разнообразие программно-аппаратных платформ, достигающих совместности с логической средой информационного взаимодействия путем использования множества, в основном программных, интерфейсов, являющихся потенциально дополнительными источниками угроз;
- доступность простых для освоения и применения высокоуровневых средств проектирования программного обеспечения (ПО), вовлекающих в процесс разработки анализирующих и вредоносных программ все более широкий круг программистов.

Разумеется, в распоряжении разработчиков СИВ находится множество разнообразных методов, технологий и приемов противодействия возникающим угрозам. Однако в силу разных причин, в том числе кадрового, производственного и организационного характера, они не в состоянии эффективно и своевременно парировать возникающие угрозы. Нельзя забывать, что СИВ создают и эксплуатируют для решения прикладных целевых за-

дач, а не для борьбы с угрозами. На наш взгляд, множество этих причин и порождаемых ими проблем можно разбить на следующие категории.

1. Неопределенность решаемой задачи безопасности СИВ.

В основном эта категория имеет отношение к проектированию новых и сложных, а следовательно, дорогостоящих СИВ. К проблемам данного класса относятся:

- неизвестность потенциальных угроз, а следовательно, невозможность заложить в проект изученные и апробированные эффективные механизмы противодействия;
- слабая обоснованность затрат на решение потенциальных задач безопасности.

2. Причины, порождаемые конкурентной борьбой и стремлением быстро и дешево парировать угрозы безопасности СИВ.

Эти причины в основном имеют отношение к модернизации (модификации) существующих СИВ и продлению их жизненного цикла. Вот основной перечень этих причин, включающий порожденные ими проблемы:

- массовое привлечение к решению задач безопасности специалистов недостаточно высокой квалификации и (или) планирование недопустимо низких временных границ решения задач, следствием чего является их низкое качество;
- неэффективная практика противодействия угрозам методом слабо контролируемого "латания дыр" или установки "заплаток" в существующем ПО, самих по себе являющихся потенциальным источником очередных "жучков" (*bugs*) и возможно уязвимостей;
- изобретение и применение "фирменных" решений проблемы, зачастую носящих характер рекламных заявлений и часто приводящих к несовместимости информационных протоколов.

3. Причины системного, проектного характера, связанные с недостаточным вниманием к точности и детальности структурной проработки проекта, включающей:

- адекватное описание предметной области и ее границ (формулировку целей и задач СИВ);
- декомпозицию системы и всех интерфейсов на функциональном, логическом и других уровнях представления;
- обоснование выбора принципа информационного взаимодействия компонентов системы (функционально связанное, синхронное или асинхронное взаимодействие, взаимодействие посредством послылки сообщений (*messaging*) и др.);
- резервирование и другие способы повышения надежности, помехо- и отказоустойчивости, скорости восстановления работоспособности системы;
- способы управления режимами функционирования подсистем, их состоянием и поведением.

Главной проблемой здесь является недостаточное качество проекта со слабой адаптацией к изменяющимся условиям функционирования и потенциальным угрозам.

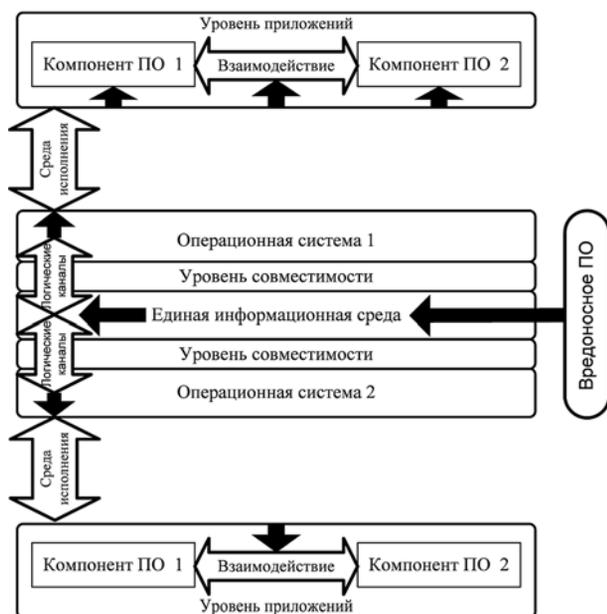
4. Причины технологического характера, связанные с выбором методов и вариантов реализации (воплощения) компонентов СИВ. К ним могут быть отнесены:

- необоснованно малое внимание, уделяемое корректности и безопасности программного кода ПО;
- неполная совместимость реализаций протоколов информационного взаимодействия на разных платформах;
- свойство новых разработок (в основном ПО) к заимствованию старых и порождению новых ошибок (*bugs*).

Результатом является неэффективный, опасный и плохо поддающийся модификации программный код ПО, одной из главных проблем которого является слабая защита от внешних активных воздействий.

## 2. Потенциальные уязвимости СИВ

Очевидно, что пути и способы проникновения исследующего (анализирующего) и вредоносного ПО (ВПО) в систему связаны с потенциальными уязвимостями ее компонентов. На рисунке схематично представлены направления и цели атакующих воздействий на систему и ее главные компоненты. Любые атакующие воздействия по существу эксплуатируют главную уязвимость всех информационных систем — единство информационной среды и уровней совместимости ее с операционными системами и их интерфейсными компонентами, которые обеспечиваются, прежде всего, протоколами, реализующими логические каналы информационного взаимодействия. На рисунке эти каналы отображены стрелкой, логически соединяющей два, возможно разнородных, элемента распределенной информационной системы.



Среда и разрушительные воздействия

### 2.1. Уязвимости логических каналов

Руководствуясь данным в работе [2] определением логических каналов и соответствующих им протоколов, рассмотрим основные источники их уязвимостей. Все многообразие протоколов распадается на два подмножества: открытые (стандартные) и закрытые (коммерческие) протоколы. Каждое из них имеет свои достоинства и недостатки.

К достоинствам открытых протоколов относятся:

- гарантия совместимости информационных процессов в оговоренных протоколами границах;
- постоянство правил информационного взаимодействия, гарантирующее массовое развитие надежных элементов информационных технологий;
- известность правил всем легальным участникам информационного процесса.

Известность правил информационного взаимодействия имеет и отрицательную сторону, открывая возможность атакующим осуществлять целенаправленные разрушительные действия. Другим важным недостатком открытых протоколов является их консервативность, вследствие чего они не отвечают растущему уровню "агрессивности" информационной среды.

Достоинства закрытых протоколов на первый взгляд являются зеркальным отображением недостатков открытых протоколов. Например, неизвестность правил, казалось бы, должна обеспечить относительную безопасность применения закрытых протоколов. Однако не следует забывать, что атакующие в силу ряда известных причин всегда имеют преимущество над защищаемыми и рано или поздно вскроют первоначально неизвестный им протокол и начнут использовать его уязвимости до того, как разработчик протокола оснастит его пользователей защитной "заплаткой". Следовательно, рассмотренная особенность закрытых протоколов вряд ли может рассматриваться как их достоинство. Гибкость правил закрытых протоколов потенциально обеспечивает возможность их адаптации к растущему уровню "агрессивности" информационной среды. Однако эту возможность следует еще реализовать — эта особенность закрытых протоколов может стать достоинством лишь в случае надежного и постоянного их сопровождения разработчиком. Практика показывает, что бесчисленные пакеты обновления (*service packs*), "заплатки" (*patches*), а также другие не слишком вразумительные программные средства (*updates*, *bugfixes* и т. д.) через некоторое время все же не справляются с нагрузкой и их заменяют новой разработкой. Иными словами, рассмотренное достоинство лишь временное.

Среди явных недостатков закрытых протоколов особо выделяются следующие:

- сильная зависимость разработчиков ПО от неизвестных им модификаций протоколов;
- неизвестный, неконтролируемый и часто просто декларируемый уровень защищенности пользователей.

В наибольшей степени и раньше всех остальных рассмотренные логические каналы и соответствующие им протоколы используют следующие общераспространенные сетевые Internet-приложения:

- Web-браузеры и Web-серверы;
- Ftp-клиенты и серверы;
- почтовые клиенты и серверы;
- клиенты и серверы систем мгновенного обмена сообщениями (IM);
- клиенты пиринговых сетей (P2P);
- клиенты и серверы баз данных.

Эти приложения, являясь, по сути, "точками входа" в удаленную систему, первыми принимают на себя атакующие воздействия. Однако эти приложения являются высокоуровневыми и их работа базируется на сетевых службах и протоколах. Наиболее опасными из них являются Net Logon, DNS, DHCP, WINS, Telnet, RPC (NetDDE), Proxy-серверы, RDS, Terminal Services, Remote Registry Service, NetBios, TCP/IP и др.

## **2.2. Уязвимости среды исполнения**

Обычно под средой исполнения понимается операционная система и предоставляемый ею программным компонентам так называемый системный сервис. На рисунке среда исполнения показана в виде стрелки, соединяющей уровень операционной системы с уровнем приложений. Уязвимости среды исполнения порождаются двумя ее особенностями: структурной и функциональной.

Структурная особенность среды исполнения состоит в широком диапазоне уровней модели OSI, на которых работают компоненты операционной системы: от MAC-уровня до прикладного. В своей работе эти компоненты создают множество параллельных программных потоков, использующих один и тот же ресурс памяти, что при плохой синхронизации (ошибки случаются всегда) создает опасность проникновения на уровень приложений вирусов и других зловредных программ. Следствия такого проникновения следующие:

- изменение режимов работы компонентов ОС, вплоть до захвата вирусами (зловредными программами) контроля над ней;
- неконтролируемые области памяти могут быть проанализированы "шпионами" на предмет вскрытия программного кода и методов шифрования;
- конфликты между потоками, которые создают условия для сбоев и отказов программных компонентов.

Функциональные особенности среды исполнения выражаются в виде широкого набора различного уровня услуг, предоставляемых ОС программам и реализованных на различных языках программирования с использованием разнообразных технологий. Так как существующие ОС не ограничивают использование как различных типов сервисных компонентов ОС, так и технологий разработки ПО, то появляется опасность несовместимости при взаимодействии ОС с прикладными программами. Дос-

тижение совместимости требует дополнительных преобразований, снижающих быстродействие СИБ. Разнообразие высокоуровневых технологий, используемых ПО, приводит к дублированию операций на низком уровне и, следовательно, к зависимости ошибок при программировании. Основой компонентного проектирования, как известно, является богатый выбор готовых компонентов и массовое их использование в разработках ПО. Выбор готовых компонентов с учетом разнообразия использованных в них языков и технологий программирования не всегда отвечает адекватному их взаимодействию друг с другом и с другим ПО.

Основными компонентами, составляющими среду исполнения, являются следующие приложения и объекты локального действия:

- интерпретирующие приложения и их объекты (текстовые и табличные процессоры), включающие клиентский код на сценарных языках Java, JavaScript, VB Script, PHP и др.;
- командные и служебные файлы операционных систем: com-, bat-, inf-файлы, WSH-приложения;
- Autorun и ключи реестра: HKLM\SYSTEM\CurrentControlSet\Services\;
- файловая система;
- драйверы устройств;
- компоненты уровня ядра ОС.

## **2.3. Уязвимости сервиса среды исполнения**

Под сервисом среды исполнения понимается программный сервис, предоставляемый операционной системой и другими прикладными службами программным компонентам для реализации межпрограммного (межкомпонентного) взаимодействия. На рисунке сервис среды исполнения показан в виде стрелки, соединяющей компоненты программного обеспечения. Особенностью данного сервиса является использование для этой цели широкого набора высокоуровневых технологий, что в совокупности с большим разнообразием технологий проектирования ПО способствует "размножению" конфликтов как между различными компонентами ПО, так и между компонентами ПО, с одной стороны, и системным сервисом, с другой стороны. Главным отличием уязвимостей данного типа от рассмотренных выше является появление нового их источника: прикладного сервиса, расширяющего системный сервис.

Очевидно, что на этом уровне рассмотрения используются наиболее разнообразные компоненты ПО и протекают наименее контролируемые процессы их взаимодействия. Поэтому несмотря на то что данный уровень изолирован от опасной информационной среды, проникшие сюда вредоносные программы имеют в своем распоряжении широкие возможности по эксплуатации появившихся уязвимостей. Наконец, прикладное ПО проектируется и отлаживается в условиях, отличающихся от рабочих, — в операционной среде, где чаще всего нет условий для возникновения подобных реальных уязвимостей.

Из изложенного вытекают следующие выводы:

- достижение совместимости требует от программистов хорошего знания структуры ОС, так как прикладное ПО воздействует на системный сервис;
- разнообразие высокоуровневых технологий приводит к дублированию операций на низком уровне и, следовательно, к зависимости ошибок при программировании;
- несоответствие версий системного сервиса, используемого при разработке и эксплуатации ПО, приводит к сбоям и отказам, являясь дополнительным источником уязвимостей СИБ.

### 3. Условия проникновения, методы обнаружения и борьбы с ВПО

#### 3.1. Условия проникновения ВПО

Проведенный выше анализ источников уязвимостей позволяет сформулировать главные условия проникновения и активизации ВПО в следующем виде:

- наличие сетевых клиентов и служб и работа в сети;
- ошибки мандатного доступа (работа под административным аккаунтом);
- ошибки дискреционного доступа (доступ к объектам);
- количество, структура активных процессов и их состояние;
- использование режима автообновления ПО;
- неправильные настройки межсетевого экрана (*Firewall*), антивирусного и диагностического ПО;
- небезопасные настройки приложений и служб;
- наличие и настройка отладчиков;
- ошибки проектирования (лишние связи между компонентами ПО и операционной средой, а также между отдельными компонентами ПО);
- ошибки программирования (прежде всего ошибки работы с памятью).

#### 3.2. Методы обнаружения ВПО

Все методы обнаружения ВПО разделяют на три типа в зависимости от используемой ими технологии обнаружения: обнаружение на этапе загрузки, обнаружение в процессе работы ОС и обнаружение при работе ОС под управлением гипервизора. Необходимость использования методов обнаружения первого типа обусловлена способностью ВПО (в частности *Rootkit* [3]) "прятаться" после загрузки ОС так, что методы второго типа просто не в состоянии обнаружить. Ясно, что такие методы являются самыми сложными и сами представляют собой некоторое "супер ВПО", способное "обогнать" ВПО в процессе загрузки ОС и "подсмотреть" таким образом процесс загрузки остальных компонентов ОС.

Одним из немногих эффективных средств обнаружения ВПО первого типа является пакет *RegRun* фирмы *Greatis*, который с помощью специального драйвера *Partizan* решает следующую задачу: имея полный доступ к объектам ОС, выявляет опасные файлы и соответствующие им ключи реестра на ос-

новании базы данных известного ВПО (*Greatis Application Database*).

К основным методам обнаружения ВПО второго типа относятся:

- поиск программ, применяющих механизмы *Rootkit*;
- анализ запущенных процессов;
- анализ процесса и режимов автозапуска программ;
- поиск опасных и новых файлов в системных каталогах на диске (каталоги *Windows*, *System32*, *Drivers*, *Security Storage* и др.);
- анализ файловых менеджеров и модулей расширения *Internet Explorer*;
- анализ файла *hosts*.

Ярким примером реализации методов третьего типа является программное средство нового класса *Hypersight Rootkit Detector*. В настоящий момент это единственная программа, способная контролировать целостность ядра ОС, при этом она работает "прозрачно" для ОС и всех программ, запущенных в ней.

#### 3.3. Проблемы использования существующих средств борьбы с ВПО

Перечислим главные проблемы рассмотренных и других программных средств борьбы с ВПО.

1. Сложность обоснования выбора и использования конкретного инструментария борьбы с ВПО.

Пользователь должен обладать высокой квалификацией в области информационной безопасности и иметь четкое представление о назначении и характеристиках объектов защиты, назначении и характеристиках средств защиты.

2. Сложность интерпретации результатов работы средств борьбы с ВПО и последующих действий пользователя.

Ряд средств борьбы в итоге генерирует отчет о предположительно найденном ВПО, а выбор дальнейших действий возлагается на пользователя. В частности, пользователь сам решает, что делать на этапе следующей загрузки системы. В случае "нейтрализации" ВПО пользователь может только довериться результатам работы используемых средств борьбы с ВПО.

3. Работа программных средств обнаружения ВПО по принципу "черного ящика".

Скрытость методов работы конкретного средства борьбы с ВПО, технологий разработки и самого программного кода препятствует их оперативному внедрению.

4. Большинство средств борьбы с ВПО являются коммерческими.

Следует, однако, отметить, что ряд производителей (например, упомянутые выше *Greatis*, Лаборатория Касперского) сегодня предоставляет на рынок бесплатные версии своих программных средств, несколько более "легкие" по сравнению с их полноценными коммерческими аналогами.

#### Заключение

Одним из главных результатов проведенного анализа является вывод о целесообразности разра-

ботки единой методики и соответствующего набора инструментальных средств борьбы с ВПО. Главные составляющие обоснования достижимости такой цели выглядят так.

#### 1. Открытость ОС.

Большинство ОС — коммерческие. Однако архитектура существующих операционных систем (Windows, Unix/Linux, Mac OS), включая мобильные платформы, известна.

#### 2. Проблема исследования "траекторий" внедрения ВПО.

Нахождение всех возможных точек проникновения ВПО в систему (см. рисунок) вполне реально. Однако требуется обоснование отсутствия других (необнаруженных) точек входа.

#### 3. Проблема исследования поведения ВПО.

Базой продуктивной работы любого ВПО является множество функций, библиотек и других объектов среды исполнения. Определение этого множества, факта и последовательности (порядка) вызова функций позволит достаточно точно выделить ВПО на фоне легальных процессов.

#### Список литературы

1. Хогланд Г., Мак-Гроу Г. Взлом программного обеспечения: анализ и использование кода / Пер. с англ. М.: Вильямс, 2005.
2. Проблемы безопасного информационного взаимодействия в распределенной среде / Под ред. Р. Б. Мазепы, В. Ю. Михайлова. М.: МАИ-ПРИНТ, 2009.
3. Зайцев О. В. Rootkits, Spyware/Adware, Keyloggers & Backdoors: обнаружение и защита. СПб.: БХВ-Петербург, 2006.

V. Yu. Mikhaylov, Moscow Aviation Institute (National Research University), Moscow,  
V. N. Gridin, Center of Information Technologies in Design, Russian Academy of Sciences,  
Odintsovo, Moscow region, e-mail: info2@ditc.ras.ru,  
R. B. Mazepa, Moscow Aviation Institute (National Research University), Moscow

## Secure Information Interaction. Problems and Solutions

*Article is devoted to information exchange systems security. Attempt of the system analysis of the main problems of security, in particular, the reasons and sources of vulnerabilities, ways of penetration into system of malicious software and methods of fight against it becomes. Such approach allows to increase efficiency of the task solution in comparison with the traditional utilitarian style using separate, poorly connected receptions, methods and means of ensuring of information security. As the main objects of the analysis proposed and used logical channels, runtime (execution) environment and service of runtime (execution) environment. Special consideration is also the problem of using existing tools to fight against malicious software. Shown actuality of integrated tools development to fight against malicious software for the use of different qualifications, including for non-commercial use.*

**Keywords:** information systems, communication, logical channel, protocol, vulnerabilities, information security

#### References

1. Hogland G., Mak-Grou G. Vzlom programmnogo obespecheniya: analiz i ispol'zovanie koda / Per. s angl. M.: Vil'jams, 2005.

2. Problemy bezopasnogo informacionnogo vzaimodejstviya v raspredelennoj srede / Ed. R. B. Mazepy, V. Ju. Mihajlova. M.: MAI-PRINT, 2009.
3. Zajcev O. V. Rootkits, Spyware/Adware, Keyloggers & Backdoors: obnaruzhenie i zashhita. SPb.: BHV-Peterburg, 2006.

УДК 004.02.021

А. С. Добрынин, зав. лаб., ст. преподаватель, e-mail: serpentfly@mail.ru,

Р. С. Койнов, зав. сектором, ст. преподаватель,

Сибирский государственный индустриальный университет (СибГИУ), e-mail: koynov\_rs@mail.ru

## Об одном алгоритме построения расписаний в условиях временных ограничений

*Задачи построения расписаний (JSSP) в различных сферах человеческой деятельности имеют важнейшее теоретическое и прикладное значение. Во многих из них присутствуют специфические временные ограничения, описывающие допустимые для планирования отрезки времени и периоды простоев на интервале планирования. Планирование — важный элемент управления. Его можно формализовать, используя такой рабочий инструмент, как план-график производства тех или иных работ. План-график представляет собой определенную последовательность этапов и сроков выполнения работы. Составляется он до начала работ. План-график, его отдельные этапы могут корректироваться в ходе ведения работ, но конечные сроки их выполнения должны выдерживаться с максимальной точностью. Описан алгоритм распределения работ в условиях временных ограничений для задач построения производственных и учебных расписаний (графиков), а также сервисной деятельности.*

**Ключевые слова:** распределение работ, расписание, ИТ-сервис, планирование работ, временные ограничения, сервисное управление

## Введение

В статье описан алгоритм построения расписаний планирования работ, операций в условиях временных ограничений. Данная задача актуальна для промышленных предприятий, организаций различных сфер деятельности, в которых присутствует временное планирование.

Рассматриваемая задача базируется на классической задаче назначения работ (сетевое планирование — JSSP) для непрерывного времени. Постановка рассматриваемой в работе задачи опирается на решение, полученное для непрерывного времени и набора временных ограничений.

Поскольку большинство организаций осуществляет деятельность, опираясь на рабочие графики, алгоритм, рассматриваемый в данной работе, позволяет получить более точные решения для реальных производственных систем, в которых присутствуют ограничения.

### 1. Элементы математической модели

Базовая задача построения производственных расписаний [1] для непрерывного времени формулируется как задача на графах, в которой узлы представляют собой события, дуги — отдельные процессы или работы. С каждой дугой ассоциирован двухкомпонентный вес, представленный вещественным числом и временной разницей с возможностью их взаимного отождествления. Этапы решения базовой задачи [1] реализованы в рамках модельно-алгоритмического комплекса (МАК) [2] и дают неплохие результаты на практике.

Особый интерес представляет задача, в которой необходимо учитывать ограничения, связанные с невозможностью распределить работы в определенный интервал времени. Сложность заключается в вариативном характере таких ограничений, которые могут изменяться в различных постановках. Рассмотрим элементы математической модели для достаточно общего случая, предполагая, что на периодических интервалах времени  $t + \Delta t$  структура ограничений одинакова.

Одним из элементов математической модели, используемой для построения расписаний в ограничениях, является **вектор кортежей работ**  $\bar{W}$ , полученный в ходе решения задачи [1], где каждая отдельная запись представляет собой параметры отдельной работы, такие как: идентификатор работы ( $ID$ ); дата начала ( $beginDate$ ); дата раннего окончания ( $earlyEndDate$ ); дата позднего окончания ( $lastEndDate$ ); компонент временного смещения ( $offsetDate$ ):

$$w_i = \{ID_i, beginDate_i, earlyEndDate_i, lastEndDate_i, offsetDate_i\}. \quad (1)$$

С точки зрения процедуры составления расписаний отдельный кортеж (запись) представляет набор связанных данных по отношению к некоторому

идентификатору работы, часть из которых используется алгоритмом построения расписаний.

Также важнейшим элементом математической модели является логическая **матрица работ и простоев**  $timeMap[d \in Days, h \in Hours]$ , которая описывает временную сетку интервалов проведения работ, такую что

$$timeMap[d, h] = \begin{cases} 1, & \text{допустимо размещение} \\ & \text{элемента работы,} \\ 0, & \text{простой, размещение} \\ & \text{не допускается.} \end{cases} \quad (2)$$

Для случаев описания детальных временных компонент матрица работ и простоев может быть трансформирована в **кортеж работ и простоев** (использование более двух временных компонент). В общем случае структура и вид матрицы или кортежа зависят от размерности времени, требуемой точности задания отрезков времени и динамики процессов. В задачах построения производственных расписаний целесообразно использовать "сжатую" интерпретацию, когда известно, что производственные процессы четко привязаны к конкретным дням недели:

$$M[DayOfWeek[d], h] = \begin{cases} 1, & \text{допустимо размещение} \\ & \text{элемента работы,} \\ 0, & \text{простой, размещение} \\ & \text{не допускается.} \end{cases} \quad (3)$$

Введем понятие левого и правого **временного сдвига**, которое будет означать единичное приращение минимальной компоненты кортежа в сторону уменьшения или увеличения времени. Таким образом, для кортежа  $K[d \in Day, h \in Hour, m \in Minute]$  сдвигом будет кортеж  $K[d, h, m \pm 1]$ . Рассматриваемый здесь алгоритм назначения работ (*time — labeling*) использует модель ограничений, представленную выражением (2).

### 2. Ключевые идеи алгоритма

Для упрощения понимания сути работы алгоритма в целом, выделим несколько ключевых идей.

1. **Двухкомпонентный, двунаправленный временной итерационный механизм.**

Итератор *workIterator* сдвигает временной кортеж в прямом направлении, итератор *durationIterator* сдвигает временной кортеж в обратном направлении (рис. 1).

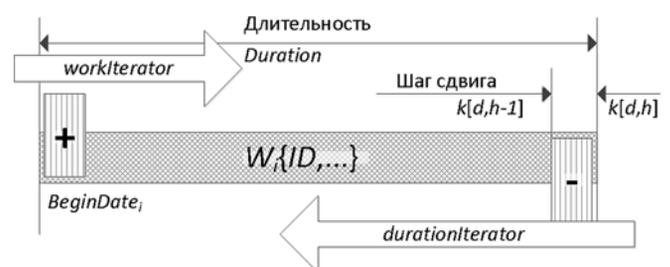


Рис. 1. Двунаправленная итерация по времени

2. Механизм сдвига, с учетом смещения, непосредственно влияющий на окрестность работ  $\bar{W}_N \notin \bar{W}$ , расположенных справа относительно текущей работы  $w_i$ .

Возможность поиска работ, расположенных в окрестности текущей работы  $w_i$  справа или слева, достигается за счет реализации в информационной модели дуги графа ссылки на стартовый и конечный узел:

```

/*Интерфейс для дуги графа IEdge<T>*/
public interface IEdge<T> : IComparable<IEdge<T>>
{
    ///Временной интервал дуги (выраженный через
    временную разницу)
    TimeSpan Duration { get; set; }
    /// Стартовый узел для дуги, как INode<T>
    INode<T> start_node { get; }
    /// Конечный узел дуги, как INode<T>
    INode<T> end_node { get; }

    string start_nodeid { get; }
    ///Стартовый идентификатор узла
    string end_nodeid { get; }
    ///Конечный идентификатор узла
    string name { get; set; } ///Наименование дуги
    string manager { get; set; }
    double Weigth { get; set; } ///Вес дуги
}

```

Информационная модель дуги графа содержит ссылки на стартовый и конечный узлы графа, реализующие поведенческий механизм  $INode<T>$ , также имеется возможность работы по уникальным строковым идентификаторам узлов. Таким образом, итерационный процесс по отдельной дуге графа воздействует на окрестность дуг, расположенных после текущей дуги (рис. 2).

### 3. Оригинальный механизм временной разметки (time-labeling) с использованием списка запретов.

Суть итерационного механизма заключается в следующем: если на очередном  $i$ -м шаге итерации элемент кортежа  $k_i[d, h]$  для работы  $w_i$  не может быть распределен, происходит сдвиг всех временных характеристик работ в окрестности справа от  $w_i$  с учетом смещения для следующей работы на интервал времени  $k_i[d, h + 1]$ , если он отсутствует в списке запретов. В противном случае длительность текущей работы уменьшается на интервал времени  $k_i[d, h - 1]$ , при этом сдвига временных характеристик работ в окрестности справа  $w_i$  не происходит. Так как имеется  $n$  работ в окрестности слева от текущей работы  $w_i$  и итерирование каждой из них приводит к сдвигам  $w_i$ , целесообразно использовать список запретов

$tabooList$ , каждый элемент которого представляет собой кортеж  $k_{taboo}[id, k_i[d, h]]$ . Список запретов создается отдельно для каждой работы  $w_i$  и содержит даты, которые уже использовались ранее для сдвига работы  $w_i$ .

### 3. Содержательное описание алгоритма

Опираясь на описанные выше идеи, сформулируем алгоритм, пригодный для построения расписаний в сколь угодно сложных практических случаях временных ограничений, при условии их однородности.

**Шаг 1.** Сортировка вектора  $w_i$  по возрастанию даты начала работы  $beginDate$ .

**Шаг 2.** Определение даты начала проекта  $prjDate$  как  $w_0\{\dots, beginDate, \dots\}$ .

**Шаг 3.** Двухнаправленная итерационная процедура по каждой работе  $w_i \in \bar{W}$ ,  $i = 0 \dots (N - 1)$ , выпол-



Рис. 2. Подмножества сдвига при итерационном движении

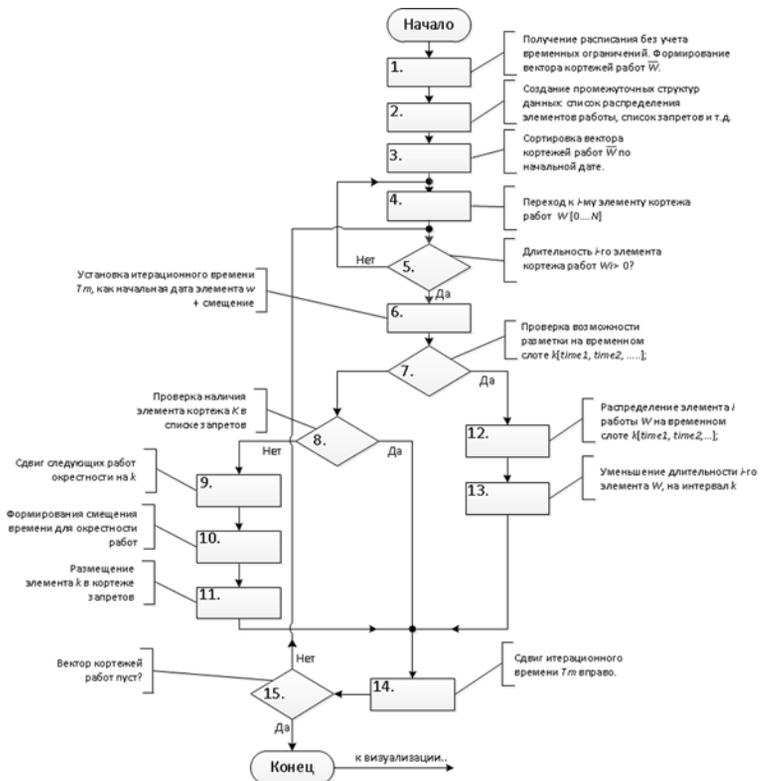


Рис. 3. Блок-схема алгоритма разметки (labeling) работ

нием действия по формированию кортежей ее размещения во времени. Формирование вектора кортежей  $LBL$ , каждый элемент которого содержит идентификатор работы и дату начала разметки для временного сдвига.

*Шаг 4.* Визуализация вектора кортежей  $LBL$  с использованием механизма рендеринга WPF.

#### 4. Блок-схема алгоритма

Представленная в данном разделе блок-схема алгоритма построена с опорой на процесс отладки работающей реализации на языке программирования C# в рамках модельно-алгоритмического комплекса (МАК) построения расписаний [2]. Алгоритм был опробован на 10 тестовых структурах графов при произвольной генерации значений для матрицы временных ограничений. Блок-схема представлена на рис. 3.

#### Заключение

Рассмотренный в работе алгоритм реализован в составе модельно-алгоритмического комплекса МАК [2], опробован на множестве модельных структур графов работ (более 10) при произвольных способах задания кортежей временных ограничений.

#### Список литературы

1. Добрынин А. С., Кулаков С. М., Зимин В. В. Формализация задачи составления расписаний для стадии внедрения ИТ-сервиса // Научное обозрение: теория и практика. 2013. № 2. С. 47–52, 110.
2. Добрынин А. С., Кулаков С. М., Зимин В. В., Бондарь Н. Ф. О формировании комплекса инструментальных средств ИТ-провайдера для построения расписаний процесса внедрения сервиса // Научное обозрение. 2013. № 8. С. 93–101.
3. Мартин Р. С., Мартин М. Принципы, паттерны и методики быстрой разработки приложений на языке программирования C#. М.: Символ-Плюс, 2013. 786 с.
4. OGC-ITIL V3-2 Service Transition, TSO.— 2007.

A. S. Dobrynin, Head of Laboratory, Senior Teacher, e-mail: serpentfly@mail.ru,

R. S. Koynov, Head of Sector, Senior Teacher, e-mail: koynov\_rs@mail.ru,  
Siberian State Industrial University (SibSIU)

## About an Algorithm for Construction of a Time-Schedule in Conditions of Timing Constraints

*Tasks of time-schedule construction (JSSP) in various fields of human activities have an important theoretical and practical significance. The main feature of these tasks is a timing requirement, describing allowed planning time periods and periods of downtime. This article describes implementation variations of the work scheduling algorithm under timing requirements for the tasks of industrial time — schedules construction, and service activities.*

**Keywords:** labeling, timetable, IT-service, job shop schedule, time constraints, service management

#### References

1. Dobrynin A. S., Kulakov S. M., Zimin V. V. Formalizacija zadachi sostavlenija raspisanij dlja stadii vnedrenija IT-servisa. *Science Review: Theory and practice*. 2013. N. 2. P. 47–52, 110.
2. Dobrynin A. S., Kulakov S. M., Zimin V. V., Bondar N. F. O formirovanii kompleksa instrumental'nyh sredstv IT-provajdera dlja postroenija raspisanij processa vnedrenija servisa. *Science Review*. 2013. N. 8. P. 93–101.
3. Martin R. S., Martin M. Principy, patterny i metodiki bystroj razrabotki prilozhenij na jazyke programmirovanija C#. М.: Simvol-Pljus, 2013. 786 p.
4. OGC-ITIL V3-2 Service Transition TSO. 2007.

#### Адрес редакции:

107076, Москва, Стромьинский 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 07.08.2014. Подписано в печать 18.09.2014. Формат 60×88 1/8. Бумага офсетная.

Усл. печ. л. 8,86. Заказ IT1014. Цена договорная.

Журнал зарегистрирован в Министерстве Российской Федерации по делам печати, телерадиовещания и средств массовых коммуникаций.

Свидетельство о регистрации ПИ № 77-15565 от 02 июня 2003 г.

Оригинал-макет ООО "Авансед солюшнз". Отпечатано в ООО "Авансед солюшнз".

119071, г. Москва, Ленинский пр-т, д. 19, стр. 1.