

ISSN 1816-0301

# ИНФОРМАТИКА

3(35)

ИЮЛЬ-СЕНТЯБРЬ

2012

## **Редакционная коллегия:**

*Главный редактор*

**А.В. Тузиков**

*Заместитель главного редактора*

**М.Я. Ковалев**

*Члены редколлегии*

С.В. Абламейко, В.В. Анищенко, П.Н. Бибило, М.Н. Бобов,  
А.Н. Дудин, А.Д. Закревский, С.Я. Килин, В.В. Краснопрошин,  
С.П. Кундас, Н.А. Лиходед, П.П. Матус, С.В. Медведев, А.А. Петровский,  
Ю.Н. Сотсков, Ю.С. Харин, А.Ф. Чернявский, В.Н. Ярмолик  
Н.А. Рудая (*заведующая редакцией*)

---

*Адрес редакции:*

220012, Минск,  
ул. Сурганова, 6, к. 305  
тел. (017) 284-26-22  
e-mail: [rio@newman.bas-net.by](mailto:rio@newman.bas-net.by)  
<http://uiip.bas-net.by>

ОБЪЕДИНЕННЫЙ ИНСТИТУТ ПРОБЛЕМ ИНФОРМАТИКИ  
НАЦИОНАЛЬНОЙ АКАДЕМИИ НАУК БЕЛАРУСИ

---

---

# ИНФОРМАТИКА

---

---

ЕЖЕКВАРТАЛЬНЫЙ НАУЧНЫЙ ЖУРНАЛ

*Издается с января 2004 г.*

---

---

№ 3(35) • июль-сентябрь 2012

## СОДЕРЖАНИЕ

### РАСПОЗНАВАНИЕ ОБРАЗОВ, ОБРАБОТКА ИЗОБРАЖЕНИЙ, СИГНАЛОВ И РЕЧИ

<b>Артемьев В.М., Наумов А.О., Кохан Л.Л.</b> Синтез фильтров подавления фона на цифровых изображениях в оптико-электронных системах.....	5
<b>Прокопович Г.А.</b> Модель нейросетевой ассоциативной памяти для управления манипулятором.....	16
<b>Пекарь Д.В., Тихоненко С.Г., Чернявский А.Ф., Садов В.С.</b> Алгоритм оптимизации входных признаков и параметров классификатора эмоционального состояния человека по его речи.....	26
<b>Шут О.В.</b> Метод решения задач распознавания в булевом пространстве на основе логической и прецедентной моделей .....	35

### ПРИКЛАДНЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ

<b>Лапицкий В.А., Лаханько Л.Н., Батраков А.Г., Кириченко В.В.</b> Распределенная телемедицинская система по цифровой маммографии .....	51
<b>Анищенко В.В., Кульбак Л.И.</b> Оценка ущерба работоспособности информационной системы .....	61
<b>Фролов Г.В.</b> Системный подход к защите информационных технологий .....	70

## МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ

<b>Ерофеев В.Т., Шушкевич Г.Ч., Грабчиков С.С., Бондаренко В.Ф.</b> Модель экранирования постоянных магнитных полей многослойным цилиндрическим экраном .....	80
<b>Липницкий С.Ф.</b> Индексирование текстовой информации на основе моделирования вербальных ассоциаций .....	94

## АВТОМАТИЗАЦИЯ ПРОЕКТИРОВАНИЯ

<b>Свирский Д.Н.</b> Формализация ранних стадий проектирования металлорежущего оборудования на основе QFD-моделирования .....	103
<b>Черемисинов Д.И.</b> Генерация выполнимых спецификаций цифровых систем из структурных описаний FPGA-проектов .....	111

---

---

Редактор Г.Б. Гончаренко  
Корректор А.А. Михайлова  
Компьютерная верстка В.И. Голенкевич

---

Сдано в набор 06.08.2012. Подписано в печать 03.09.2012.  
Формат 60×84 1/8. Бумага офсетная. Гарнитура Таймс.  
Усл. печ. л. 14,4. Уч.-изд. л. 14,1. Тираж 100 экз. Заказ 12.

---

Государственное научное учреждение «Объединенный институт проблем информатики Национальной академии наук Беларуси».  
ЛИ № 02330/0549421 от 08.04.2009.  
Ул. Сурганова, 6, 220012, Минск.

Отпечатано с оригинала-макета на ризографе Объединенного института проблем информатики Национальной академии наук Беларуси.  
Ул. Сурганова, 6, 220012, Минск.

© Объединенный институт проблем информатики  
Национальной академии наук Беларуси, 2012

THE UNITED INSTITUTE OF INFORMATICS PROBLEMS  
OF THE NATIONAL ACADEMY OF SCIENCES OF BELARUS

---

---

# INFORMATICS

---

---

PUBLISHED QUATERLY

*Issued since 2004*

---

---

№ 3(35) • July-September 2012

## CONTENTS

### PATTERN RECOGNITION, IMAGE, SIGNAL AND SPEECH PROCESSING

- Artemiev V.M., Naumov A.O., Kokhan L.L.** Synthesis of filters for suppression of background on digital images in optronic systems..... 5
- Prakapovich R.A.** Neuronet associative memory model for the robot manipulator control..... 16
- Pekar D.V., Tsikhanenka S.G., Chernyavskii A.F., Sadov V.S.** Algorithm for optimization of features and parameters of speaker's emotional state classifier ..... 26
- Shut O.V.** A solution method of recognition in boolean space based on logical and precedent models..... 35

### APPLIED INFORMATION TECHNOLOGIES AND INFORMATION SECURITY

- Lapitsky V.A., Lakhanko L.N., Batrakov A.G., Kirichenko V.V.** Distributed telemedicine system of digital mammography ..... 51
- Anishchanka U.V., Kulbak L.I.** Assessment of loss of information system processing capability..... 61
- Fralou H.V.** A system approach to information technology protection..... 70

## MATHEMATICAL MODELING

<b>Erofeenko V.T., Shushkevich G.Ch., Grabchikov S.S., Bondarenko V.F.</b> Model of constant magnetic fields screening by multilayer cylindrical screen .....	80
<b>Lipnitsky S.F.</b> Text information indexing on the basis of verbal associations modeling.....	94

## DESIGN AUTOMATION

<b>Svirsky D.N.</b> Early stages design formalization of the metal-cutting equipment on the base of QFD-modeling .....	103
<b>Cheremisinov D.I.</b> Generating executable circuit specifications from structural FPGA descriptions .....	111

*РАСПОЗНАВАНИЕ ОБРАЗОВ, ОБРАБОТКА ИЗОБРАЖЕНИЙ,  
СИГНАЛОВ И РЕЧИ*

УДК 621.391.268

**В.М. Артемьев, А.О. Наумов, Л.Л. Кохан****СИНТЕЗ ФИЛЬТРОВ ПОДАВЛЕНИЯ ФОНА НА ЦИФРОВЫХ  
ИЗОБРАЖЕНИЯХ В ОПТИКО-ЭЛЕКТРОННЫХ СИСТЕМАХ**

*Для обзорных оптико-электронных систем решается задача оптимальной фильтрации изображений объектов на облачном фоне. Используется условие подавления постоянной и линейной составляющих фона в пределах апертуры фильтра с минимизацией дисперсии шумовой составляющей. Синтез осуществляется для выделения изображений объектов, представленных точечными и многоточечными сигналами. Для решения задачи применяется метод оптимизации Лагранжа.*

**Введение**

Первым этапом обработки изображений в оптико-электронных системах (ОЭС) наблюдения воздушного пространства является выделение полезных сигналов объектов на фоне помех, к которым относятся изображения фона и шумов. Фоновая составляющая обусловлена наличием сигналов от облаков, местных предметов, Солнца и некоторых других. Поскольку пространственное положение местных предметов остается неизменным, а положение Солнца изменяется достаточно медленно и по известному закону, в процессе настройки системы угловые положения этих объектов на изображении бланкируются и исключаются из дальнейшей обработки. В работе рассматриваются лишь изображения облачного фона, существенно влияющего на качество работы системы.

Помимо фона на изображении присутствуют шумы, обусловленные рядом причин: светимостью неба, свойствами среды прохождения излучения, тепловыми шумами фотоприемного устройства, шумами квантования и др. По своим характеристикам они могут аппроксимироваться дискретными белыми шумами.

Изображения объектов занимают лишь малую область в поле кадра. При больших дальностях они представлены на изображении одним элементом и им соответствуют одноточечные сигналы, а при малых их размеры существенно возрастают, образуя многоточечный сигнал.

Подавления фона можно добиться двумя основными способами: вычитанием и фильтрацией. Сущность первого способа заключается в предварительной оценке изображения фона в кадре и его вычитании из исходного изображения. Его недостатком является большая вычислительная сложность. Способ фильтрации более прост и нагляден в реализации. Для фильтрации изображений в ОЭС могут использоваться линейные и нелинейные пространственные фильтры. Среди линейных наиболее широкое применение получили лапласианные фильтры [1, 2] и фильтры градиентного типа [3], среди нелинейных – разностно-медианные [4, 5]. Преимущество линейных фильтров заключается в том, что их выходные данные имеют законы распределения, близкие к гауссовым, что упрощает методику анализа. Нелинейная фильтрация менее чувствительна к учету законов распределения входных данных, однако приводит к негауссовым распределениям на выходе, что затрудняет методику исследования. В данной работе рассматриваются лишь линейные пространственные фильтры с учетом особенностей подавления облачных фонов при наличии шумов и сигналов объектов.

В настоящее время для снижения влияния облачного фона используются типовые фильтры, указанные выше. Они позволяют подавлять постоянную составляющую изображения в пределах апертуры фильтра за счет его дифференцирующих свойств, однако при этом разру-

шают изображения многоточечных полезных сигналов. Для их выделения используются вейвлет-фильтры [6], каждый из которых «настроен» на изображение объекта определенной размерности. Это приводит к необходимости применения многоканальных фильтров с соответствующими усложнениями.

В работе предлагается методика синтеза фильтров, подавляющих не только постоянную, но и линейную составляющую поля изображений и при этом позволяющих выделять точечные и многоточечные изображения объектов при условии минимизации дисперсии шумов на выходе.

### 1. Исходные соотношения

В метеорологии приводится подробная классификация облаков с различной структурой [7]. Однако с точки зрения обработки изображений их целесообразно классифицировать по двум признакам: сильно- и слабоструктурированные (рис. 1).

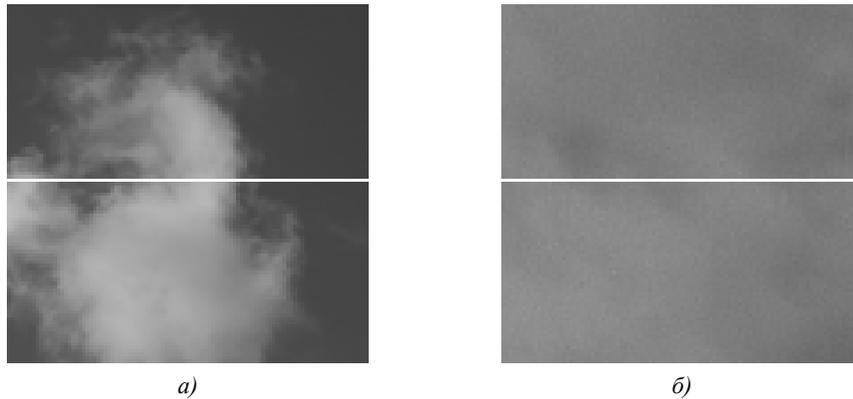


Рис. 1. Примеры изображений облачного фона: а) сильноструктурированного; б) слабоструктурированного

На рис. 2 показаны сечения изображений облачного фона вдоль горизонтальных линий, отмеченных на рис. 1.

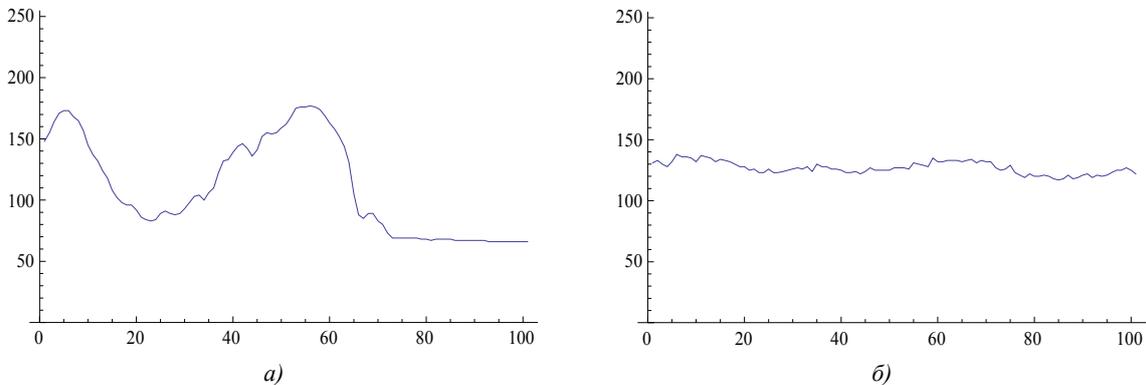


Рис. 2. Сечения изображений облачного фона: а) сильноструктурированного; б) слабоструктурированного

В дальнейшем будем рассматривать области с размерами в несколько элементов изображения, поэтому облачный фон в этих диапазонах можно аппроксимировать конечным числом членов двухмерного ряда Тейлора в окрестности средней точки области с координатами  $(i, j)$ :

$$v_{i+l, j+r} = v_{i,j}^{(0)} + \alpha_{1i,j}^{(1)}l + \alpha_{2i,j}^{(1)}r + \alpha_{1i,j}^{(2)}l^2 + \alpha_{2i,j}^{(2)}r^2 + \alpha_{12i,j}^{(2)}lr + \dots, \quad (1)$$

где  $\alpha_{ki,j}^{(i)}$  – коэффициенты разложения;  $v_{i,j}^{(0)}$  – среднее значение фона в области. Слагаемые

$$\alpha_{1i,j}^{(1)}l + \alpha_{2i,j}^{(1)}r = v_{i,j}^{(1)} \quad (2)$$

в дальнейшем называются линейной составляющей фона, а слагаемые

$$\alpha_{1i,j}^{(2)}J^2 + \alpha_{2i,j}^{(2)}r^2 + \alpha_{12i,j}^{(2)}lr = v_{i,j}^{(2)} \quad (3)$$

являются квадратичной составляющей. Сумма среднего значения  $v_{i,j}^{(0)}$  и линейной составляющей  $v_{i,j}^{(1)}$  будет в дальнейшем называться плоской составляющей.

Приведем известные уравнения линейных фильтров во временной области, необходимые для дальнейшего анализа. Пусть яркость изображения  $x_{i,j}$  на входе в точке с координатами  $(i, j)$  задается уравнением

$$x_{i,j} = u_{i,j} \oplus v_{i,j} + w_{i,j},$$

где  $u_{i,j}$  – сигнал объекта;  $v_{i,j}$  – сигнал фона;  $w_{i,j}$  – шумовая составляющая;  $\oplus$  – обозначение операции дизъюнктивной комбинации сигналов объекта и фона. Для дальнейшего анализа удобно ввести в рассмотрение величину  $q_{i,j}(v_{i,j})$  для разности полезного сигнала и фона:

$$q_{i,j}(v_{i,j}) = q_{i,j} = u_{i,j} - v_{i,j}. \quad (4)$$

Она определяет знак и уровень контраста полезного сигнала относительно фона в точке с координатами  $(i, j)$ . В результате смесь полезного сигнала, фона и помех приводится к аддитивной форме записи:

$$x_{i,j} = q_{i,j} + v_{i,j} + w_{i,j}, \quad (5)$$

где  $q_{i,j}$  зависит от уровня фона  $v_{i,j}$  и поэтому уравнение остается нелинейным.

На выходе фильтра имеет место сигнал, который в точке  $(i, j)$  определяется выражением [8]

$$y_{i,j} = \sum_{l=-n}^n \sum_{r=-m}^m x_{i+l,j+r} h_{i+l,j+r}, \quad (6)$$

где  $x_{i+l,j+r}$  – значения входных сигналов;  $h_{i+l,j+r}$  – весовые коэффициенты фильтра;  $(2n+1) \times (2m+1) = S_\Phi$  – размерность фильтра. Исходя из структуры входного сигнала (5), выходной будет состоять из трех составляющих  $y_{qi,j}$ ,  $y_{vi,j}$  и  $y_{wi,j}$ , соответствующих полезному сигналу, фону и шуму. Фоновая составляющая выходного сигнала может быть представлена в виде ряда в соответствии с формулой (1):

$$y_{vi,j} = \sum_{l=-n}^n \sum_{r=-m}^m \left( v_{i+r,j+l}^{(0)} + v_{i+r,j+l}^{(1)} + v_{i+r,j+l}^{(2)} + \dots \right) h_{i+l,j+r}. \quad (7)$$

Шумовой сигнал выходного изображения  $y_{wi,j}$  в дальнейшем определяется с учетом предположения о том, что шум на входе может быть аппроксимирован белым шумом с нулевым математическим ожиданием и дисперсией  $\sigma_{i,j}^2$ . Тогда математическое ожидание шума на выходе останется равным нулю, а его дисперсия

$$\sigma_{yi,j}^2 = \sum_{l=-n}^n \sum_{r=-m}^m \sigma_{i,j}^2 h_{i+l,j+r}^2. \quad (8)$$

При статистически однородном входном белом шуме с дисперсией  $\sigma_{i,j}^2 = \sigma^2$  выходная дисперсия  $\sigma_y^2 = \sigma^2 a$ , где  $a$  есть коэффициент преобразования шума:

$$a = \sum_{l=-n}^n \sum_{r=-m}^m h_{i+l,j+r}^2. \quad (9)$$

При отсутствии помех амплитуды полезного сигнала должны точно выделяться в каждом элементе изображения. Это приводит к необходимости выбора значения центрального коэффициента фильтра ( $r = 0, l = 0$ ), равного единице, т. е.

$$h_{i,j} = 1. \quad (10)$$

Тип фильтра определяется структурой матрицы его коэффициентов и синтезируется с учетом особенностей решаемой задачи с целью наилучшего выделения полезного сигнала, подавления фона и шумов.

## 2. Синтез фильтра точечных сигналов

На максимальных дальностях изображения объектов имеют минимальные размеры, в пределе равные одному элементу, и сигналы от них будут точечными. Этот случай представляет особый интерес для ОЭС наблюдения за воздушным пространством. Рассмотрим задачу синтеза фильтра точечных сигналов при наличии облачного фона и шумов. Будем определять требуемые значения его коэффициентов при следующих условиях: сохранение амплитуды точечного сигнала, полное подавление плоской составляющей фона и минимизация дисперсии однородных шумов на его выходе. По методическим соображениям первоначально рассмотрим вариант задачи с подавлением лишь постоянной составляющей фона, а в последующем распространим результаты на случай плоской составляющей. Сформулированные условия синтеза могут быть представлены следующими соотношениями:

$$\begin{cases} \sum_{l=-n}^n \sum_{r=-m}^m h_{i+l,j+r}^2 \rightarrow \min; \\ \sum_{l=-n}^n \sum_{r=-m}^m h_{i+l,j+r} = 0; \\ h_{i,j} = 1. \end{cases} \quad (11)$$

Первое условие соответствует минимизации дисперсии шумов на выходе фильтра, второе – подавлению постоянной составляющей фона, третье – сохранению амплитуды точечного сигнала.

Задачу определения оптимальных параметров фильтра можно решить с помощью метода Лагранжа [9]. Для этого используется лагранжиан в виде

$$L(h_{i+l,j+r}; \lambda_1; \lambda_2) = \sum_{l=-n}^n \sum_{r=-m}^m h_{i+l,j+r}^2 + \lambda_1 \sum_{l=-n}^n \sum_{r=-m}^m h_{i+l,j+r} + \lambda_2 (h_{i,j} - 1).$$

Его минимизация по искомым параметрам и неопределенным множителям  $\lambda$  приводит к следующим выражениям:

$$\begin{cases} h_{i,j} = 1; \\ h_{i+l,j+r} = -\frac{1}{(2n+1)(2m+1)-1} = -\frac{1}{S_\Phi - 1} = h; \end{cases} \quad (12)$$

$$a = \frac{(2n+1)(2m+1)}{(2n+1)(2m+1)-1} = \frac{S_\Phi}{S_\Phi - 1}. \quad (13)$$

Оказалось, что для введенных условий синтеза коэффициенты оптимального фильтра  $h_{i+l,j+r}$  (за исключением центрального  $h_{i,j}$ ) должны быть одинаковыми отрицательными величинами, зависящими от размерности фильтра  $S_\Phi = (2n+1)(2m+1)$ .

Оценим величину ошибки подавления линейной составляющей фона  $v_{i,j}^{(1)}$  при условии того, что  $h_{i,j} = 1$  и  $h_{i+l,j+r} = h$ . Ошибка определяется из выражения

$$y_{v,i,j}^{(1)} = \sum_{l=-n}^n \sum_{r=-m}^m v_{i+r,j+l}^{(1)} h_{i+l,j+r} = -\alpha_{1,i,j}^{(1)} h(2m+1) \sum_{l=-n}^n l - \alpha_{1,i,j}^{(2)} h(2n+1) \sum_{r=-m}^m r. \quad (14)$$

Поскольку суммы  $\sum_{l=-n}^n l = \sum_{r=-m}^m r = 0$ , то и ошибка будет равна нулю. Таким образом, синтезированный фильтр подавляет не только постоянную, но и линейную составляющие фона, т. е. его плоскую часть. Это говорит о том, что фильтр обладает свойством пространственного дифференцирования второго порядка и относится к классу лапласианных. Однако его структура отличается от подобных фильтров выделения краев, например ЛОГ-фильтров [10], поскольку рассматриваемый фильтр синтезирован для условий подавления плоских составляющих фона.

Если фон в пределах апертуры фильтра отличается от плоского, его неполное подавление можно оценить величиной ошибки за счет наличия квадратичной составляющей:

$$\begin{aligned} y_{v,i,j}^{(2)} &= \sum_{l=-n}^n \sum_{r=-m}^m v_{i+r,j+l}^{(2)} h_{i+l,j+r} = \\ &= -\alpha_{1,i,j}^{(2)} h(2m+1) \sum_{l=-n}^n l^2 - \alpha_{1,i,j}^{(2)} h(2n+1) \sum_{r=-m}^m r^2 + \alpha_{12,i,j}^{(2)} h \sum_{l=-n}^n \sum_{r=-m}^m lr = \\ &= -\frac{S_\Phi}{6(S_\Phi - 1)} \left[ \alpha_{1,i,j}^{(2)} n(n+1) + \alpha_{2,i,j}^{(2)} m(m+1) \right]. \end{aligned} \quad (15)$$

В табл. 1 приведены относительные величины этих ошибок для фильтров различной размерности при одинаковых значениях  $\alpha_{1,i,j}^{(2)} = \alpha_{2,i,j}^{(2)} = \alpha^{(2)}$ .

Таблица 1  
Относительные величины ошибок подавления фона за счет квадратичных составляющих

Размерность фильтра $S_\Phi$	Относительные ошибки подавления фона $y_{v,i,j}^{(2)}/\alpha^{(2)}$
3×3	-0,75
5×5	-2,08
7×7	-4,08
9×9	-6,75

Видно, что с ростом размерности фильтра увеличивается и ошибка за счет квадратичной составляющей.

Из формулы (13) для коэффициента преобразования шума  $a$  следует, что с ростом размерности фильтра дисперсия шумов на выходе уменьшается, однако остается большей ( $a > 1$ ) по сравнению с дисперсией на входе (табл. 2). Это происходит за счет дифференцирующих свойств фильтра.

Таблица 2  
Значения коэффициентов преобразования шума  
для фильтров точечных сигналов

Размерность фильтра $S_\Phi$	Коэффициенты преобразования шума $a$
3×3	1,125
5×5	1,041
7×7	1,02
9×9	1,01

Из сопоставления данных табл. 1 и 2 следует, что выбор размерности фильтра  $S_\Phi$  должен учитывать противоречивые свойства изменения ошибок подавления фона и дисперсии шумов.

### 3. Синтез фильтра многоточечных сигналов

Более сложной является ситуация, когда имеются несколько точечных сигналов, которые принадлежат протяженному в пространстве объекту, находящемуся на малой дальности. В таком случае задача фильтрации может быть сформулирована при следующих условиях: обеспечение подавления плоских участков фона, минимизация дисперсии шумовой составляющей на выходе, выделение всех элементов многоточечного сигнала.

Для решения данной задачи можно предложить фильтр размерности  $S_\Phi = (2n+1) \times (2m+1)$ , состоящий из двух частей. Основная часть, расположенная вокруг центральной точки, должна иметь размерность  $S_{\Phi_0} = (2k+1) \times (2t+1)$  с центральным коэффициентом  $h_{i,j} = 1$  и остальными коэффициентами, равными нулю. Помимо основной фильтр должен иметь периферийную часть с  $S_\Phi - S_{\Phi_0}$  элементами, отличными от нуля. Эти коэффициенты выбираются исходя из условия подавления плоской составляющей фона и минимизации дисперсии шумов. Выбор подобной структуры можно обосновать следующим образом. При отсутствии сигналов фона и шумов для выделения многоточечных сигналов объекта следует выбирать фильтр с нулевыми значениями коэффициентов, кроме центрального, равного единице. В противном случае на периферийной части фильтра следует использовать коэффициенты, отличные от нуля, исходя из условий подавления изображения плоского участка фона и минимизации дисперсии шумов. Однако при этом сократятся размеры полного выделения многоточечного изображения объекта, которые будут определяться неравенством  $S_0 \leq (k+1) \times (t+1)$ , что следует из рассмотрения уравнения фильтра (6).

Как и ранее, первоначально рассмотрим случай подавления лишь постоянной составляющей фона. В такой постановке коэффициенты фильтра должны выбираться с учетом соотношений

$$\left\{ \begin{array}{l} \sum_{l=-n}^n \sum_{r=-m}^m h_{i+l, j+r}^2 \rightarrow \min; \\ \sum_{l=-n}^n \sum_{r=-m}^{-t-1} h_{i+l, j+r} + \sum_{l=-n}^n \sum_{r=t+1}^m h_{i+l, j+r} + \sum_{l=-n}^{-k-1} \sum_{r=-t}^t h_{i+l, j+r} + \sum_{l=k+1}^n \sum_{r=-t}^t h_{i+l, j+r} = 0; \\ h_{i,j} = 1; \\ h_{i+l, j+r} = 0; \quad l = -k, -k+1, \dots, k; \quad l \neq 0; \quad r = -t, -t+1, \dots, t; \quad r \neq 0. \end{array} \right. \quad (16)$$

Первое условие определяет минимизацию дисперсии шума на выходе, второе – подавление постоянной составляющей фона за счет элементов периферийной части фильтра, третье и четвертое – выделение элементов многоточечного сигнала за счет нулевых коэффициентов центральной части фильтра. Решая задачу методом Лагранжа [9], приходим к следующим результатам:

$$\begin{cases} h_{i,j} = 1; \\ h_{i+l,j+r} = 0; \quad l = -k, -k+1, \dots, k; \quad l \neq 0; \quad r = -t, -t+1, \dots, t; \quad r \neq 0; \\ h_{i+l,j+r} = -\frac{1}{(2n+1)(2m+1) - (2k+1)(2t+1)} = -\frac{1}{S_\Phi - S_{\Phi_0}} = h; \end{cases} \quad (17)$$

$$a = \frac{(2n+1)(2m+1)}{(2n+1)(2m+1) - (2k+1)(2t+1)} = \frac{S_\Phi}{S_\Phi - S_{\Phi_0}}. \quad (18)$$

Как и в предыдущем случае, коэффициенты периферийной части фильтра оказываются одинаковыми, что обеспечивает подавление как постоянной, так и линейной составляющих фона в пределах апертуры фильтра. Величина  $S_0$  задает максимальный размер полного выделения изображения объекта. Размерность  $S_\Phi$  оказывается наименьшей при выборе периферийной части в виде одного слоя, окружающего центральную часть с размерностью  $S_{\Phi_0}$ . При этом размерность фильтра  $S_\Phi = (2k+3) \times (2t+3)$ .

Величина ошибки подавления фона за счет наличия квадратичной составляющей для фильтра с однослойной периферией находится из выражения

$$\begin{aligned} y_{v,i,j}^{(2)} &= \sum_{l=-n}^n \sum_{r=-m}^m v_{i+r,j+l}^{(2)} h_{i+l,j+r} = \\ &= -\frac{2}{3(S_\Phi - S_{\Phi_0})} \left[ \alpha_{1,i,j}^{(2)} n(n+1)(2n+1) + \alpha_{2,i,j}^{(2)} m(m+1)(2m+1) \right]. \end{aligned} \quad (19)$$

В табл. 3 приводятся величины относительных ошибок подавления фона за счет квадратичной составляющей при одинаковых значениях  $\alpha_{1,i,j}^{(2)} = \alpha_{2,i,j}^{(2)} = \alpha^{(2)}$  для фильтров с однослойной периферией различных размерностей.

Таблица 3  
Относительные величины ошибок подавления фона за счет квадратичной составляющей

Размерность фильтра $S_\Phi$	Относительные ошибки подавления фона $y_{v,i,j}^{(2)} / \alpha^{(2)}$
5×5	-2,5
7×7	-4,7
9×9	-7,5

При увеличении размерности фильтра возрастает влияние неоднородностей в пределах его апертуры, что и показывают приведенные результаты. Сравнение данных этой таблицы с результатами табл. 1 говорит о том, что выделение многоточечных сигналов приводит к увеличению ошибок подавления фона по сравнению с одноточечным случаем. Это является своеобразной «платой» за расширение возможностей фильтра.

Коэффициент преобразования шума  $a$  определяется формулой (18). В табл. 4 приведены его значения для фильтров с однослойной периферией.

Таблица 4  
Значения коэффициентов преобразования шума для фильтров многоточечных сигналов с однослойной периферией

Размерность фильтра $S_\Phi$	Коэффициенты преобразования шума $a$
5×5	1,56
7×7	2,04
9×9	1,53

Сравнение результатов табл. 2 и 4 для фильтров точечных и многоточечных сигналов позволяет сделать выводы о том, что у последних дисперсии шумовых помех выше, чем у первых. Однако фильтры многоточечных сигналов позволяют полностью выделять элементы изображений объектов конечной размерности посредством соответствующего выбора размерности центральной части фильтра.

#### 4. Примеры фильтрации

Рассмотрим пример фильтрации одноточечных сигналов. На рис. 3, а показана структура фильтра точечных сигналов минимальной размерности  $3 \times 3$ , а на рис. 3, б дан вариант изображения с облачным фоном и тремя искусственными точечными объектами без шумовой составляющей.

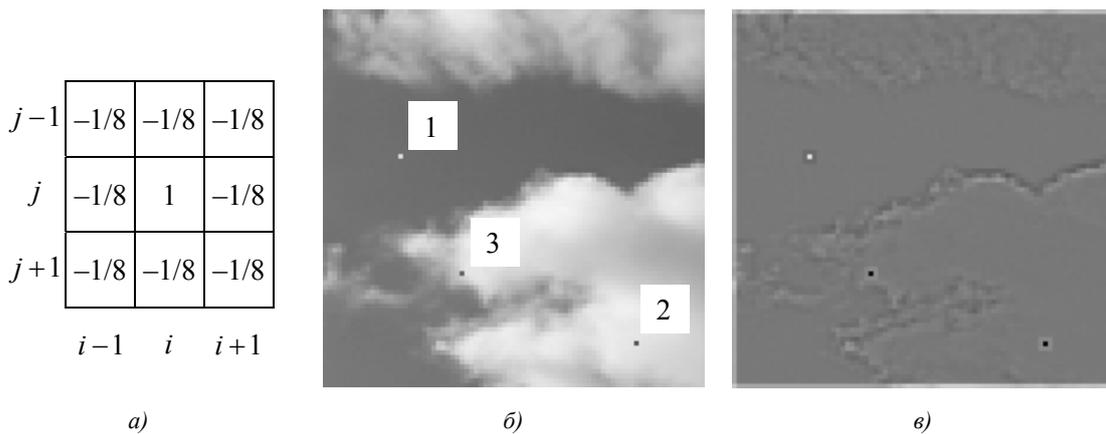


Рис. 3. Изображения искусственных точечных объектов на облачном фоне:  
а) фильтр точечных сигналов; б) исходное изображение; в) изображение на выходе фильтра

Объект 1 расположен на приблизительно постоянном фоне со средним уровнем  $v^{(0)} = 110,4$  и положительным значением сигнала контраста  $q = 108,6$ . Объект 2 также расположен на постоянном фоне  $v^{(0)} = 227,9$  и имеет отрицательное значение сигнала контраста  $q = -137,9$ . Для объекта 3 выбрано положение с неоднородным фоном, средним уровнем  $v^{(0)} = 205,3$  в точке положения сигнала и отрицательным контрастом  $q = -105,3$ .

На рис. 3, в показаны результаты фильтрации, где наряду с сигналами объектов видны остатки некомпенсированных элементов фона, которые расположены в окрестностях границ облаков. Это происходит по причине того, что облачный фон на границе неоднороден и отличается от плоского в пределах апертуры фильтра.

Результаты фильтрации можно оценить двумя параметрами: величиной выходного сигнала контраста  $q_y$  и величиной уровня фона  $v_y$  в точке положения сигнала (табл. 5).

Таблица 5  
Результаты фильтрации изображения на рис. 3, б

Номер объекта	Исходное изображение	Выходное изображение
1	$q = 108,6$ $v^{(0)} = 110,4$	$q_y/q = 0,9982$ $v_y/v^{(0)} = -0,0018$
2	$q = -137,9$ $v^{(0)} = 227,9$	$q_y/q = 1,0021$ $v_y/v^{(0)} = 0,0009$
3	$q = -105,3$ $v^{(0)} = 205,3$	$q_y/q = 1,0969$ $v_y/v^{(0)} = 0,0175$

Применительно к объектам 1 и 2, расположенным приблизительно на постоянном фоне, фильтрация приводит к существенной компенсации уровня фона. При этом амплитуды точечных сигналов в центре фильтра остаются практически неизменными. Сложнее ситуация с объектом 3, расположенным на неоднородном фоне, где степень компенсации фона значительно хуже, чем в предыдущем случае, в том числе за счет наличия квадратичной составляющей.

Кроме того, отметим, что при фильтрации изменяется и структура изображения точечного объекта. На рис. 4 в увеличенном масштабе показано выходное изображение точечного объекта 2.

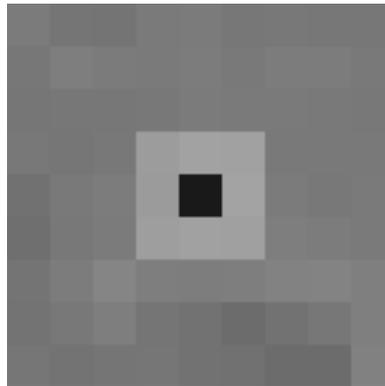


Рис. 4. Изображение точечного объекта на выходе фильтра с размерностью  $3 \times 3$

Вместо одноточечного на выходе появляется более сложное изображение, повторяющее структуру фильтра, т. е. выходное изображение расширяется, что приводит к ухудшению разрешающей способности. Однако следует отметить, что уровень боковых составляющих пропорционален значениям коэффициентов фильтра  $h$  и уменьшается с увеличением размерности фильтра. В итоге выбор размерности производится из компромиссных соображений с учетом условий подавления фона, минимизации дисперсии шумов и допустимой разрешающей способности.

В качестве второго примера рассмотрим фильтрацию многоточечных сигналов искусственных объектов квадратной формы с одинаковой яркостью элементов и размерностями  $S_0 = 2 \times 2$  и  $S_0 = 7 \times 7$  на облачном фоне, показанном на рис. 3, б. С этой целью использован фильтр, периферийная часть которого состоит из одного слоя коэффициентов, отличных от нуля (рис. 5, а).

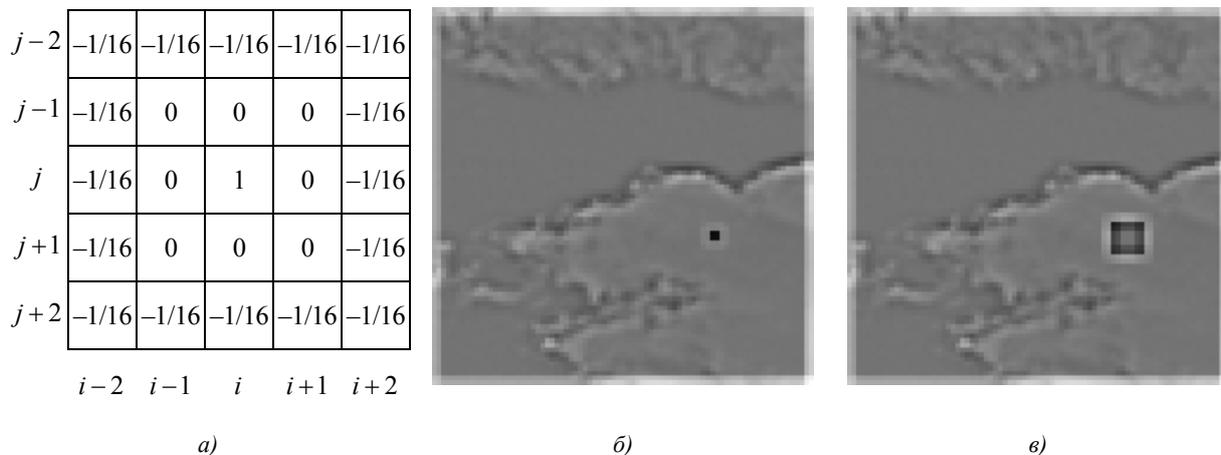


Рис. 5. Изображение искусственных многоточечных объектов на выходе фильтра размерности  $5 \times 5$ : а) структура фильтра; б) выходное изображение объекта размерности  $2 \times 2$ ; в) выходное изображение объекта размерности  $7 \times 7$

На рис. 5, б и в приведены результаты фильтрации изображений объектов с одинаковыми сигналами контраста  $q = -137,9$ , расположенными на постоянном фоне с  $v^{(0)} = 227,9$ . Поскольку фильтр «настроен» на размерность объекта  $2 \times 2$ , на его выходе структура изображения такого объекта сохраняется полностью. В то же время изображение с размерностью  $7 \times 7$  восстанавливается неполностью и содержит границы шириной в два элемента изображения.

Приведем пример фильтрации реальных изображений. На рис. 6, а дается изображение объекта на облачном фоне в инфракрасном диапазоне (8–12 мкм). Изображение объекта имеет размерность  $10 \times 10$  элементов. Для обработки использовались фильтры с размерностями  $3 \times 3$  и  $21 \times 21$ . В соответствии со сказанным выше второй вариант фильтра позволяет полностью выделить изображение объекта. На рис. 6, б и в показаны результаты обработки рассмотренными фильтрами.

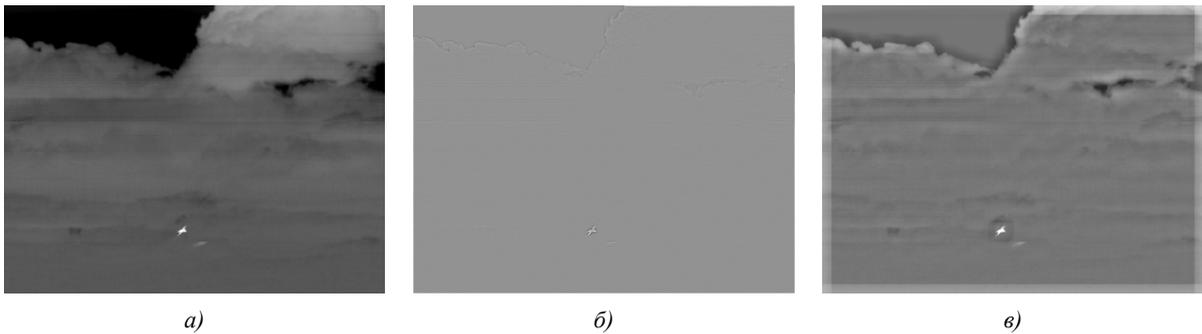


Рис. 6. Изображение объекта на облачном фоне: а) исходное; б) обработка фильтром размерности  $3 \times 3$ ; в) обработка фильтром многоточечных сигналов размерности  $21 \times 21$

Из рис. 5 и 6 видно, что фильтр размерности  $3 \times 3$  существенно лучше подавляет изображение облачного фона. На рис. 7 изображены объекты в увеличенном масштабе.

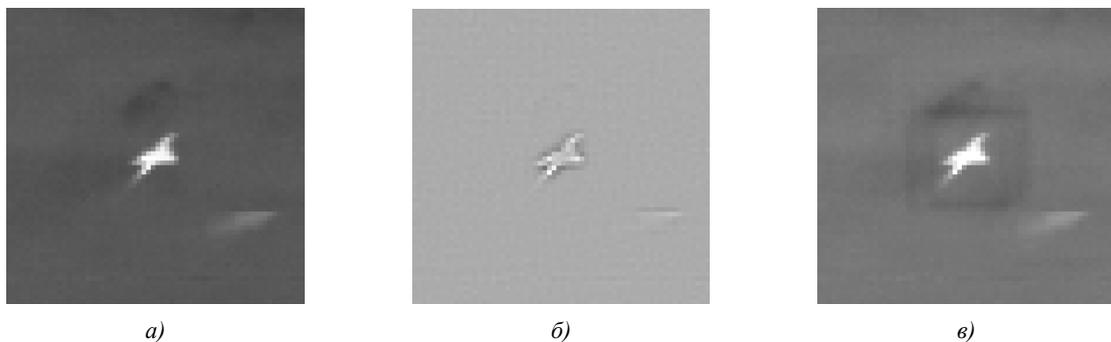


Рис. 7. Изображение объекта в увеличенном масштабе: а) исходное; б) обработка фильтром размерности  $3 \times 3$ ; в) обработка фильтром многоточечных сигналов размерности  $21 \times 21$

Результаты показывают, что увеличение размерности фильтра улучшает качество изображения объекта, однако ухудшает степень подавления облачного фона. При выборе размерности фильтра следует исходить из компромиссных условий, учитывающих особенности конкретной системы.

### Заключение

В работе решена задача синтеза фильтров, обеспечивающих подавление постоянной и линейной составляющих облачного фона в пределах его апертуры при условии минимизации дисперсии шумов на выходе и полного выделения элементов изображений сигналов объектов. На основе метода оптимизации Лагранжа получены формулы для нахождения параметров таких

фильтров. Эффективность фильтрации показана на примерах выделения изображений объектов различной размерности на облачном фоне.

### Список литературы

1. Robinson, D.Z. Methods of background description and their utility / D.Z. Robinson // Proc. IRE. – 1959. – Vol. 47, № 9. – P. 1554–1561.
2. Hwang, S. Spatial filter performance on point-target detection in various clutter conditions: M. E. thesis / S. Hwang. – Massachusetts Institute of Technology, USA, 2007. – 88 p.
3. Reiss, D. Spatial signal processing for infrared detection / D. Reiss // SPIE. – 1994. – Vol. 2235. – P. 38–51.
4. Barnett, J.T. Statistical analysis of median subtraction filtering with application to point-target detection in infrared backgrounds / J.T. Barnett // Proc. Infrared Systems and Components. – 1989. – Vol. 1050. – P. 10–18.
5. Цифровая обработка сигналов. Лекции и практикум на ПК [Электронный ресурс]. – Минск, 2009. – Режим доступа : <http://www.prodav.narod.ru/dsp/dsp16.htm>. – Дата доступа : 30.10.2009.
6. Штарк, Г.-Г. Применение вейвлетов для цифровой обработки сигналов / Г.-Г. Штарк. – М. : Техносфера, 2007. – 183 с.
7. Атмосфера : справочник / Ю.С. Седунов [и др.] ; под ред. Ю.С. Седунова. – Л. : Гидрометеоиздат, 1991. – 510 с.
8. Методы компьютерной обработки изображений / М.В. Гашников [и др.] ; под общ. ред. В.А. Сойфера. – М. : Физматлит, 2003. – 784 с.
9. Банди, Б. Методы оптимизации. Вводный курс / Б. Банди ; пер. с англ. О.В. Шихеевой ; под ред. В.А. Волынского. – М. : Радио и связь, 1988. – 128 с.
10. Шапиро, Л. Компьютерное зрение / Л. Шапиро, Д. Стокман. – М. : Бином, 2006. – 752 с.

Поступила 24.05.12

*Институт прикладной физики  
НАН Беларуси,  
Минск, Академическая, 16  
e-mail: [naumov@iapf.bas-net.by](mailto:naumov@iapf.bas-net.by)*

**V.M. Artemiev, A.O. Naumov, L.L. Kokhan**

### **SYNTHESIS OF FILTERS FOR SUPPRESSION OF BACKGROUND ON DIGITAL IMAGES IN OPTRONIC SYSTEMS**

For survey optronic systems, the problem of an optimal filtration of objects images on cloud background is studied. A condition for suppression of constant and linear components of the background within the aperture of the filter together with the minimization of the noise component variance is used. The synthesis is implemented for a selection of objects images presented by pixel-size and finite-size signals. The problem is solved by the Lagrange multipliers method.

УДК 004.272.25, 004.622

Г.А. Прокопович

## МОДЕЛЬ НЕЙРОСЕТЕВОЙ АССОЦИАТИВНОЙ ПАМЯТИ ДЛЯ УПРАВЛЕНИЯ МАНИПУЛЯТОРОМ

*Рассматривается проблема управления динамическими системами, обладающими большим числом степеней свободы, на примере шестизвенного антропоморфного манипулятора UNIMATE PUMA-762. Предлагается модель нейросетевой ассоциативной памяти, основанной на записи иерархической последовательности движений манипулятора с помощью синаптических весов нейроподобной сети «двунаправленная ассоциативная память». Проводится ряд численных экспериментов, доказывающих возможность реализации схемы позиционного управления с помощью предложенной нейросетевой модели памяти.*

### Введение

Быстрый рост числа и разнообразия роботов в последнее время объясняется их большими функциональными возможностями для применения не только в науке и промышленности, но и в социально значимых сферах.

Как правило, робот состоит из двух основных частей – исполнительных систем (эффекторов) и информационно-управляющей системы, снабженной сенсорами. Общий вид робота зависит от системы координат, в которой манипулятор со схватом совершает рабочие операции; количества манипуляторов; конструкции опор крепления или устройств передвижения; типов двигателей и источников энергии [1].

В настоящее время основным типом манипуляционных систем роботов являются механические манипуляторы, оснащенные рабочим органом, причем для перемещения в пространстве рабочей зоны им достаточно трех степеней подвижности. Однако для расширения манипуляционных возможностей и реализации более сложных траекторий движения, например с обходом препятствий, а также для повышения быстродействия манипуляторы обычно снабжают несколькими избыточными степенями подвижности. Подобные манипуляторы необходимы как в промышленности, так и для работы в экстремальных условиях.

Для большинства роботов каждое звено приводится в движение от своего привода. Взаимодействие звеньев обеспечивается системой управления, задачей которой является быстрое и точное формирование управляющих команд. Поэтому робот должен быть снабжен системой управления, состоящей из блока памяти, в котором записаны программы перемещения всех звеньев, а также вычислительным блоком, который должен последовательно извлекать из блока памяти соответствующие координаты.

Стандартными задачами для современного робота являются манипулирование деталями и предметами, определение состояния внешней среды и перемещение в пространстве, осмысление и принятие решений, а также участие в диалоге с человеком-оператором. Несмотря на большие успехи развития информатики и робототехники, с технической точки зрения реализация указанных функций очень сложна даже при последовательном их выполнении.

### 1. Постановка задачи

Целью данной работы является разработка антропоморфной системы управления манипулятором промышленного робота, обладающей такими качествами, как адаптивность и способность к самоорганизации. Под адаптивностью понимается возможность смены одной пространственной траектории движения манипулятора на другую при изменении внешних условий, а под самоорганизацией – возможность самостоятельного изменения внутренней структуры системы управления, что позволит ей обучаться новым траекториям не с помощью программного добавления новых координат, а по примерам. Таким образом, для достижения указанной цели была поставлена задача разработать и исследовать нейроподобную модель памяти для системы управления манипулятором.

Головной мозг и нервная система человека устроены таким образом, что для воспроизведения заранее заученных действий не требуется знать последовательность и величину усилий каждой из огромного числа участвующих в процессе движения мышц. Как правило, это достигается автоматическим извлечением из памяти управляющих сигналов для соответствующих групп мышц [2]. Однако на пути создания роботов, обладающих подобными свойствами, специалисты встретились со значительными трудностями, которые с помощью современных методов управления и обработки информации до сих пор еще не преодолены.

Одним из возможных способов управления такими сложными системами является использование нейроконтроллеров, архитектура которых функционирует по аналогии с нервной системой живых организмов [3–6]. Данное направление связано с построением моделей искусственных нейронных сетей, состоящих из элементарных функциональных элементов, напоминающих нервные клетки мозга. Важная особенность нейросетевых систем состоит в параллельной и распределенной обработке информации.

Традиционными областями применения искусственных нейронных сетей являются задачи управления, распознавания образов и звуков, прогнозирования, цифровой обработки сигналов, эмуляции ассоциативной памяти. При этом изучению ассоциативной памяти уделяется большое внимание, так как на основе наблюдений и опытов выдвигаются теории, что именно этот процесс является одним из главных принципов эффективной обработки информации мозгом живых существ [7–9]. Более того, наблюдения и исследования различных типов поведения живых организмов показывают, что они больше зависят от свойств и возможностей самой памяти, чем от степени возможностей оперирования с ее содержанием или от возможности преобразования хранящейся в ней информации [10]. Другими словами, организация сложных форм поведения зависит от организации работы механизмов памяти.

## 2. Гетероассоциативная память как среда для обработки информации

Информация в головном мозге человека хранится распределенно с помощью формирования пространственных образов на большом количестве взаимосвязанных нейронов, которые функционируют почти параллельно [7]. С одной стороны, в распределенной памяти каждый запоминающий элемент или участок запоминающей среды содержит следы множества образов, т. е. имеет место пространственное наложение запоминаемых образов. С другой стороны, каждый элемент записываемой информации распределяется по большой области запоминающей среды. Это приводит к тому, что, несмотря на присутствие в выходных образах естественных шумов и искажений, они оптимально корректируются относительно всего объема информации, содержащейся в ключевом образе [11].

Благодаря ассоциативным свойствам память человека имеет огромную емкость и способна к адаптивной селекции поступающей информации [8]. Среди известных моделей, описывающих принципы работы ассоциативной памяти, можно выделить искусственную нейронную сеть Б. Коско, или двунаправленную ассоциативную память (ДАП) [12, 13]. Сеть ДАП, как и широко известная сеть Хопфилда, способна к обобщению, вырабатывая правильные реакции даже на искаженные входы. Данные сети показывают, каким образом в принципе может быть организована память, состоящая из коллектива недостаточно надежных элементов. Однако сеть Хопфилда является автоассоциативной. Это означает, что входной образ может быть завершен или исправлен, но не может быть ассоциирован с другим образом. Данный факт является результатом одноуровневой структуры ассоциативной памяти, в которой выходной вектор генерируется на тех же нейронах, на которые поступает входной вектор.

Двухслойная сеть ДАП является гетероассоциативной, когда входной вектор  $X^{(n)}$  поступает на один набор нейронов, а соответствующий выходной вектор  $Y^{(m)}$  с помощью матрицы весов  $W_{m \times n}$  вырабатывается на другом наборе нейронов. После этого полученный вектор-отклик  $Y^{(m)}$  через элементы обратной связи  $t_A$  и синаптические веса матрицы  $W_{m \times n}^T$  взаимодействует с первым слоем нейронов, в результате чего восстанавливается ключевой вектор  $X^{(n)}$ . Поэтому обученная сеть ДАП на поступающие двоичные ключевые векторы, обладающие общими характерными признаками, вырабатывает на выходе один и тот же вектор-отклик, т. е. наблюдается процесс адресации памяти по ее содержанию.

Используя принципы функционирования сети ДАП, а также теорию адаптивного резонанса С. Гроссберга [14], в статье [15] автором предложен гибридный нейросетевой классификатор с функцией дообучения, на основе которого в [16] был разработан нейросетевой блок памяти для интеллектуальной системы управления автономным мобильным роботом. Основным назначением указанного блока памяти является установление ассоциативных причинно-следственных связей между внешними возбуждающими признаками, действующими на объект, и его ответной реакцией. В результате сигнал о наличии возбуждающего признака, которым может быть графический образ, звук или данные тактильных датчиков, должен замещать собой весь объект, т. е. вызывать представление о целом объекте, которому присущ данный признак.

Дальнейшим развитием идей использования сети ДАП стала итерационная модель нейросетевой ассоциативной памяти, которая может применяться для записи, хранения и обработки информации в виде ассоциативных последовательностей [17]. Суть предложенной модели заключается в том, что сеть ДАП способна запоминать бинарные образы, длина которых многократно превышает размер ее входного нейронного слоя. Это достигается благодаря использованию нового способа функционирования, а также соответствующего алгоритма предобработки начальных данных.

Предложенная модель функционирования сети ДАП отличается от классической тем, что в сеть добавляется дополнительная обратная связь  $B$  (рис. 1), которая идет от второго слоя к первому напрямую через элементы задержки  $t_B$ , не взаимодействуя с весовой матрицей синаптических связей  $W_{m \times n}^T$ . При этом следует учесть, что введение обратной связи  $B$  требует от сети, чтобы число выходных нейронов равнялось числу входных нейронов, т. е.  $m = n$ . Процесс генерации информации внутри сети остался без изменений [17].

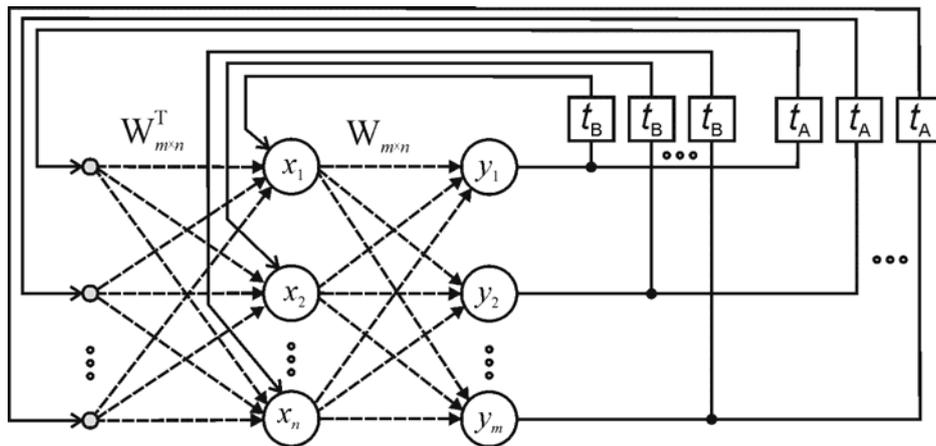


Рис. 1. Обобщенная структура сети ДАП

Алгоритм предобработки (секционирования) входных данных заключается в представлении запоминаемого бинарного вектора  $\mathbf{X}^{(p)} = (x_1, x_2, x_3, \dots, x_p)^T$ , содержащего некую информацию, в виде последовательности  $k$  уникальных векторов  $\mathbf{x}_i$  меньшей, но одинаковой длины  $n$  (рис. 2, а и б). По предложенному алгоритму величины  $k$  и  $n$  выбираются исходя из следующих условий:

$$\begin{cases} \mathbf{x}_i \neq \mathbf{x}_j, i \neq j, i, j = \overline{1, k}; \\ n = p/k; \\ p \bmod k = 0, \end{cases} \quad (1)$$

где  $n$  – число нейронов во входном слое сети ДАП.

Предложенный метод записи информации заключается в запоминании вектора  $\mathbf{X}^{(p)} = \mathbf{x}_1 \oplus \mathbf{x}_2 \oplus \mathbf{x}_3 \oplus \dots \oplus \mathbf{x}_k = \bigoplus_{i=1}^k \mathbf{x}_i$  как замкнутой последовательности пар уникальных векторов

$x_1 \rightarrow x_2, x_2 \rightarrow x_3, \dots, x_{k-1} \rightarrow x_k$  и  $x_k \rightarrow x_1$  с помощью сети ДАП (рис. 2, в). Таким образом, обученная сеть ДАП при вводе в нее вектора  $x_i$  на выходе сгенерирует вектор  $x_{i+1}$ . В результате реализуется отображение специального вида  $x_{i+1} = F(x_i)$ , причем  $x_1 = F(x_k)$ , где  $k$  – количество информационных блоков, на которые был разбит первоначальный вектор  $X^{(p)}$ . Другими словами, носителями информации являются предельные циклы фазового пространства состояний сети ДАП. В качестве примера на рис. 2 показан процесс записи вектора  $X^{(12)}$ , разбитого на четыре равные части.

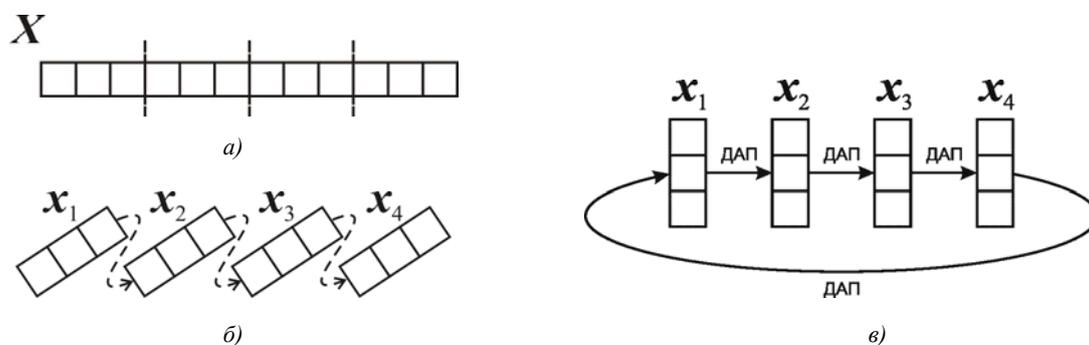


Рис. 2. Запись вектора  $X$  по предложенному методу: а) и б) подготовка; в) запись

После проведенного процесса обучения сеть ДАП способна восстанавливать и идентифицировать вектор  $X^{(p)}$  по любому известному ключевому вектору  $x_i$ . Так как итерационный процесс извлечения информации является циклическим, предложенный метод регламентирует и правило останова. Процесс распознавания считается успешно завершенным, если за конечное число итераций на выходе сети ДАП появится вектор, идентичный ключевому. Если описанное правило не выполняется, то считается, что начальная точка не попала в зону аттрактора фазового пространства состояний сети ДАП.

Благодаря тому что сеть ДАП работает с бинарными векторами, она способна хранить, восстанавливать и обрабатывать информацию различных типов. Это могут быть как текстовые, так и графические данные, предварительно закодированные в бинарный вид. Таким образом, описанную итерационную модель ассоциативной памяти в некотором роде можно считать универсальной средой обработки информации.

### 3. Позиционная система управления на основе нейросетевой модели достижения цели

Анализ процесса итерационного извлечения информации с помощью описанной выше модели ассоциативной памяти позволяет заметить наличие строгой зависимости порядка следования информационных блоков. Поэтому в работе [18] было предложено использовать данные свойства для имитации поведения реальных динамических систем, характеризующихся вертикальной иерархической структурой. В указанной модели предполагается, что для достижения системой управления роботом какой-либо цели она может быть разбита на иерархию подцелей, которые представляют собой набор элементарных действий эффекторов, подкрепленных соответствующими ответными значениями показаний рецепторов. Полученная последовательность элементарных действий должна быть записана в гетероассоциативную память в виде последовательностей ассоциируемых друг с другом векторных пар типа стимул-отклик. Преимуществом описываемой модели достижения цели является простота реализации, а также наличие адаптивных свойств, заключающихся в простой схеме корректировки подцелей.

В качестве примера иерархического поведения динамических систем можно привести тривиальную задачу из повседневной жизни, когда человеку требуется взять рукой стакан со стола. Несмотря на кажущуюся простоту, в процессе реализации потребуются выполнение ряда последовательных действий, заключающихся в сокращении и расслаблении определенных групп мышц плеча, предплечья, локтя, кисти и пальцев руки. Если всю траекторию движения руки разбить на ряд характерных точек, то сочетания величин напряжения всех мышц в каждой из характерных точек будут различны. Другими словами, чтобы кисть руки попала в конкретную точку пространства, головной мозг должен послать различным группам мышц строгую



будет воздействовать на исполнительные механизмы и изменять их пространственное положение. Блок СР сравнивает реальные значения  $\Theta$ , полученные из блока С, и ожидаемые, которые к нему поступают из ГАП. Если реальные значения  $\Theta$  с определенной степенью точности совпадут с заданными от ГАП, блок СР выдаст положительный сигнал блоку Ш. При получении положительного сигнала Ш позволит передать текущие данные от С на вход ГАП в таком виде, который понимает ГАП. В результате этого через элемент единичной задержки  $\Delta t$  замкнется обратная связь и на выходе блока ГАП появится уже второй информационный вектор. Аналогичным образом, если будет достигнут промежуточный результат и блок СР выдаст положительный сигнал, второй информационный вектор извлечет из ГАП третий и т. д.

Во избежание непредусмотренных действий обучение модели производится таким образом, чтобы последний информационный блок не возбуждал в ГАП новых управляющих сигналов. В итоге предложенная модель гетероассоциативной памяти позволяет генерировать строго определенную последовательность управляющих сигналов, которые могут быть интерпретированы как инструкции к выполнению каких-либо активных действий различными техническими устройствами.

#### 4. Эксперименты

С целью доказательства работоспособности предложенной модели памяти для управления манипулятором было решено провести ряд численных экспериментов, которые потребовали решения следующих задач:

- выбор соответствующего объекта, обладающего достаточно большим количеством степеней подвижности и точной математической моделью;
- выбор способа кодирования пространственных паттернов объекта в бинарный вид;
- определение максимального числа пространственных паттернов, которые могут безошибочно восстанавливаться из сети ДАП, чтобы реализовать максимально гладкую траекторию перемещения схвата между крайними точками.

Для проведения эксперимента в качестве исследуемого объекта был выбран промышленный робот-манипулятор UNIMATE PUMA-762. Выбор данной модели был обусловлен не только тем, что семейство роботов PUMA хорошо известно и широко используется на отечественных предприятиях, но и благодаря его антропоморфному строению, которое имеет шесть вращательных степеней подвижности (не считая схвата).

Эксперимент проводился в два этапа. На первом этапе с помощью виртуального пульта управления задавались начальные и конечные положения звеньев манипулятора (рис. 4, *а* и *б* соответственно), который вычерчивает в пространстве кривую, представляющую собой часть окружности. Начальные и конечные значения углов для каждой степени подвижности были записаны на электронный носитель (таблица). Эксперимент проводился на компьютерной трехмерной модели робота PUMA-762, реализованной в пакете MatLab [19].

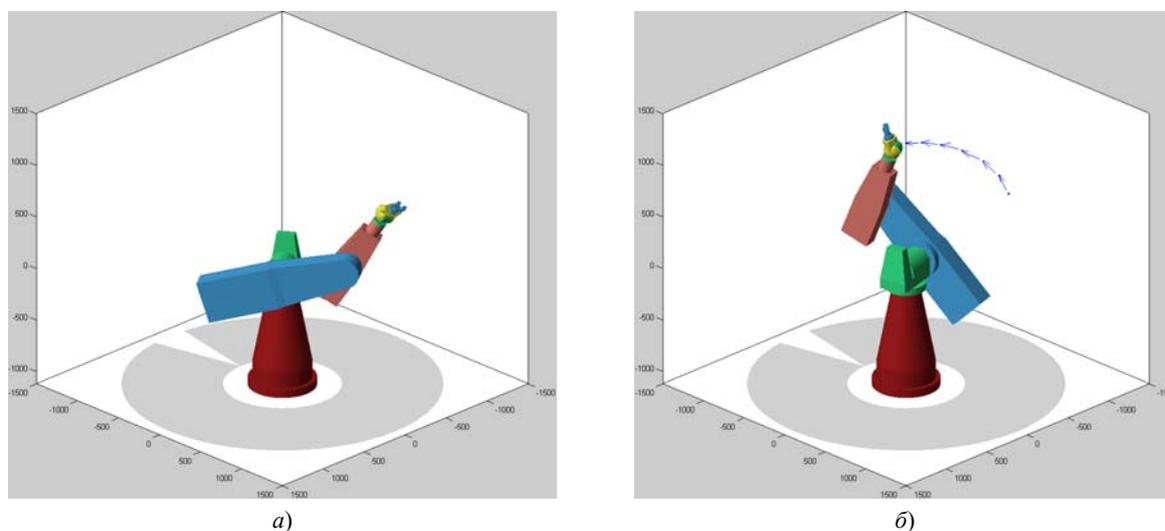


Рис. 4. Положения звеньев манипулятора: *а*) начальные; *б*) конечные

Второй этап эксперимента заключался в непосредственном моделировании позиционного способа управления, описанного в предыдущем разделе, согласно которому звенья манипулятора должны совершать вращательные движения и повторять заранее записанную траекторию. Для этого в предложенную нейросетевую модель памяти требовалось записать соответствующие паттерны пространственной траектории. Как уже говорилось выше, в позиционных системах управления не регламентируется положение звеньев между крайними точками во время движения. Поэтому промежуточные точки траектории было решено выбирать таким образом, чтобы в плоскости вращения каждого звена они равно отстояли друг от друга.

Значения положений звеньев манипулятора

Положение	Номер звена					
	1	2	3	4	5	6
Начальное, °	105,04	-134,81	-23,93	-165,48	37,36	-168,37
Конечное, °	-56,23	-162,82	-49,74	123,28	29,55	-26,11

В качестве примера на рис. 5, а показано последовательное изменение значений углов поворота подвижных звеньев манипулятора, которые образуют пространственную траекторию, состоящую из 19 отрезков, между двумя заданными в таблице крайними точками. Следует учесть, что в эксперименте по моделированию позиционного управления манипулятором не учитывалось время движения его звеньев.

По принятому методу кодирования каждый пространственный паттерн представляется бинарным вектором, состоящим из значений шести углов основных звеньев манипулятора. Далее значения углов округлялись до десятых и переводились с помощью двоичной системы счисления в вектор длиной 8 бит, а направление вращения кодировалось двумя битами: знак «-» – битами 00 и 01, а «+» – соответственно 10 и 11. Избыточность необходима для уменьшения влияния шумов в сети ДАП. Таким образом, блок ГАП реализуется на основе нейросети ДАП, состоящей из 60 нейронов и имеющей соответственно весовую матрицу  $W_{60 \times 60}$ .

Однако с помощью выбранного способа кодирования сеть ДАП способна качественно восстановить лишь небольшое число паттернов. Это связано со спецификой работы рекуррентных искусственных нейронных сетей, обучаемых по правилу Хэбба [6]. Их функционирование очень сильно зависит от степени корреляции между запоминаемыми образами [11, 13]. Сеть ДАП, как и сеть Хопфилда, имеет свое теоретическое ограничение на число запоминаемых образов, которое составляет  $V_{\text{ДАП}} \leq \frac{n}{4 \log n}$ , где  $n$  – число нейронов в слое [20]. Таким образом,

при выбранной в данной работе кодировке сеть ДАП способна восстановить не более двух образов ( $V_{\text{ДАП}} = 2,54 \approx 2$ ). В связи с этим часто применяются дополнительные ортогональные преобразования, которые увеличивают число запоминаемых образов [6].

Для решения указанной проблемы, т. е. декорреляции входных и выходных векторов, была модифицирована сеть ДАП путем добавления промежуточного слоя, состоящего из 24 элементов. В итоге получилась трехслойная сеть, синаптические связи которой содержатся в двух весовых матрицах  $W'_{60 \times 24}$  и  $W''_{24 \times 60}$ . Это позволило значительно увеличить емкости памяти и приступить к выполнению запланированных экспериментов. Так как увеличение емкости памяти искусственных нейронных сетей является отдельным направлением и не является целью текущей работы, устройство и принцип действия предложенной модификации более подробно будут рассмотрены в следующих работах.

Чтобы определить максимальное число паттернов, необходимых для реализации максимально гладкой траектории перемещения схвата между крайними точками, было решено последовательно уменьшать число запоминаемых образов путем деления пройденной траектории на меньшее число промежуточных точек. Полученные результаты (рис. 5, б-г) демонстрируют качество ассоциативного извлечения последовательности образов длиной 18, 16 и 14 образов соответственно.

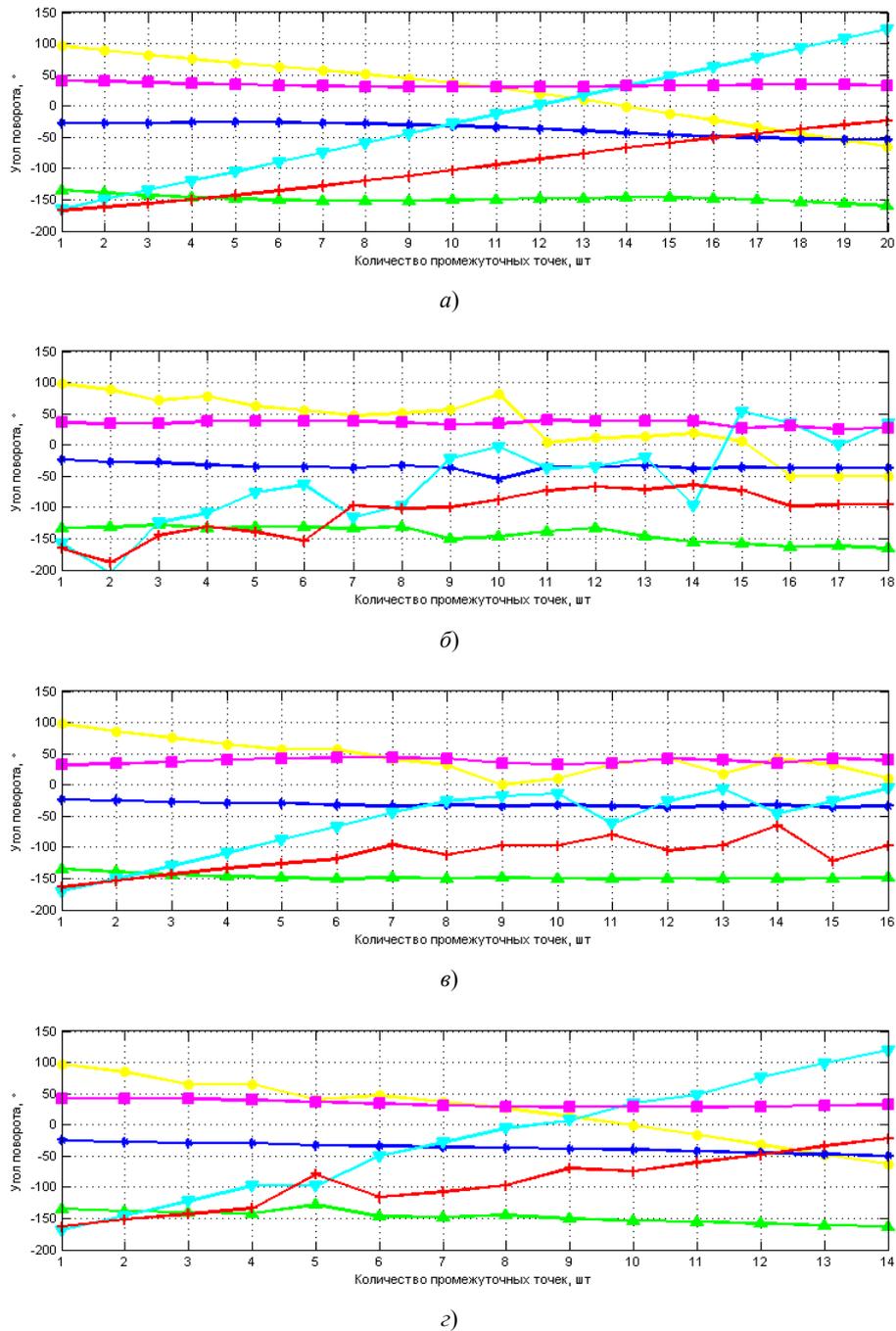


Рис. 5. Значения численного эксперимента

Как видно из графиков, деление траектории на 18 точек позволяет качественно восстанавливать из предложенной итерационной модели ассоциативной памяти только пять первых пространственных паттернов. Если в память записать 16 образов, то качественно могут восстановиться уже 10 первых образов. И только при записи 14 образов можно наблюдать почти 100%-е совпадение крайних положений манипулятора. Таким образом, поставленная задача позиционного управления манипулятором может быть решена при помощи гетероассоциативной памяти. Эмпирическим путем было найдено, что для описанного примера число образов, на которое делится траектория движения, не должно превышать  $0,23n$ , где  $n$  – число входных нейронов и одновременно длина вектора, содержащего информацию о соответствующих пространственных паттернах.

### Заключение

В работе рассматривалась возможность моделирования такой важной функции памяти головного мозга живых организмов, как хранение данных и программ управления эффекторами. Предложена схема позиционного управления, основанная на нейросетевой модели памяти, способная к ассоциативной записи и последующему воспроизведению состояний различных динамических систем.

Численные эксперименты показали возможность решения поставленной задачи управления манипулятором с помощью нейросетевой модели памяти, основанной на сети ДАП. При некотором оптимальном подборе параметров  $p$  и  $k$  системы (1) предложенная схема позиционного управления позволила безошибочно переместить рабочий орган манипулятора в заданную пространственную точку. В проводимых экспериментах размер входного слоя  $n$  оставался фиксированным и равнялся длине бинарных векторов, кодирующих паттерны пространственного положения звеньев манипулятора. Полученные результаты также показали, что неоптимальный подбор параметров  $p$  и  $k$  приводит к накоплению ошибки в процессе извлечения информационных блоков, состоящих из значений углов между звеньями.

В связи с этим целью дальнейших исследований является решение ряда задач, включающих выбор универсальных способов кодирования и представления состояний исследуемых объектов, а также определение точности и надежности используемых ассоциативных сетей, выбранных в качестве блока гетероассоциативной памяти. Немаловажной темой дальнейших исследований является биологическая интерпретация блока сравнения в предложенной модели моторной памяти.

### Список литературы

1. Юревич, Е.И. Основы робототехники / Е.И. Юревич. – СПб. : БХВ-Петербург, 2010. – 416 с.
2. Хакен, Г. Принципы работы головного мозга: синергетический подход к активности мозга, поведению и когнитивной деятельности / Г. Хакен. – М. : ПЕРСЭ, 2001. – 351 с.
3. Нейрокомпьютеры и интеллектуальные роботы / Н.М. Амосов [и др.]. – Киев : Наукова думка, 1991. – 272 с.
4. Галушкин, А.И. Теория нейронных сетей / А.И. Галушкин. – М. : ИПРЖР, 2000. – 415 с.
5. Сигеру, О. Нейроуправление и его приложения / О. Сигеру, Х. Марзуки, Ю. Рубия. – М. : ИПРЖР, 2000. – 272 с.
6. Головкин, В.А. Нейронные сети: обучение, организация и применение. Кн. 4 : учеб. пособие для вузов / В.А. Головкин ; под общ. ред. Л.И. Галушкина. – М. : ИПРЖР, 2001. – 256 с.
7. Прибрам, К. Языки мозга / К. Прибрам. – М. : Прогресс, 1975. – 464 с.
8. Кохонен, Т. Ассоциативная память / Т. Кохонен. – М. : Мир, 1980. – 240 с.
9. Associative Processing and Processors / ed. A. Krikelis, C.C. Weems. – IEEE Computer Science Press, 1997. – 349 p.
10. Гаазе-Рапопорт, М.Г. От амебы до робота: модели поведения / М.Г. Гаазе-Рапопорт, Д.А. Поспелов. – М. : Наука, 1987. – 288 с.
11. Хайкин, С. Нейронные сети: полный курс / С. Хайкин. – 2-е изд. – М. : Изд. дом «Вильямс», 2006. – 1104 с.
12. Kosko, B. Bidirectional associative memories / B. Kosko // IEEE Transactions on Systems, Man, and Cybernetics. – 1988. – Vol. 18, № 1. – P. 49–60.
13. Уоссермен, Ф. Нейрокомпьютерная техника: теория и практика / Ф. Уоссермен. – М. : Мир, 1990. – 240 с.
14. Grossberg, S. Competitive learning: From interactive activation to adaptive resonance / S. Grossberg // Cognitive Science. – 1987. – Vol. 11. – P. 23–63.
15. Прокопович, Г.А. Адаптивный нейросетевой классификатор / Г.А. Прокопович // Информатика. – 2009. – № 3 (23). – С. 68–81.
16. Прокопович, Г.А. Нейросетевой блок памяти для адаптивной работы сложных технических систем в динамической среде / Г.А. Прокопович // Информатика. – 2010. – № 2 (26). – С. 54–65.

17. Прокопович, Г.А. Применение гетероассоциативных нейронных сетей для записи и восстановления информации / Г.А. Прокопович // Информатика. – 2012. – № 2 (34). – С. 38–49.
18. Прокопович, Г.А. Нейросетевая ассоциативная модель достижения цели / Г.А. Прокопович // Сборник тезисов XVIII Междунар. науч. конф. студентов, аспирантов и молодых ученых «Ломоносов-2011», секция «Вычислительная математика и кибернетика». – М. : МАКС Пресс, 2011. – С. 91–93.
19. Riley, D. 3D Puma Robot Demo // MATLAB Central [Электронный ресурс]. – 16 августа 2011. – Режим доступа : <http://www.mathworks.com/matlabcentral/fileexchange/14932-3d-puma-robot-demo>. – Дата доступа : 07.02.2012.
20. The capacity of the Hopfield associative memory / R.J. McEliece [et al.] // IEEE Transactions on Information Theory. – 1987. – Vol. 33, № 4. – P. 461–482.

Поступила 20.02.12

*Объединенный институт проблем  
информатики НАН Беларуси,  
Минск, Сурганова, 6  
e-mail: bigznich@tut.by*

**R.A. Prakapovich**

**NEURONET ASSOCIATIVE MEMORY MODEL  
FOR THE ROBOT MANIPULATOR CONTROL**

The control problem of dynamic systems possessing a large number of motion freedoms is considered on the example of six-member anthropomorphic manipulator UNIMATE PUMA-762. A model of the neural associative memory is proposed, which is based on recording hierarchical sequence of manipulator movements with the aid of the synaptic weights of a neural bi-directional associative memory network. A number of numerical experiments is conducted that prove the possibility of solving the problem using the proposed neural network model of memory.

УДК 004.934.2

**Д.В. Пекарь, С.Г. Тихоненко, А.Ф. Чернявский, В.С. Садов**

## **АЛГОРИТМ ОПТИМИЗАЦИИ ВХОДНЫХ ПРИЗНАКОВ И ПАРАМЕТРОВ КЛАССИФИКАТОРА ЭМОЦИОНАЛЬНОГО СОСТОЯНИЯ ЧЕЛОВЕКА ПО ЕГО РЕЧИ**

*Рассматривается алгоритм оптимизации входного вектора признаков и параметров классификатора эмоционального состояния человека, основанный на использовании стохастического метода роя частиц. Предлагаются пути его совершенствования с целью повышения скорости сходимости алгоритма. Доказывается улучшение точности классификации эмоционального состояния диктора по его речи при использовании оптимизированного вектора признаков и найденных параметров классификатора.*

### **Введение**

Значительная информационная составляющая, содержащаяся в эмоциональной окраске речи человека, представляет огромный интерес как для научного, так и для практического изучения. Возможность анализа и интерпретации невербальной информации позволяет создавать более совершенные речевые аналитические и экспертные системы, а также системы безопасности на основе использования речевого информационного потока. В представленной работе сравниваются возможности различных классификаторов эмоционального состояния человека и предлагается алгоритм оптимизации входного вектора признаков.

### **1. Признаки речевого сигнала и база данных речевых фрагментов**

Эмоциональное состояние говорящего человека определенным образом влияет на его голосовой тракт, что в конечном счете сказывается на физических параметрах произносимой им речи [1, 2]. Классификация эмоциональной окраски речи предполагает формирование вектора признаков, эффективного с вычислительной точки зрения и репрезентативного в плане дифференцирования различных типов эмоций.

Для вычисления различных характеристик речевого сигнала применялся инструментальный openEAR [3], разработанный Мюнхенским техническим университетом для изучения влияния эмоциональной составляющей на параметры речевого сигнала. Программный комплекс openEAR содержит предопределенные конфигурационные скрипты, согласно которым осуществляется вычисление тех или иных характеристик сигнала. В представленной работе применялись следующие конфигурационные скрипты:

Emo\_IS09 – вычисление 384 отдельных признаков [4];

Emo\_large – вычисление 6669 отдельных признаков [5].

Приведенный набор признаков описывает широкий спектр параметров речевого сигнала и, следовательно, отражает различные эмоциональные изменения в речевом тракте человека. Способность работы классификаторов с многомерными данными и зависимость времени классификации от размерности пространства признаков определяются длиной вектора признаков.

Речевой сигнал исследовался с помощью специализированной базы записей эмоциональной речи профессиональных актеров Берлинского университета [6]. База данных включает 535 речевых записей семи эмоциональных состояний 10 дикторов на немецком языке.

### **2. Оценка эффективности различных методов классификации эмоционального состояния человека по его речи**

Классификация эмоционального состояния диктора по его речи является нетривиальной задачей из-за многообразия человеческих голосов. Это связано со значительным разбросом параметров речевого сигнала и с невозможностью выделения четких кластеров их значений для каждого типа эмоций. Физиологические и филологические особенности произношения отдель-

ного человека также усложняют процесс классификации речевых сигналов. Дополнительная сложность процесса классификации обусловлена высокой размерностью вектора признаков и, следовательно, необходимостью использования значительных вычислительных ресурсов.

Целью данного исследования является анализ и сравнение работы различных классификаторов с многомерными векторами признаков и выбор оптимального метода для решения задачи классификации эмоциональной окраски речи. В качестве базового инструмента применялся программный комплекс WEKA [7], разработанный университетом Вайкато (Новая Зеландия) для обработки, классификации, кластеризации и селекции данных. Инструментарий обладает удобным интерфейсом и широким набором методов для осуществления всестороннего анализа данных. Эффективность классификации анализировалась при использовании следующих методов:

- наивного байесовского классификатора (НБК) [8];
- метода опорных векторов (МОВ) [9];
- дерева принятых решений (ДПР) [10];
- К-ближайших соседей (КБС) [11].

При испытании методов классификации для каждого диктора и набора признаков определялись следующие параметры:

- средняя точность классификации  $P$ :

$$P = \frac{1}{N_E} \sum_{i=1}^{N_E} \frac{|M_i \cap C_i|}{|C_i|}, \quad (1)$$

где  $N_E$  – число исследуемых типов эмоций (в представленном исследовании  $N_E = 7$ );  $M_i$  – множество записей, принадлежащих к  $i$ -й эмоции;  $C_i$  – множество записей, отнесенных к  $i$ -й эмоции;  $|M_i|$  – мощность множества, определяющая число элементов в данном множестве;

- средняя полнота классификации  $R$ :

$$R = \frac{1}{N_E} \sum_{i=1}^{N_E} \frac{|M_i \cap C_i|}{|M_i|}; \quad (2)$$

- среднее значение  $F$ -оценки:

$$F = \frac{1}{\alpha \frac{1}{P} + (1 - \alpha) \frac{1}{R}} \quad (3)$$

при  $\alpha = 0,5$ .

Данные табл. 1 показывают, что наиболее эффективным из протестированных является классификатор на основе метода опорных векторов. Дополнительным преимуществом этого метода является устойчивость к масштабируемости пространства признаков. Принимая во внимание тот факт, что исследуемые методы классификации имеют входные настроечные параметры, которые влияют на конечный результат, а также значительный размер вектора признаков, можно предположить, что входной вектор можно оптимизировать удалением малоинформативных признаков, а классификатор адаптировать к входным данным и используемому вектору признаков.

Таблица 1

Результаты испытаний различных методов для классификации эмоциональной окраски диктора

Набор признаков	Метод	Средняя точность классификации, $P$	Средняя полнота классификации, $R$	$F$ -оценка
IS09	НБК	0,57	0,54	0,49
	МОВ	<b>0,78</b>	<b>0,76</b>	<b>0,76</b>
	ДПР	0,51	0,50	0,48
	КБС	0,64	0,55	0,54
Emo_large	НБК	0,60	0,57	0,52
	МОВ	<b>0,77</b>	<b>0,76</b>	<b>0,75</b>
	ДПР	0,50	0,45	0,43
	КБС	0,72	0,64	0,64

### 3. Стохастический алгоритм оптимизации вектора признаков и параметров классификатора эмоциональной окраски речи

В ранее упомянутых работах предложены подходы к оптимизации входных данных, фильтрации малоинформативных признаков и их ранжированию для достижения наибольшей эффективности классификации. Предприняты попытки поиска единого оптимального метода для решения задачи классификации эмоциональной окраски речи. Методы и подходы предшествующих исследователей имеют значительный недостаток, состоящий в том, что оптимизация вектора признаков, селекция его атрибутов и настройка параметров классификатора происходят отдельно друг от друга. В предлагаемом алгоритме одновременно осуществляются оптимизация вектора признаков и оптимизация параметров классификатора. Входные данные и классификатор рассматриваются как взаимосвязанные составляющие алгоритма.

Не существует единого аналитического подхода для решения задачи оптимизации вектора признаков и параметров классификатора. Подобные задачи выбора из заданного множества параметров наилучшей их комбинации или нахождения оптимального значения параметра относятся к классу комбинаторно-оптимизационных задач, для решения которых применяются эволюционные алгоритмы.

Разработанные алгоритмы отличаются принципом поиска нового решения, способом кодирования решения, скоростью сходимости и т. д. В предложенном алгоритме используется метод роя частиц, принцип которого основан на имитации поведения квантовых частиц. Данный тип эволюционного алгоритма был предложен Е. Кеннеди для моделирования поведения птиц в стае [12]. Дальнейшим развитием данного подхода явилось моделирование поведения частиц, подчиняющегося законам квантовой механики [13]. Важным преимуществом нового алгоритма является более быстрая сходимость относительно других эволюционных алгоритмов [14], что позволяет находить решение за меньшее число итераций.

Основным положением метода роя частиц является движение множества частиц в многомерном пространстве возможных решений, в котором каждое измерение соответствует отдельному оптимизируемому признаку. Каждая частица представлена точкой в многомерном пространстве  $X_i = (x_{i1}, x_{i2}, x_{i3}, \dots, x_{iD-1}, x_{iD})$ , которая соответствует текущему решению, найденному  $i$ -й частицей, где  $D$  – размерность пространства признаков. С каждой частицей роя  $X_i$  ассоциируется текущее положение  $P_i^{loc}(t)$  и  $P_i^{Gloc}(t)$  – наилучшее положение, достигнутое до текущей итерации  $t$ . С роем ассоциируется наилучшее достигнутое положение  $P^G(t)$  до текущей итерации  $t$  среди всех частиц. Положения частицы обновляются согласно следующим выражениям, которые учитывают имитацию поведения квантовых частиц [15]:

$$P_{ij}^{loc}(t+1) = p_{ij}(t) + \beta \cdot |m_j(t) - P_{ij}^{loc}(t)| \cdot \ln\left(\frac{1}{u}\right) \quad \text{при } k \leq 0,5; \quad (4)$$

$$P_{ij}^{loc}(t+1) = p_{ij}(t) - \beta \cdot |m_j(t) - P_{ij}^{loc}(t)| \cdot \ln\left(\frac{1}{u}\right) \quad \text{при } k > 0,5, \quad (5)$$

где  $u$ ,  $k$  – случайные величины, равномерно распределенные в интервале  $(0;1)$ ;  $\beta$  – коэффициент, определяющий величину изменения положения частицы;  $p_{ij}(t)$  – координаты аттрактора частицы, которые определяются выражением

$$p_{ij}(t) = (c_1 P_{ij}^{loc}(t) + c_2 P_j^G(t)) / (c_1 + c_2). \quad (6)$$

Здесь  $c_1$ ,  $c_2$  – параметры, определяющие величину связи отдельной частицы с остальными частицами роя.

Функция  $m_j(t)$  представляет собой  $j$ -ю усредненную координату, вычисляемую по наилучшим положениям для каждой частицы роя согласно следующему выражению:

$$m(t) = (m_1(t) \dots m_j(t) \dots m_D(t)) = \left( \frac{1}{S} \sum_{i=1}^S \alpha_i P_{i1}^{Gloc}(t) \dots \frac{1}{S} \sum_{i=1}^S \alpha_i P_{ij}^{Gloc}(t) \dots \frac{1}{S} \sum_{i=1}^S \alpha_i P_{iD}^{Gloc}(t) \right), \quad (7)$$

где  $S$  – количество частиц в рое;  $\alpha_i$  – вес отдельной частицы, который определяется из правила «чем лучше решение, найденное частицей, тем больший вес она имеет». Описанный подход позволяет улучшить сходимость метода роя частиц.

Найденное решение оценивалось точностью классификации при использовании вычисленных параметров классификатора и оптимизированного вектора признаков. Для получения надежных оценок работы классификатора использовался метод *K-fold cross-validation* при значении  $K = 3$ . Суть данного подхода заключается в том, что входная обучающая выборка данных разбивается на три непересекающиеся части случайным образом. Две части используются для обучения классификатора, а по третьей части оценивается точность  $P$ , полнота  $R$  и  $F$ -оценки классификации эмоциональной окраски речи. Описанная процедура повторяется три раза, при этом каждая часть выборки данных используется как для обучения, так и для тестирования классификатора. Подобная техника исключает влияние разбивки данных на конечный результат, обеспечивая эффективное использование данных при обучении и получение более надежных оценок, поскольку результирующее значение является усреднением нескольких испытаний классификатора.

Новшеством, привнесенным в существующий метод и позволяющим ускорить его сходимость, а также повысить поисковую способность алгоритма, является перераспределение частиц в рое согласно алгоритму, изображенному на рис. 1.

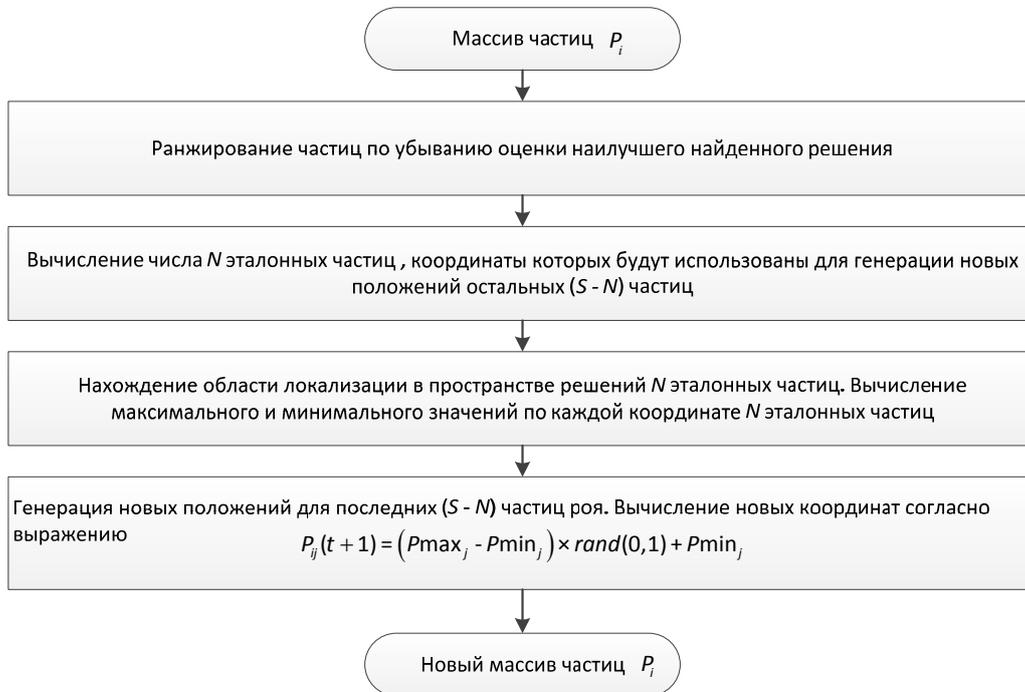


Рис. 1. Алгоритм перераспределения частиц в рое

Число эталонных частиц  $N$ , которые не изменят своего положения в пространстве решений при перераспределении частиц, рассчитывалось следующим образом:

$$N = \left\lceil S \cdot \left( 1 - \frac{\lg\left(\frac{t \cdot 1000}{MI} \cdot 0,9\right)}{3} \right) \right\rceil, \quad (8)$$

где  $MI$  – максимальное число итераций;  $t$  – текущая итерация алгоритма. Предложенная функция вычисления числа эталонных частиц  $N$ , которые не изменят своего положения в пространстве решений, на начальных итерациях алгоритма не ограничивает пространство поиска (значение  $N$  составляет более половины числа всех частиц), а на завершающих итерациях позволяет более детально исследовать предполагаемое подпространство решений большим коли-

чеством частиц. На рис. 2 изображен вид дробного множителя при параметре  $S$  в функции (8) для числа итераций  $MI = 200$ .

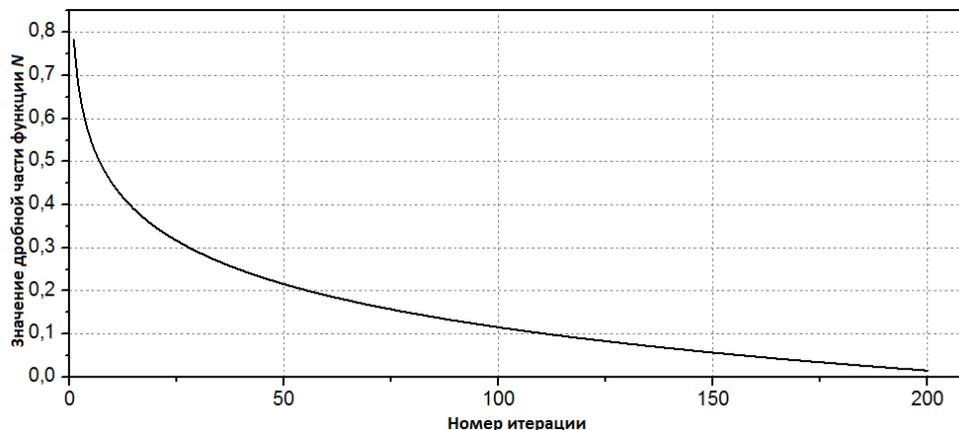


Рис. 2. График функции для вычисления числа эталонных частиц в рое

Для сравнения эффективности существующего алгоритма с его версией, включающей предложенный алгоритм перераспределения частиц в рое, было проведено тестирование, которое заключалось в сравнении точности классификации при использовании найденных решений, оптимизированных векторов признаков и параметров классификатора. Было проведено 10 тестовых испытаний, в течение которых фиксировались значения точностей оценок найденного решения на каждой итерации алгоритма (рис. 3).

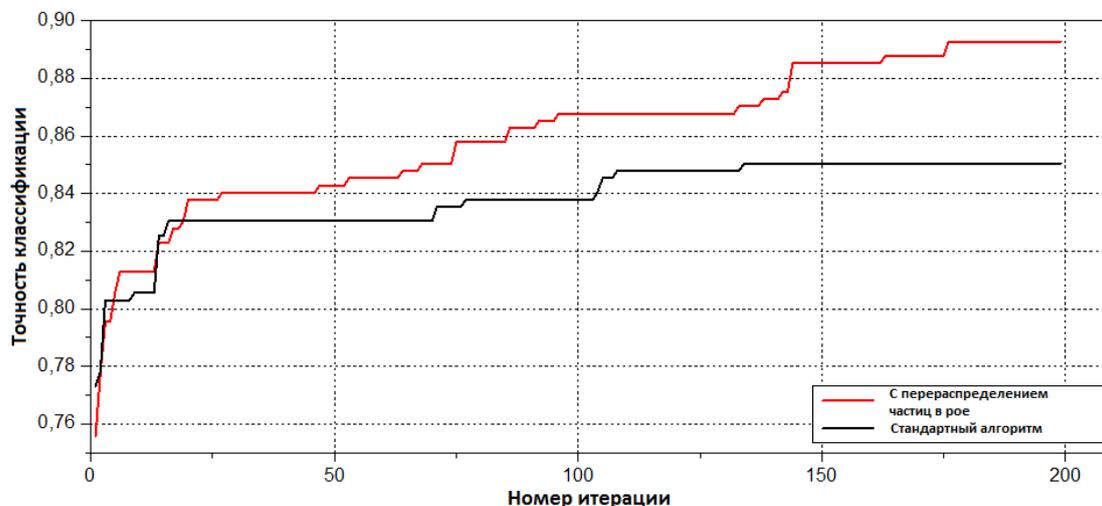


Рис. 3. Значения оценки найденного решения в зависимости от итерации алгоритма

Важным этапом при реализации метода роя частиц является кодирование решения или задание соответствия между положением частицы и найденным решением. Поскольку в предложенном алгоритме происходит одновременная оптимизация вектора признаков и параметров классификатора, размерность пространства решений будет определяться следующим выражением:

$$D = N_F + N_P, \quad (9)$$

где  $N_F$  – число признаков речевого сигнала;  $N_P$  – число оптимизируемых параметров классификатора. Без потери общности полагается, что первые  $N_F$  координат частицы соответствуют  $N_F$  признакам речевого сигнала, а последние  $N_P$  координат вектора соответствуют оптимизируемым параметрам классификатора. В предложенном алгоритме  $N_F = 384$  для набора признаков IS09 и  $N_F = 6669$  для набора признаков Emo\_large,  $N_P = 2$  для обоих случаев. Поскольку

каждая координата является вещественным числом и может принимать любое значение из заданного диапазона, а каждый признак речевого сигнала может как присутствовать (что соответствует значению координаты, равному 1), так и отсутствовать (значение соответствующей координаты равно -1) в векторе найденного решения, необходимо провести дискретизацию значений первых  $N_F$  координат для оценки найденного решения. Для решения задачи декодирования значения координаты применялось следующее правило:

$$\begin{cases} 1, & \text{если } d_j \geq 0 \text{ при } j = \overline{1 \dots N_F}; \\ -1, & \text{если } d_j < 0 \text{ при } j = \overline{1 \dots N_F}, \end{cases} \quad (10)$$

где  $d_j$  –  $j$ -я координата пространства решений. Рассмотренные этапы реализации алгоритма составляют единый стохастический алгоритм оптимизации входного вектора признаков и параметров классификатора (рис. 4).

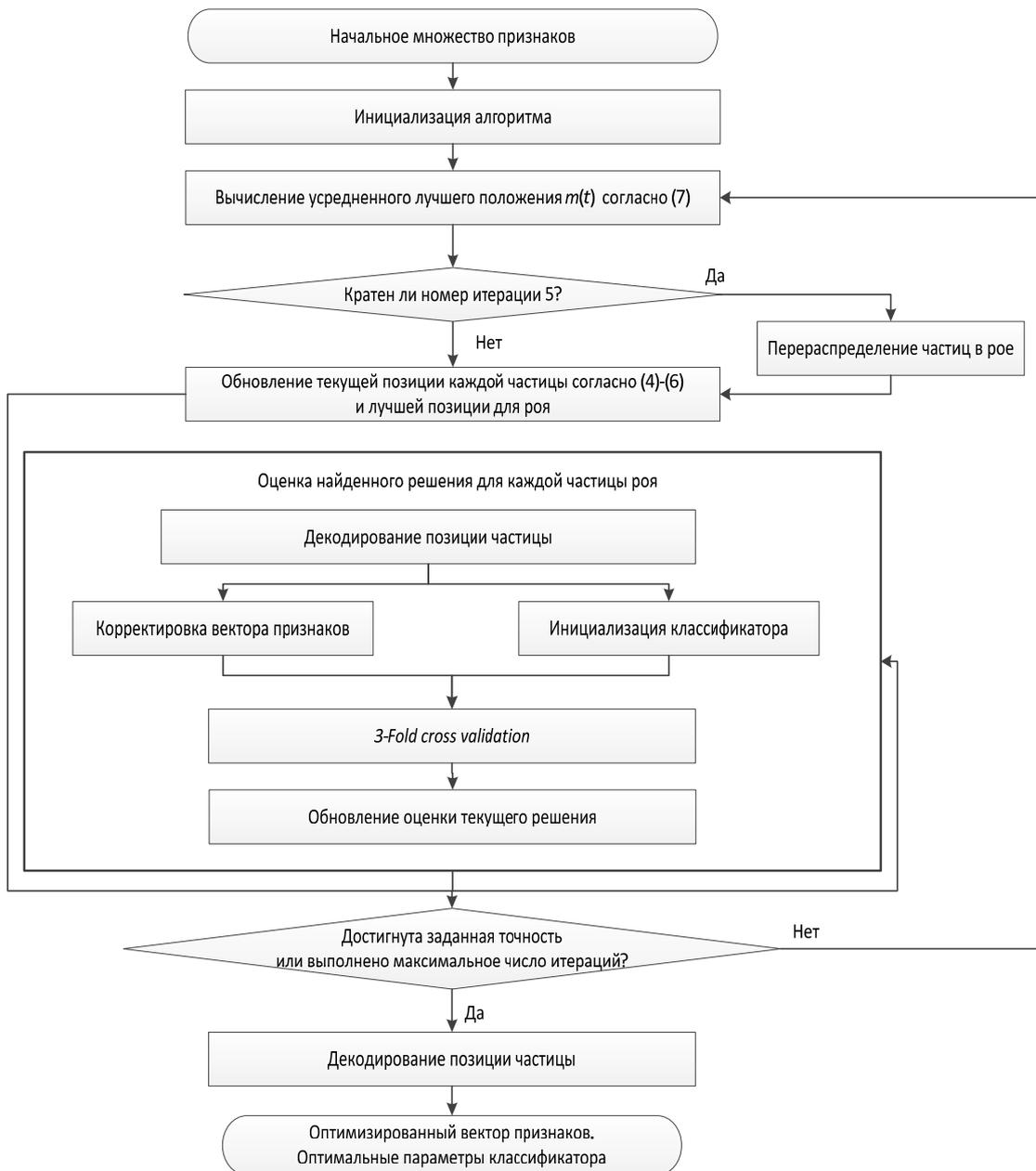


Рис. 4. Обобщенная схема алгоритма оптимизации вектора признаков и параметров классификатора

#### 4. Результаты тестирования

Практическое тестирование имело целью апробацию предложенного алгоритма оптимизации входного вектора признаков и параметров классификатора эмоционального состояния человека. Главными исследуемыми параметрами являлись размер найденного вектора признаков, а также точность классификации эмоциональной окраски речи при использовании оптимизированного вектора признаков относительно начального набора признаков и параметров классификатора. Работоспособность алгоритма оценивалась по следующей методике. Записи из описанной выше базы данных делились случайным образом на две непересекающиеся части в соотношении 3:1 для каждого диктора. Первая часть данных, соответственно 75 % от общего числа записей, использовалась для оптимизации вектора признаков и параметров классификатора с помощью представленного выше алгоритма. Вторая часть данных, соответственно 25 % от общего числа записей, использовалась для оценки параметров работы классификатора с новым оптимизированным вектором признаков. В качестве классификатора использовался МОВ с линейным ядром [9]. Данный тип МОВ был выбран ввиду значительной размерности вектора признаков, который позволяет осуществлять задачу классификации при меньших временных затратах и требованиях к аппаратным ресурсам. Начальными параметрами алгоритма во всех испытаниях являлись исходные наборы признаков, соответственно включающих в себя 384 и 6669 признака, и параметры по умолчанию для классификатора, заданные в программном комплексе WEKA. Значение параметра  $\beta$ , найденное экспериментальным путем, линейно уменьшалось от значения 0,8 до 0,2.

В табл. 2 и 3 сравниваются результаты работы оптимизированного классификатора с найденным вектором признаков и его работы с начальными значениями параметров и полно-размерным вектором признаков.

Таблица 2

Результаты тестирования алгоритма для набора признаков IS09

Номер диктора	Начальная длина вектора признаков	Точность классификации с начальными параметрами, %	Длина оптимизированного вектора признаков	Точность классификации с найденными параметрами, %	Изменение точности классификации, %
1	386	76,92	220	69,23	-7,69
2	386	100,00	188	86,66	-13,34
3	386	81,82	191	72,73	-9,09
4	386	60,00	185	70,00	10,00
5	386	71,43	198	57,14	-14,29
6	386	66,67	192	55,56	-11,11
7	386	75,00	193	68,75	-6,25
8	386	94,44	202	77,78	-16,66
9	386	71,43	187	78,57	7,14
10	386	83,33	206	77,78	-5,55
<b>Среднее</b>	<b>386</b>	<b>78,10</b>	<b>196</b>	<b>71,42</b>	<b>-6,68</b>

Таблица 3

Результаты тестирования алгоритма для набора признаков Emo\_large

Номер диктора	Начальная длина вектора признаков	Точность классификации с начальными параметрами, %	Длина оптимизированного вектора признаков	Точность классификации с найденными параметрами, %	Изменение точности классификации, %
1	6671	69,23	3334	76,92	7,69
2	6671	100,00	3282	93,33	-6,67
3	6671	90,90	3318	72,72	-18,18
4	6671	70,00	3291	70,00	0,00
5	6671	64,28	3356	64,28	0,00
6	6671	55,55	3285	66,66	11,11
7	6671	75,00	3334	75,00	0,00
8	6671	88,88	3341	94,44	5,56
9	6671	85,71	3377	92,86	7,15
10	6671	77,77	3392	83,33	5,56
<b>Среднее</b>	<b>6671</b>	<b>77,73</b>	<b>3331</b>	<b>78,95</b>	<b>1,22</b>

Следующим этапом исследования было тестирование алгоритма в дикторонезависимом режиме классификации эмоций (табл. 4), для чего записи всех дикторов объединялись в одно множество, которое затем разделялось случайным образом в соотношении 3:1 на обучающую и тестовую выборку соответственно. Остальные параметры алгоритма оставались прежними, как и при предыдущем тестировании.

Таблица 4

Результаты тестирования алгоритма в дикторонезависимом режиме

Набор признаков	Начальная длина вектора признаков	Точность классификации с начальными параметрами, %	Длина оптимизированного вектора признаков	Точность классификации с найденными параметрами, %	Изменение точности классификации, %
IS09	386	76,12	186	79,10	2,98
Emo large	6671	86,57	3337	88,06	1,49

### Заключение

Полученные результаты показали работоспособность алгоритма. Преимущества предложенного алгоритма проявляются при применении вектора признаков со значительным начальным числом признаков порядка нескольких тысяч. Увеличение обучающей выборки также положительно влияет на точность классификации. Стоит отметить такое важное свойство предложенного алгоритма, как выделение наиболее значимых и информативных признаков речевого сигнала для классификации эмоциональной окраски речи человека.

### Список литературы

1. Williams, C.E. Emotions and Speech: Some Acoustical Correlates / C.E. Williams // Journal of the Acoustical Society of America. – 1972. – Vol. 52, № 4. – P. 1238–1250.
2. Banse, R. Acoustic profiles in vocal emotion expression / R. Banse // Journal of personality and social psychology. – 1996. – Vol. 70, № 3. – P. 614–636.
3. Eyben, F. OpenEAR – Introducing the Munich Open-Source Emotion and Affect / F. Eyben // Proc. ACII. – Amsterdam, Netherlands, 2009. – P. 576–581.
4. Schuller, B. The interspeech 2009 emotion / B. Schuller // Interspeech. – Brighton, UK, 2009.
5. Eyben, F. OpenSMILE : the Munich open Speech and Music Interpretation by Large Space Extraction toolkit / F. Eyben, M. Woellmer, B. Schuller. – 2010. – Mode of access : [http://sourceforge.net/projects/opensmile/files/openSMILE\\_book\\_1.0.1.pdf](http://sourceforge.net/projects/opensmile/files/openSMILE_book_1.0.1.pdf). – Date of access : 20.01.2012.
6. Berlin Database of Emotional Speech [Electronic resource]. – Technical University Berlin, 1999. – Mode of access : <http://pascal.kgw.tu-berlin.de/emodb/index-1024.html>. – Date of access : 20.01.2012.
7. Holmes, G. WEKA: a machine learning / G. Holmes // Intelligent Information Systems. – 1994. – № 29. – P. 357–361.
8. John G.H. Estimating Continuous Distributions in Bayesian Classifiers / G.H. John // Proc. of the Eleventh Conf. on Uncertainty in Artificial Intelligence. – San Francisco, USA, 1995. – Vol. 1. – P. 338–345.
9. LIBLINEAR – A Library for Large Linear Classification // Rong-En Fan [et al.]. – 2008. – Mode of access : <http://www.csie.ntu.edu.tw/~cjlin/liblinear/>. – Date of access : 20.01.2012.
10. Quinlan, R.J. C4.5: Programs for Machine Learning / R.J. Quinlan // Machine Learning. – 2004. – Vol. 16, № 3. – P. 235–240.
11. Aha, D.W. Instance-Based Learning Algorithms / D.W. Aha // Machine Learning. – Hingham, USA, 1991. – Vol. 6, № 1. – P. 37–66.
12. Kennedy, J. Particle Swarm Optimization / J. Kennedy // Proc. of IEEE Intern. Conf. on Neural Networks. – Washington, USA, 1995. – Vol. 4. – P. 1942–1948.
13. Sun, J. Particle Swarm Optimization with particles having Quantum Behavior / J. Sun // Proc. of Congress on Evolutionary Computation. – Portland, USA, 2004. – P. 325–331.

14. Gao, H. A Simple Quantum-inspired Particle swarm Optimization and its Application / H. Gao // Information Technology Journal. – 2011. – № 10. – P. 2315–2321.

15. Maolong, X. An improved quantum-behaved particle swarm optimization algorithm with weighted mean best position / X. Maolong // Applied Mathematics and Computation. – 2008. – № 205. – P. 751–759.

**Поступила 07.07.12**

*Белорусский государственный университет,  
Минск, Курчатова, 5  
e-mail: pekar.dzmitry@gmail.com*

**D.V. Pekar, S.G. Tsikhanenka, A.F. Chernyavskii, V.S. Sadov**

**ALGORITHM FOR OPTIMIZATION OF FEATURES  
AND PARAMETERS OF SPEAKER'S EMOTIONAL STATE CLASSIFIER**

The paper presents a particle swarm based algorithm for optimization of feature vector and parameters of speaker's emotional state classifier. To speed up the convergence, an enhancement of existing quantum inspired particle swarm optimization algorithm is proposed. The algorithm shows better quality when using optimized feature vector and new classifier's parameters.

УДК 519.71

О.В. Шут

**МЕТОД РЕШЕНИЯ ЗАДАЧ РАСПОЗНАВАНИЯ В БУЛЕВОМ ПРОСТРАНСТВЕ  
НА ОСНОВЕ ЛОГИЧЕСКОЙ И ПРЕЦЕДЕНТНОЙ МОДЕЛЕЙ**

*Рассматриваются дедуктивный и индуктивный подходы к решению задач распознавания, в которых используются соответственно логическая и прецедентная модели представления начальной информации. Строится алгебра объектов и показывается ее изоморфность булевой алгебре. Разрабатываются алгоритмы перехода между моделями представления информации. Предлагается модификация метода резолюций для прецедентной модели. Разрабатывается комбинированный алгоритм, объединяющий метод резолюций с семейством алгоритмов распознавания, и показывается, что он работает не хуже любого из алгоритмов, образующих комбинацию.*

**Введение**

Теория распознавания как наука прошла в процессе эволюции два больших этапа. На первом этапе рассматривались прикладные вопросы, которые в постановочном смысле сводились к задачам распознавания [1]. Большинство применений теории распознавания связано с областями науки, плохо поддающимися формализации: медициной, химией, социологией и т. д. В этих областях трудно строить формальные теории, поэтому на первом этапе использовались эвристические алгоритмы. В основу их обоснованности был положен принцип сходимости на множестве объектов, информация о которых априори известна.

Второй этап развития теории распознавания характеризуется переходом от отдельных алгоритмов к рассмотрению моделей решения прикладных задач. На этом этапе эвристикой являлся уже не выбор алгоритма, а выбор принципа, в соответствии с которым алгоритмы могли быть построены стандартным образом. На базе эвристических алгоритмов выстраивались схемы, которые позволяли нивелировать недостатки эвристических алгоритмов. К числу таких схем относятся схемы логической корректировки, алгебраические модели, в основу обоснованности которых положены необходимые условия правильности в виде, например, корректности алгоритмов [2].

В настоящее время можно говорить о третьем этапе развития теории распознавания. Суть его заключается в следующем. Существуют классические алгоритмы, вопрос об обосновании которых давно изучен и не вызывает сомнений. В качестве примера можно назвать метод резолюций, который имеет дедуктивную природу. Задача распознавания может быть сформулирована в двух постановках, которым соответствуют два подхода к решению: дедуктивный (метод резолюций) и индуктивный (алгоритмы распознавания). На третьем этапе разрабатываются алгоритмы, сочетающие преимущества обоих подходов.

Традиционно в каждой из двух постановок задачи распознавания используются свои модели представления начальной информации: логическая для дедуктивного подхода и прецедентная для индуктивного. В данной работе рассматриваются обе эти модели для случая, когда все признаки объектов могут принимать два значения. Показана эквивалентность этих моделей при определенных ограничениях, что дает возможность использовать метод резолюций для решения задач распознавания. На основе метода резолюций и параметрического семейства алгоритмов распознавания построен комбинированный алгоритм, работающий не хуже любого из двух исходных алгоритмов.

**1. Постановка задачи**

Рассмотрим следующую задачу  $Z$  в общей постановке, сформулированной в [3]:

*Пусть на конечном множестве объектов  $X$  произвольной природы задано конечное число подмножеств (классов)  $X_1, \dots, X_l$ . Имеется начальная информация  $I_0$  о принадлежности к классам множества объектов  $X^0 \subset X$ . Требуется указать алгоритм, определенный на всем*

множестве  $X$ , который на основании информации  $I_0$  для произвольного объекта  $x \in X$  вычисляет принадлежность  $x$  к классам  $X_1, \dots, X_l$ .

Конкретные варианты постановки данной задачи определяются способами представления множеств  $X$  и  $I_0$ , числом классов  $l$  и т. д. В данной работе рассматриваются два варианта:

1. Задача  $Z_1$ . Информация  $I_0$  задана логическим способом: через предикаты (правила, логические формулы), которые одновременно используются как для описания объектов, так и для описания функции принадлежности классам. Информация о принадлежности объектов, удовлетворяющих правилам, считается заданной. Необходимо для указанного объекта  $x$  определить, выводим ли он из правил, описывающих класс  $X_i$ ,  $i = \overline{1, l}$ .

В качестве примера задачи  $Z_1$  можно привести задачу классификации формул в исчислении высказываний. Такие задачи решаются дедуктивными методами. Общепринятым методом решения задач, сформулированных в постановке задачи  $Z_1$ , является метод резолюций [4].

2. Задача  $Z_2$ . Информация  $I_0$  задана прецедентным способом: для каждого из классов  $X_1, \dots, X_l$  явно указаны объекты, принадлежащие этому классу (т. е. для каждого объекта задан его информационный вектор [1]). Требуется указать алгоритм, который на основании информации  $I_0$  для произвольного объекта  $x \in X$  вычисляет принадлежность  $x$  к классам  $X_1, \dots, X_l$ .

Такая постановка типична для задач распознавания образов с обучением. Для решения задачи  $Z_2$  разработано огромное количество алгоритмов.

Сравнивая задачи  $Z_1$  и  $Z_2$ , можно заметить, что задача  $Z_1$  носит чисто теоретический характер, а обоснованность метода ее решения не вызывает сомнений; задача  $Z_2$ , в свою очередь, имеет прикладной характер. Однако поскольку задачи  $Z_1$  и  $Z_2$  являются вариантами постановки одной и той же задачи  $Z$ , их решения должны быть сравнимы. Поэтому поставим задачу  $Z_0$  исследования связи между решениями задач  $Z_1$  и  $Z_2$ . Именно задача  $Z_0$  и рассматривается в данной работе.

На рис. 1 показана связь всех вышеупомянутых задач. Здесь через  $I_0^1$  и  $I_0^2$  обозначены исходные данные (начальная информация  $I_0$ ), через  $Out_1$  и  $Out_2$  – результаты решения задач  $Z_1$  и  $Z_2$  соответственно, через  $A_1$  и  $A_2$  – алгоритмы решения задач: для задачи  $Z_1$  алгоритм  $A_1$  – это метод резолюций; для задачи  $Z_2$  под алгоритмом  $A_2$  может подразумеваться любой алгоритм, применяемый для решения задач распознавания. Таким образом, на рис. 1 видно, что задача  $Z_0$  состоит в исследовании связи между  $Out_1$  и  $Out_2$ . Разобьем эту задачу на следующие подзадачи:

$Z_{01}$  – установление связи между моделями представления начальной информации  $I_0^1$  и  $I_0^2$  и построение алгоритмов перехода между ними;

$Z_{02}$  – сравнение алгоритмов  $A_1$  и  $A_2$  на множестве  $X$ ;

$Z_{03}$  – построение комбинированного алгоритма, объединяющего алгоритмы  $A_1$  и  $A_2$  и работающего не хуже любого из этих двух алгоритмов на множестве  $X$ .

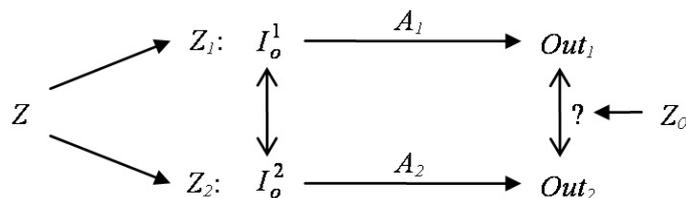


Рис. 1. Схема связи задач  $Z$ ,  $Z_1$ ,  $Z_2$ ,  $Z_0$

Рассмотрим подзадачи по порядку. Для решения подзадачи  $Z_{01}$  введем алгебру объектов, описанную далее, и покажем, что между прецедентной и логической моделями представления информации можно установить взаимно-однозначное соответствие.

## 2. Алгебра объектов

Покажем, что прецедентная и логическая модели представления информации являются эквивалентными в следующем смысле: при определенных ограничениях на число объектов и значения их признаков любая начальная информация может быть представлена с применением любой из этих двух моделей. Для этого рассмотрим алгебру объектов, которая используется в случае прецедентного представления информации, и покажем ее изоморфность булевой алгебре, используемой в логическом подходе.

### 2.1. Основные определения

Воспользуемся моделью представления объектов, предложенной в [5]. Пусть объект имеет конечное число признаков. Обозначим через  $S = \{s_1, s_2, \dots, s_n\}$  множество всех признаков, используемых в предметной области, в которой рассматривается задача  $Z_1$  или  $Z_2$ , и зафиксируем порядок признаков. Для всех признаков будем предполагать, что если значение признака известно, то оно принадлежит множеству  $D = \{0, 1\}$ . Возможна также ситуация, когда значение признака неизвестно. Введем специальный символ для неизвестного значения, например «?», и обозначим расширенное таким образом множество значений признаков через  $\bar{D} = \{0, 1, ?\}$ .

Определение 1. *Объектом* называется отображение вида  $p: s_1 \times s_2 \times \dots \times s_n \rightarrow \bar{D}^n$ , где  $\bar{D}^n = \underbrace{\bar{D} \times \dots \times \bar{D}}_n$ .

Объект  $p$ , обладающий признаками  $s_j \in S$ ,  $j = \overline{1, n}$ , которые соответственно имеют значения  $d_j^p \in \bar{D}$ , запишем как  $p(s_1, s_2, \dots, s_n) = (d_1^p, d_2^p, \dots, d_n^p)$ .

Определение 2. Объекты  $p$  и  $q$  называются *равными*, если  $\forall j d_j^p = d_j^q$ .

Иначе говоря, объекты равны, если значения соответствующих признаков этих объектов совпадают.

Определение 3. *Набором объектов* называется любое множество объектов из предметной области, рассматриваемой в задаче  $Z_2$ .

Набор  $P$ , состоящий из объектов  $p_1, p_2, \dots, p_r$ , записывается как  $P = \{p_1, p_2, \dots, p_r\}$ .

Определение 4. *Объектом-признаком* называется объект, для которого известно значение только одного признака из  $n$ .

Определение 5. Объект называется *нормализованным*, если известны значения всех его признаков.

Определение 6. Набор называется *нормализованным*, если все содержащиеся в нем объекты нормализованы.

Рассмотрим произвольный объект  $p$ , у которого значение признака  $s_j$  неизвестно. Если нет дополнительных ограничений на неизвестное значение признака, этот признак может принимать любое значение из  $D$ . Поэтому в данной работе используется следующая интерпретация: объект  $p$ , значение признака  $s_j$  которого неизвестно, рассматривается как набор из двух объектов, у одного из которых значение этого признака равно 0, а у другого 1; значения же всех остальных признаков совпадают со значениями соответствующих признаков  $p$ :

$$p = (\dots, ?^j, \dots) = \{(\dots, 0^j, \dots), (\dots, 1^j, \dots)\}.$$

Опишем процедуру нормализации произвольного набора  $P$ .

Алгоритм нормализации *Norm* :

*Шаг 1.* Выберем объект  $p \in P$  и признак  $s_j$  такие, что  $d_j^p = ?$ . Если такие объект и признак выбрать невозможно, то переходим к шагу 4.

*Шаг 2.* Построим объекты  $p'$  и  $p''$  по следующим правилам:

$$d_i^{p'} = \begin{cases} 0, & i = j; \\ d_i^p, & i \neq j; \end{cases} \quad d_i^{p''} = \begin{cases} 1, & i = j; \\ d_i^p, & i \neq j. \end{cases}$$

*Шаг 3.* Заменяем в наборе  $P$  объект  $p$  на два объекта:  $P := (P \setminus \{p\}) \cup \{p', p''\}$ . Переходим к шагу 1.

*Шаг 4.* Алгоритм завершает работу.

Если необходимо нормализовать не набор, а отдельный объект, то на вход алгоритма *Norm* подается набор, состоящий из одного этого объекта.

Исследуем свойства алгоритма нормализации.

*Свойство 1.* Если набор  $P$  нормализован, то  $Norm(P) = P$ .

*Доказательство.* Если набор  $P$  нормализован, то после шага 1 алгоритма *Norm* произойдет переход на шаг 4, т. е. в результате нормализации набор  $P$  не будет изменен. ■

*Свойство 2.*  $Norm(Norm(P)) = Norm(P)$ .

*Доказательство свойства 2* следует из свойства 1 и того факта, что набор  $Norm(P)$  всегда является нормализованным. ■

*Свойство 3.*  $Norm(P) \setminus Norm(Q) = Norm(P \setminus Q)$ .

*Свойство 4.*  $Norm(P) \cap Norm(Q) = Norm(P \cap Q)$ .

*Свойство 5.*  $Norm(P) \cup Norm(Q) = Norm(P \cup Q)$ .

*Доказательство свойств 3–5* следует из того, что процедура нормализации, применяемая к произвольному объекту, не затрагивает другие объекты. ■

Заметим, что в процессе нормализации может возникнуть ситуация, когда новый объект совпадает с существующим. Для однозначности изложения будем считать, что эти новые объекты не включаются в набор, т. е. набор не может содержать одинаковые объекты. При необходимости будем ограничиваться рассмотрением только нормализованных наборов и объектов.

## 2.2. Основные операции алгебры объектов

Рассмотрим основные операции над объектами и их наборами. Пусть заданы наборы  $P$  и  $Q$ . Обозначим через  $0_n$  пустой набор, не содержащий ни одного объекта, а через  $1_n$  – набор всех возможных объектов, обладающих  $n$  признаками из множества  $S$ .

Введем следующие операции:

1. *Отрицание.*

Отрицание  $\bar{P}$  набора  $P$  определяется следующим образом:

$$\bar{P} = 1_n \setminus P.$$

2. *Умножение.*

Объекты  $p$  и  $q$  назовем *совместимыми*, если для любого признака  $s_j$  выполняется хотя бы одно из следующих условий:

а)  $d_j^p = d_j^q$ ;

б)  $d_j^p = ? \vee d_j^q = ?$ .

Произведение  $p \wedge q$  (или  $pq$ ) совместимых объектов  $p$  и  $q$  есть объект, значения признаков которого определяются следующим образом:

$$d_j^{pq} = \begin{cases} d_j^q, & d_j^q \neq ?; \\ d_j^p, & d_j^p \neq ?; \\ ?, & d_j^p = d_j^q = ?. \end{cases}$$

Операция умножения определена только для совместимых объектов.

Произведение  $P \wedge Q$  (или  $PQ$ ) наборов  $P$  и  $Q$  есть набор, состоящий из всевозможных произведений совместимых объектов, один из которых входит в  $P$ , а другой в  $Q$ :

$$P \wedge Q = \bigcup_{p \in P, q \in Q} \{pq\}.$$

3. Сложение.

Сумма  $P \vee Q$  наборов  $P$  и  $Q$  определяется по следующему правилу:

$$P \vee Q = (PQ) \cup (\overline{PQ}) \cup (P\overline{Q}).$$

В работе [5] показано, что любой набор может быть представлен в виде суммы объектов, входящих в этот набор, а любой объект, в свою очередь, может быть представлен в виде произведения его объектов-признаков:

$$P = \{p_1, \dots, p_r\} = \bigvee_{i=1}^r p_i = \bigvee_{i=1}^r \bigwedge_{j=1}^n (? , \dots, d_j^{p_i}, \dots, ?).$$

Для краткости обозначим объекты-признаки через  $d_j^p = (? , \dots, d_j^p, \dots, ?)$ . Тогда

$$P = \{p_1, \dots, p_r\} = \bigvee_{i=1}^r p_i = \bigvee_{i=1}^r \bigwedge_{j=1}^n d_j^{p_i}.$$

Таким образом, построена алгебра объектов  $G_n = \langle P_n, \{\neg, \wedge, \vee\} \rangle$ , основным множеством которой является множество  $P_n$  наборов объектов, обладающих  $n$  признаками, а основными операциями – операции  $\neg, \wedge, \vee$  на этом множестве [6].

Введем также следующие обозначения. Пусть  $N$  – количество всех возможных объектов, обладающих  $n$  признаками. Видно, что  $N = \prod_{j=1}^n |D| = 2^n$ . Обозначим множество  $N$ -мерных двоичных векторов через  $C_N$ , а булеву алгебру  $N$ -мерных двоичных векторов – через  $B_N = \langle C_N, \{\neg, \wedge, \vee\} \rangle$ . Покажем, что алгебры  $G_n$  и  $B_N$  изоморфны.

### 2.3. Изоморфизм алгебры объектов и булевой алгебры

Поставим в соответствие каждому объекту и набору объектов его код, представляющий собой последовательность нулей и единиц. Занумеруем все объекты от 1 до  $N$ . Произвольному объекту  $p$  поставим в соответствие код

$$c(p) = (\underbrace{0 \dots 0 \quad 1 \quad 0 \dots 0}_N),$$

где единица стоит в позиции, порядковый номер которой равен номеру объекта  $p$ .

Каждому набору объектов сопоставим код

$$c(P) = \bigvee_{p \in P} c(p).$$

Например, набор  $0_n$  имеет код  $C(0_n) = \underbrace{0\dots 0}_N$ , а набор  $1_n$  – код  $c(1_n) = \underbrace{1\dots 1}_N$ .

Так как во всех кодах наборов единицы стоят в различных позициях, то  $c$  устанавливает взаимно-однозначное соответствие между множеством кодов  $C_N$  и множеством наборов  $P_n$ .

Покажем, что  $c$  сохраняет операции над объектами и их наборами: код результата выполнения любой операции над набором или парой наборов равен результату соответствующей булевой операции над кодом данного набора или пары наборов, выполненной покомпонентно.

**Теорема 1.** Для любых наборов  $P$  и  $Q$  справедливы равенства:

- 1)  $\overline{c(P)} = c(\overline{P})$ ;
- 2)  $c(P) \wedge c(Q) = c(P \wedge Q)$ ;
- 3)  $c(P) \vee c(Q) = c(P \vee Q)$ .

Доказательство. Выберем произвольный объект  $p \in P$ , имеющий номер  $t$ . Проверим выполнение равенств 1–3 для соответствующих операций:

1. Отрицание:

$$c(P) = (\dots \overset{t}{1} \dots) \Leftrightarrow c(\overline{P}) = (\dots \overset{t}{0} \dots);$$

$$p \in P \Leftrightarrow p \notin \overline{P} \Leftrightarrow c(\overline{P}) = (\dots \overset{t}{0} \dots) = \overline{c(P)}.$$

2. Умножение: пусть одновременно выполняется  $p \in P$  и  $p \in Q$ ;

$$\left. \begin{array}{l} p \in P \Leftrightarrow c(P) = (\dots \overset{t}{1} \dots) \\ p \in Q \Leftrightarrow c(Q) = (\dots \overset{t}{1} \dots) \end{array} \right\} \Leftrightarrow c(P \wedge Q) = (\dots \overset{t}{1} \dots) = c(P) \wedge c(Q).$$

3. Сложение: пусть  $p$  не входит ни в  $P$ , ни в  $Q$ ;

$$\left. \begin{array}{l} p \notin P \Leftrightarrow c(P) = (\dots \overset{t}{0} \dots) \\ p \notin Q \Leftrightarrow c(Q) = (\dots \overset{t}{0} \dots) \end{array} \right\} \Leftrightarrow c(P \vee Q) = (\dots \overset{t}{0} \dots) = c(P) \vee c(Q).$$

Теорема доказана. ■

**Теорема 2.** Алгебра объектов  $G_n$  изоморфна булевой алгебре  $B_N$ .

Доказательство. Чтобы показать изоморфизм алгебр, нужно построить отображение  $h$  множества  $P_n$  в множество  $C_N$ , причем  $h$  должно обладать следующими свойствами [6]:

1. Отображение  $h$  должно являться гомоморфизмом, т. е. сохранять операции отрицания, пересечения и объединения:  $\overline{h(P)} = h(\overline{P})$ ,  $h(P) \wedge h(Q) = h(P \wedge Q)$ ,  $h(P) \vee h(Q) = h(P \vee Q)$ .

2. Отображение  $h$  должно быть взаимно-однозначным.

В качестве искомого отображения можно взять отображение  $c$ , ставящее в соответствие произвольному набору его код в виде последовательности нулей и единиц длины  $N$ . По теореме 1  $c$  – гомоморфизм; кроме того, как было показано выше,  $c$  является взаимно-однозначным, что и доказывает изоморфизм алгебр. ■

Из теоремы 2 следует эквивалентность прецедентного и логического способов представления информации. Для завершения решения подзадачи  $Z_{01}$  предложим конкретные алгоритмы перехода от логической модели представления информации к прецедентной и наоборот.

**2.4. Алгоритмы преобразования моделей представления информации**

Обозначим через  $In_{12}$  алгоритм перехода от логической модели к прецедентной (т. е. от  $I_0^1$  к  $I_0^2$ ), а через  $In_{21}$  – алгоритм обратного перехода. На рис. 2 показана связь между  $I_0^1$  и  $I_0^2$  через алгоритмы  $In_{12}$  и  $In_{21}$ .

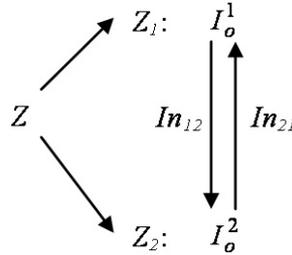


Рис. 2. Алгоритмы  $In_{12}$  и  $In_{21}$

В данном разделе рассматриваются только нормализованные объекты. Поэтому вначале покажем, что нормализация наборов сохраняет операции над ними.

**Теорема 3.** Для любых наборов  $P$  и  $Q$  справедливы равенства:

1.  $Norm(P) = Norm(\overline{P})$ .
2.  $Norm(P) \wedge Norm(Q) = Norm(P \wedge Q)$ .
3.  $Norm(P) \vee Norm(Q) = Norm(P \vee Q)$ .

Доказательство: Проверим выполнение равенств 1–3 для соответствующих операций:

1. Отрицание:

$$Norm(\overline{P}) = Norm(1_n \setminus P) = Norm(1_n) \setminus Norm(P) = 1_n \setminus Norm(P) = \overline{Norm(P)}.$$

2. Умножение.

Рассмотрим совместимые объекты  $p$  и  $q$ . Так как  $d_j^{Norm(\{p\})} \neq ?$ ,  $d_j^{Norm(\{q\})} \neq ?$ ,  $j = \overline{1, n}$ , то по определению произведения совместимых объектов  $d_j^{Norm(\{pq\})} = d_j^{Norm(\{p\})} = d_j^{Norm(\{q\})}$ . Из произвольности выбора  $j$  следует, что  $Norm(\{pq\}) = Norm(\{p\}) \wedge Norm(\{q\})$ .

$$\begin{aligned} Norm(P \wedge Q) &= Norm\left(\bigcup_{p \in P, q \in Q} \{pq\}\right) = \bigcup_{p \in P, q \in Q} Norm(\{pq\}) = \bigcup_{p \in P, q \in Q} (Norm(\{p\}) \wedge Norm(\{q\})) = \\ &= \left(\bigcup_{p \in P} Norm(\{p\})\right) \wedge \left(\bigcup_{q \in Q} Norm(\{q\})\right) = Norm\left(\bigcup_{p \in P} \{p\}\right) \wedge Norm\left(\bigcup_{q \in Q} \{q\}\right) = Norm(P) \wedge Norm(Q). \end{aligned}$$

3. Сложение:

$$\begin{aligned} Norm(P) \vee Norm(Q) &= Norm(P)Norm(Q) \cup \overline{Norm(P)Norm(Q)} \cup Norm(P)\overline{Norm(Q)} = \\ &= Norm(PQ) \cup Norm(\overline{PQ}) \cup Norm(P\overline{Q}) = Norm(P \vee Q). \end{aligned}$$

Теорема доказана. ■

Опишем подробнее логическую модель представления информации. В задаче  $Z_1$  начальная информация представлена в виде логических формул, используемых для описания принадлежности объектов классам. Введем переменные  $x_1, x_2, \dots, x_n$ , соответствующие значениям признаков  $s_1, s_2, \dots, s_n$ , где  $x_i \in D$ . Множество объектов, принадлежащих классу  $X_i$ , описывается логической формулой  $\phi$ , состоящей из переменных  $x_1, x_2, \dots, x_n$  и операций классической алгебры логики. Как принято в логике, не будем проводить различие между формулой и ее значением. Так как  $x_i \in D$ , то  $\phi: D^n \rightarrow D$ , т. е.  $\phi$  можно рассматривать как булеву функцию [7].

Рассмотрим объект  $p(s_1, s_2, \dots, s_n) = (d_1^p, d_2^p, \dots, d_n^p)$ . Пусть значение 0 на выходе  $\phi$  означает, что  $p \notin X_i$ , а значение 1 означает, что  $p \in X_i$ :

$$\phi(p) = \phi(d_1^p, \dots, d_n^p) = \begin{cases} 1, & p \in X_i; \\ 0, & p \notin X_i. \end{cases}$$

Поскольку  $\phi$  – булева функция, ее можно представить в виде СДНФ [7]:

$$\phi(x_1, \dots, x_n) = \bigvee_{(d_1, \dots, d_n)} x_1^{d_1} \dots x_n^{d_n} \phi(d_1, \dots, d_n), \quad (1)$$

где  $x_i^{d_i} = \begin{cases} x_i, & d_i = 1; \\ \overline{x_i}, & d_i = 0. \end{cases}$

Если  $X_i \neq \emptyset$ , то (1) можно сократить:

$$\phi(x_1, \dots, x_n) = \bigvee_{\substack{(d_1, \dots, d_n) \\ \phi(d_1, \dots, d_n)=1}} x_1^{d_1} \dots x_n^{d_n}. \quad (2)$$

Пусть  $r$  – число элементарных конъюнкций (ЭК) в (2). Обозначим  $j$ -ю ЭК через  $\phi_j(x_1, \dots, x_n) = x_1^{d_{j1}} \dots x_n^{d_{jn}}$ . Тогда  $\phi$  можно представить следующим образом:

$$\phi(x_1, \dots, x_n) = \bigvee_{j=1}^r \phi_j(x_1, \dots, x_n).$$

Исходя из результатов предыдущего раздела, можно утверждать, что каждой формуле  $\phi$  взаимно-однозначно соответствует набор  $P$ , представляющий собой множество объектов, описываемое формулой  $\phi$ . Поэтому алгоритм  $In_{12}$  по формуле  $\phi$  должен построить такой набор  $P$ , а алгоритм  $In_{21}$  по набору  $P$  – такую формулу  $\phi$ , что выполняется равенство

$$\phi(p) = 1 \Leftrightarrow p \in P. \quad (3)$$

*2.4.1. Алгоритмы  $In_{12}$  и  $In_{21}$ .* Опишем алгоритм  $In_{12}$ . Пусть задано правило  $\phi$  в виде (2). Необходимо построить набор  $P$ , для которого выполняется (3).

Алгоритм  $In_{12}$ :

*Шаг 1.* Каждой ЭК  $\phi_j(x_1, \dots, x_n) = x_1^{d_{j1}} \dots x_n^{d_{jn}}$  в (2) поставим в соответствие объект  $p_j(s_1, s_2, \dots, s_n) = (d_{j1}, \dots, d_{jn})$ .

*Шаг 2.* Построим набор  $P = p_1 \vee p_2 \vee \dots \vee p_r$ .

*Шаг 3.* Алгоритм завершает работу.

Опишем теперь алгоритм  $In_{21}$ . Пусть задан набор  $P$ . Необходимо указать правило  $\phi$  в виде (2), для которого выполняется (3).

Алгоритм  $In_{21}$ :

*Шаг 1.* Каждому объекту  $p_j(s_1, s_2, \dots, s_n) = (d_{j1}^p, \dots, d_{jn}^p)$ ,  $p_j \in P$ , поставим в соответствие ЭК  $\phi_j(x_1, \dots, x_n) = x_1^{d_{j1}^p} \dots x_n^{d_{jn}^p}$ .

*Шаг 2.* Набору  $P$  поставим в соответствие ДНФ  $\varphi(x_1, \dots, x_n) = \bigvee_{j=1}^r \varphi_j(x_1, \dots, x_n)$ , где  $r$  – количество объектов в наборе  $P$ .

*Шаг 3.* Алгоритм завершает работу.

2.4.2. *Свойства алгоритмов  $In_{12}$  и  $In_{21}$ .* Покажем, что для алгоритмов  $In_{12}$  и  $In_{21}$  выполняется (3). Пусть  $P = In_{12}(\varphi)$  обозначает, что набор  $P$  является результатом применения алгоритма  $In_{12}$  к формуле  $\varphi$ , а  $\varphi = In_{21}(P)$  обозначает, что  $\varphi$  является результатом применения алгоритма  $In_{21}$  к набору  $P$ .

**Теорема 4.** Для произвольных правила  $\varphi$  и объекта  $p$  выполняется (3):

$$\varphi(p) = 1 \Leftrightarrow p \in P, \text{ где } P = In_{12}(\varphi), p(s_1, s_2, \dots, s_n) = (d_1^p, d_2^p, \dots, d_n^p).$$

*Доказательство:*

1. Пусть  $\varphi(p) = 1$ . Тогда в  $\varphi$  существует хотя бы одна ЭК, для которой выполняется равенство  $\varphi_j(d_1^p, \dots, d_n^p) = 1$ . Для ЭК  $\varphi_j$  на шаге 1 построен объект  $p(s_1, s_2, \dots, s_n) = (d_1^p, d_2^p, \dots, d_n^p)$ , который на шаге 2 включен в набор  $P$ , что и требовалось доказать.

2. Пусть  $p \in P$ . На шаге 1 объект  $p$  поставлен в соответствие ЭК  $\varphi_j(d_1^p, \dots, d_n^p) = x_1^{d_1^p} \dots x_n^{d_n^p}$ . Так как ЭК  $\varphi_j$  входит в  $\varphi$ , то  $\varphi(p) = \varphi(d_1^p, \dots, d_n^p) = 1$ . ■

**Теорема 5.** Для произвольных набора  $P$  и объекта  $p$  выполняется (3):

$$p \in P \Leftrightarrow \varphi(p) = 1, \text{ где } \varphi = In_{21}(P), p(s_1, s_2, \dots, s_n) = (d_1^p, d_2^p, \dots, d_n^p).$$

*Доказательство.* Видно, что формула  $\varphi = In_{21}(P)$  имеет вид (2).

1. Пусть  $p \in P$ . На шаге 1 объекту  $p$  поставлена в соответствие ЭК  $\varphi_j(d_1^p, \dots, d_n^p) = 1$ . На шаге 2 построена ДНФ  $\varphi(d_1^p, \dots, d_n^p) = \bigvee_{j=1}^r \varphi_j(d_{j1}^p, \dots, d_{jn}^p) = 1$ , что и требовалось.

2. Пусть  $\varphi(p) = 1$ . Тогда в  $\varphi$  существует хотя бы одна ЭК, для которой выполняется равенство  $\varphi_j(d_1^p, \dots, d_n^p) = 1$ . На шаге 1 ЭК  $\varphi_j$  поставлена в соответствие объекту  $p(s_1, s_2, \dots, s_n) = (d_1^p, d_2^p, \dots, d_n^p)$ . Следовательно,  $p \in P$ . ■

**Следствие.** Преобразования, производимые алгоритмами  $In_{12}$  и  $In_{21}$ , взаимно обратны: для произвольных правила  $\varphi$ , набора  $P$  и объекта  $p$  выполняются следующие свойства:

$$1) \varphi(p) = 1 \Leftrightarrow (In_{21} \circ In_{12}(\varphi))(p) = 1;$$

$$2) p \in P \Leftrightarrow p \in (In_{12} \circ In_{21}(P)).$$

*Доказательство.*

$$1. \varphi(p) = 1 \Leftrightarrow [\text{по теореме 4}] \Leftrightarrow p \in In_{12}(\varphi) \Leftrightarrow [\text{по теореме 5}] \Leftrightarrow (In_{21} \circ In_{12}(\varphi))(p) = 1.$$

$$2. p \in P \Leftrightarrow [\text{по теореме 5}] \Leftrightarrow \varphi(d_1^p, \dots, d_n^p) = 1 \Leftrightarrow [\text{по теореме 4}] \Leftrightarrow p \in (In_{12} \circ In_{21}(P)). \blacksquare$$

### 2.5. Задача $Z_3$

Итак, решена подзадача  $Z_{01}$ : установлена эквивалентность прецедентной и логической моделей представления информации и разработаны алгоритмы для перехода между этими моделями. Преобразуем задачу  $Z_1$ : представим начальную информацию  $I_0^1$ , заданную логическим способом, в прецедентном виде (т. е. в том же виде, в котором представлена начальная

информация  $I_0^2$ ). Эту новую задачу обозначим через  $Z_3$ , а ее исходные данные и алгоритм решения – через  $I_0^3$  и  $A_3$  соответственно. Связь между задачами  $Z_1$ ,  $Z_2$  и  $Z_3$  показана на рис. 3.

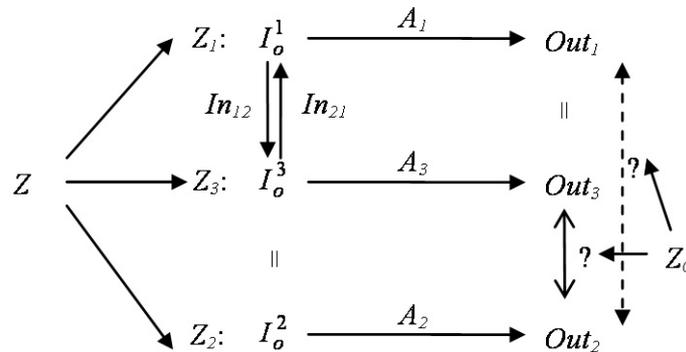


Рис. 3. Схема связи задач  $Z$ ,  $Z_1$ ,  $Z_2$ ,  $Z_3$ ,  $Z_0$

С одной стороны, задача  $Z_3$  представляет собой задачу  $Z_1$  с преобразованными исходными данными, поэтому задача  $Z_0$  сводится к сравнению решений задач  $Z_2$  и  $Z_3$ . С другой стороны, так как информация  $I_0^3$  представлена тем же способом, что и  $I_0^2$ , для решения задачи  $Z_3$  может быть применен алгоритм  $A_2$ , а для решения задачи  $Z_2$  – алгоритм  $A_3$ . Поэтому в дальнейшем будем отождествлять задачи  $Z_2$  и  $Z_3$  и говорить только о задаче  $Z_2$ , подразумеваемая задача, начальная информация в которой задана прецедентным способом.

### 3. Сравнение алгоритмов решения задачи $Z_2$

В данном разделе рассмотрим алгоритм  $A_3$ , представляющий собой модификацию метода резолюций для решения задачи  $Z_2$ . Этот модифицированный метод назовем *методом объектных резолюций*. Для решения подзадачи  $Z_{02}$  опишем параметрическое семейство алгоритмов, предложенное в [3], и сравним его с алгоритмом  $A_3$ .

#### 3.1. Метод объектных резолюций

Рассмотрим объекты  $p$  и  $q$  и признак  $s_k$ , такие, что  $d_k^p = 1$ ,  $d_k^q = 0$ . Будем считать, что для объектов  $p$  и  $q$  признак  $s_k$  единственный (т. е. значения остальных признаков либо совпадают, либо не заданы хотя бы в одном объекте). *Объектной резольвентой* назовем объект  $r$ , значения признаков которого определяются по следующему правилу:

$$d_j^r = \begin{cases} ?, j = k; \\ d_j^p \wedge d_j^q, j \neq k. \end{cases}$$

Через  $Or$  обозначим операцию построения объектной резольвенты:  $r = Or(p, q)$ .

Покажем, что операция построения объектной резольвенты эквивалентна операции построения резольвенты в классическом методе резолюций в следующем смысле: результаты применения метода объектных резолюций к объектам и классического метода резолюций к формулам, описывающим эти объекты, совпадают.

**Теорема 6.** Пусть для объектов  $p$  и  $q$   $\exists! s_k, d_k^p = 1, d_k^q = 0$ . Пусть  $t$  – объект, соответствующий обычной резольвенте для формул, описывающих  $p$  и  $q$ ;  $r = Or(p, q)$ . Тогда  $t = r$ .

Доказательство. Представим  $p$  и  $q$  в виде конъюнктов значений их объектов-признаков, как показано в [5], обозначив эти представления через  $P'$  и  $Q'$  соответственно:

$$P' = d_1^p \wedge \dots \wedge d_k^p \wedge \dots \wedge d_n^p, \quad Q' = d_1^q \wedge \dots \wedge d_k^q \wedge \dots \wedge d_n^q.$$

Чтобы применить к  $P'$  и  $Q'$  классический метод резолюций, преобразуем их в КНФ:

$$\begin{aligned} P' &= \overline{\overline{d_1^p} \vee \dots \vee \overline{d_k^p} \vee \dots \vee \overline{d_n^p}}, \quad Q' = \overline{\overline{d_1^q} \vee \dots \vee \overline{d_k^q} \vee \dots \vee \overline{d_n^q}} \Rightarrow \\ \Rightarrow \overline{P'} &= \overline{\overline{d_1^p} \vee \dots \vee \overline{d_k^p} \vee \dots \vee \overline{d_n^p}}, \quad \overline{Q'} = \overline{\overline{d_1^q} \vee \dots \vee \overline{d_k^q} \vee \dots \vee \overline{d_n^q}}. \end{aligned}$$

Так как  $\overline{d_k^p} = 0$ ,  $\overline{d_k^q} = 1$ , для  $\overline{P'}$  и  $\overline{Q'}$  можно построить резольвенту

$$R = \overline{\overline{d_1^p} \vee \dots \vee \overline{d_k^p} \vee \dots \vee \overline{d_n^p} \vee \overline{d_1^q} \vee \dots \vee \overline{d_k^q} \vee \dots \vee \overline{d_n^q}}.$$

Построим объект  $t$ , которому соответствует  $R$ :

$$\begin{aligned} R &= \overline{\overline{d_1^t} \vee \dots \vee \overline{d_{k-1}^t} \vee \overline{d_{k+1}^t} \vee \dots \vee \overline{d_n^t}} \Rightarrow \overline{R} = \overline{\overline{\overline{d_1^t} \vee \dots \vee \overline{d_{k-1}^t} \vee \overline{d_{k+1}^t} \vee \dots \vee \overline{d_n^t}}} = \\ &= \overline{\overline{d_1^t} \wedge \dots \wedge \overline{d_{k-1}^t} \wedge \overline{d_{k+1}^t} \wedge \dots \wedge \overline{d_n^t}} \Rightarrow t(s_1, \dots, s_n) = (d_1^t, \dots, d_{k-1}^t, ?, d_{k+1}^t, \dots, d_n^t). \end{aligned}$$

Сравним значения признаков  $t$  со значениями признаков  $r$ :

$$d_j^t = \begin{cases} \overline{\overline{?}}, & j = k \\ \overline{\overline{d_j^p \vee d_j^q}} = \overline{\overline{d_j^p \wedge d_j^q}}, & j \neq k \end{cases} = d_j^r, \quad j = \overline{1, n}.$$

Следовательно,  $d_j^t = d_j^r$  для всех  $s_j \in S$ . Поэтому  $t = r$ . ■

### 3.2. Алгоритмическая реализация метода объектных резолюций

Зафиксируем номер  $i$  класса  $X_i$  и определим, принадлежит ли объект  $x$  этому классу.

Алгоритм объектных резолюций  $A_3$ :

*Шаг 1.* Введем множество  $Y = X_i^0$ .

*Шаг 2.* Если  $x \in Y$ , то переходим к шагу 6, иначе – к шагу 3.

*Шаг 3.* Если все пары объектов уже рассматривались, то переходим к шагу 6. Иначе выбираем из  $Y$  такую нерассмотренную пару объектов  $p$  и  $q$ , что  $\exists s_k, d_k^p = 1, d_k^q = 0$ .

*Шаг 4.* Вычисляем  $r = Or(p, q)$ .

*Шаг 5.* Добавляем к  $Y$  объект  $r$ :  $Y := Y \cup \{r\}$ . Возвращаемся на шаг 2.

*Шаг 6.* Алгоритм завершает работу.

Алгоритм  $A_3$  можно применять как для прямого, так и для обратного вывода. В последнем случае на шаге 1 вводится множество  $Y = X_i^0 \cup \{x\}$ , а в качестве объекта  $x$  рассматривается объект, значения всех признаков которого неизвестны:  $o(s_1, \dots, s_n) = (? \dots ?)$ . В зависимости от типа вывода результат работы алгоритма  $A_3$  можно интерпретировать следующим образом:

1. Прямой вывод: если алгоритм закончил работу из-за получения объекта  $x$ , это значит, что, заменяя в описании множества  $X_i^0$  знаки «?» на конкретные значения признаков, можно получить  $x$ . Поэтому объект  $x$  принадлежит классу  $X_i$ .

2. Обратный вывод: если алгоритм закончил работу после шага 2 из-за получения объекта  $o$ , это значит, что, заменяя знаки «?» на конкретные значения признаков, из множества  $Y$  можно получить любой объект, т. е.  $Y$  потенциально содержит все возможные объекты из  $X$ . Поэтому объект  $x$  не может принадлежать классу  $X_i$ .

Если ни один из результатов 1 и 2 не получен, с помощью данного алгоритма нельзя сделать никаких выводов о принадлежности объекта  $x$  классу  $X_i$ .

Итак, алгоритм  $A_3$  представляет собой алгоритм  $A_1$ , преобразованный для прецедентной модели представления начальной информации. В процессе решения подзадачи  $Z_{01}$  была установлена эквивалентность прецедентной и логической моделей. Поэтому не будем проводить различие между алгоритмами  $A_1$  и  $A_3$  и при решении оставшихся подзадач задачи  $Z_0$  в качестве алгоритма  $A_1$  используем алгоритм  $A_3$ .

### 3.3. Семейство алгоритмов распознавания

Опишем вкратце параметрическое семейство алгоритмов, предложенное в [3].

Введем предварительно следующую функцию  $\mu: X^2 \rightarrow [0, 1]$ , характеризующую близость объекта  $x$  к объекту обучающей выборки  $y \in X_i^0$ :

$$\mu(x, y) = \max \left\{ 0, \left( \sum_{j \in S} (-1)^t a_{ij} \right) \cdot \left( \sum_{j \in S} a_{ij} \right)^{-1} \right\},$$

где  $\|a_{ij}\|$  – матрица, поставленная в соответствие множеству  $\{1, \dots, l\} \times |S|$  и удовлетворяющая следующим ограничениям:  $a_{ij} \in R$ ,  $\forall i, j$  ( $a_{ij} \geq 0$ ),  $\forall i$  ( $\sum_j a_{ij} > 0$ );  $l$  – число классов;  $S$  – мно-

жество признаков всех объектов;  $t = \begin{cases} 1, & d_j^x \neq d_j^y; \\ 2, & d_j^x = d_j^y. \end{cases}$

В определении функции предполагается, что все объекты нормализованы. В [3] предложена также одна из возможных схем вычисления параметров  $\|a_{ij}\|$ .

Опишем теперь алгоритм распознавания  $A_2$ , применяемый для решения задачи  $Z_2$ :

*Шаг 1.* Для всех объектов  $x \in X$  и  $x^0 \in X^0$  вычислим  $\mu(x, x^0)$ .

*Шаг 2.* Для всех  $i = \overline{1, l}$  вычислим  $P_i^{A_2}(x) = \max_{x_j^0 \in X_i^0} \{\mu(x, x_j^0)\}$ .

*Шаг 3.* Алгоритм завершает работу.

Видно, что  $P_i^{A_2}(x) \in [0, 1]$ ,  $i = \overline{1, l}$ . В данном случае предполагается, что  $P_i^{A_2}(x)$  отражает близость объекта  $x$  к классу  $X_i$ ; в частности, если  $P_i^{A_2}(x) = 1$ , то  $x \in X_i$ .

### 3.4. Сравнение алгоритмов объектных резолюций и распознавания

Введем систему предикатов  $P_1, \dots, P_l$ , характеризующих принадлежность объектов к классам  $X_1, \dots, X_l$ , как это сделано в [8]:

$$\forall x \in X \quad (P_i(x) \in \{0, 1\} \wedge (P_i(x) = 1 \Leftrightarrow x \in X_i)).$$

Как правило, точные значения этих предикатов известны только для объектов из обучающей выборки  $X_0$ ; задача  $Z_2$  состоит именно в том, чтобы определить значения предикатов  $P_1, \dots, P_l$  для остальных объектов.

По результатам работы алгоритмов  $A_2$  и  $A_3$  сформируем классификационные векторы  $A_2(x) = (P_1^{A_2}(x), \dots, P_l^{A_2}(x))$ ,  $A_3(x) = (P_1^{A_3}(x), \dots, P_l^{A_3}(x))$ . Предварительно приведем результаты алгоритма  $A_3$  к численному виду. Определим  $P_i^{A_3}(x)$  следующим образом:

$$P_i^{A_3}(x) = \begin{cases} 1, & x \in X_i; \\ -1, & x \notin X_i. \end{cases}$$

В определении  $P_i^{A_3}$  решение о принадлежности объекта  $x$  классу  $X_i$  принимается алгоритмом  $A_3$ .

Традиционно в теории распознавания в качестве показателя эффективности алгоритмов используется функционал качества, основанный на доле правильно классифицированных объектов [1]. Введем такой функционал качества, значения которого легко интерпретировались бы в терминах совпадения или близости  $P_i(x)$  и  $P_i^A(x)$ . Рассмотрим следующий функционал для произвольного алгоритма  $A$ , решающего задачу  $Z_2$  на множестве  $X' \subseteq X$ :

$$\Phi_A(X') = 1 - \frac{1}{l} \frac{1}{|X'|} \sum_{i=1}^l \sum_{x \in X'} \left| P_i(x) - P_i^{A_j}(x) \right|.$$

Функционал  $\Phi_A$  является частным случаем функционала, рассматриваемого в [8].

Из определения  $\Phi_A$  следует, что чем ближе значение  $\Phi_A(X')$  к 1, тем точнее алгоритм  $A$  работает на множестве  $X'$ ; в частности, если  $\Phi_A(X') = 1$ , то  $\forall x \in X' P_i^A(x) = P_i(x)$ . Поэтому из нескольких алгоритмов предпочтительным будем считать тот, который имеет наибольшее значение функционала  $\Phi_A$ .

**Теорема 7.** Пусть  $Y_i = \{x \mid P_i^{A_3}(x) = 1\}$ ,  $Y = \bigcup_{i=1}^l Y_i$ . Тогда

1)  $\Phi_{A_2}(Y) \leq \Phi_{A_3}(Y)$ ;

2)  $\Phi_{A_3}(X \setminus Y) \leq \Phi_{A_2}(X \setminus Y)$ .

**Доказательство:**

1. Если  $x \in Y_i$ , то  $P_i^{A_3}(x) = P_i(x) = 1 \Rightarrow |P_i(x) - P_i^{A_3}(x)| = 0 \Rightarrow \sum_{x \in Y} (|P_i(x) - P_i^{A_3}(x)|) = 0 \Rightarrow$   
 $\Rightarrow \Phi_{A_3}(Y) = 1 \Rightarrow \Phi_{A_2}(Y) \leq 1 = \Phi_{A_3}(Y)$ .

2. Если  $x \in X \setminus Y$ , то  $\forall i = \overline{1, l} P_i^{A_3}(x) = -1, P_i(x) \in \{0, 1\} \Rightarrow \forall i = \overline{1, l} |P_i(x) - P_i^{A_3}(x)| \geq 1 \Rightarrow$   
 $\Rightarrow \frac{1}{|X \setminus Y|} \sum_{x \in X \setminus Y} (|P_i(x) - P_i^{A_3}(x)|) \geq 1 \Rightarrow \Phi_{A_3}(X \setminus Y) \leq 0 \Rightarrow \Phi_{A_3}(X \setminus Y) \leq 0 \leq \Phi_{A_2}(X \setminus Y)$ . ■

Из теоремы 7 можно сделать вывод, что на множестве  $Y$  предпочтительно использовать алгоритм  $A_3$ , а на множестве  $X \setminus Y$  – алгоритм  $A_2$ . Таким образом, подзадача  $Z_{02}$  решена.

#### 4. Комбинированный алгоритм

Перейдем к подзадаче  $Z_{03}$ . При сравнении алгоритмов было показано, что на некотором множестве наиболее точные результаты дает алгоритм  $A_3$ , а на дополнении этого множества – алгоритм  $A_2$ . Чтобы достичь наилучших результатов, построим комбинированный алгоритм  $A_C$ , объединяющий  $A_2$  и  $A_3$ , и покажем, что  $\Phi_{A_C}(X)$  не меньше, чем  $\Phi_{A_2}(X)$  и  $\Phi_{A_3}(X)$ .

#### 4.1. Описание комбинированного алгоритма

Введем алгоритм  $A_C$ , являющийся объединением алгоритмов  $A_2$  и  $A_3$ . Алгоритмы объектных резолюций и распознавания были описаны выше, поэтому не будем приводить их в подробностях, а лишь включим их в описание алгоритма  $A_C$  как отдельные шаги.

Комбинированный алгоритм  $A_C$ :

*Шаг 1.* Выберем объект  $x \in X$ .

*Шаг 2.* Выберем номер класса  $i = \overline{1, l}$ .

*Шаг 3.* Применим алгоритм объектных резолюций  $A_3$  к объекту  $x$  и множеству  $X_i^0$ .

*Шаг 4.* Если  $P_i^{A_3}(x) = 1$ , то переходим к шагу 7. Иначе переходим к шагу 5.

*Шаг 5.* Если все классы исчерпаны, то переходим к шагу 6. Иначе переходим к шагу 2.

*Шаг 6.* Применим алгоритм распознавания  $A_2$  к объекту  $x$ .

*Шаг 7.* Если все объекты исчерпаны, то переходим к шагу 8. Иначе переходим к шагу 1.

*Шаг 8.* Алгоритм завершает работу.

Результаты алгоритма  $A_C$  для объекта  $x$  также представим в виде классификационного вектора  $A_C(x) = (P_1^{A_C}(x), \dots, P_l^{A_C}(x))$ , где  $P_i^{A_C}(x) \in [0, 1]$ .

*Замечание.* В алгоритме  $A_C$  алгоритм объектных резолюций используется для прямого вывода. Иногда целесообразно использовать обратный вывод; например, это ситуация, когда  $l = 2$ , т. е. требуется определить, принадлежит ли объект классу или его дополнению. В этом случае нужно на шаге 3 применить алгоритм  $A_3$  дважды: для прямого и обратного вывода.

#### 4.2. Сравнение алгоритмов

Докажем предварительно одно свойство функционала  $\Phi_A$ , которое используем при исследовании комбинированного алгоритма.

*Свойство функционала  $\Phi_A$ .* Если  $X = Y \cup Z$ , где  $Y \cap Z = \emptyset$ , то

$$|X|\Phi_A(X) = |Y|\Phi_A(Y) + |Z|\Phi_A(Z). \quad (4)$$

*Доказательство.* Так как  $Y \cap Z = \emptyset$ , то  $|X| = |Y| + |Z|$ . Тогда по определению  $\Phi_A$

$$\begin{aligned} (|Y| + |Z|)\Phi_A(Y \cup Z) &= (|Y| + |Z|) - \frac{1}{l} \sum_{i=1}^l \sum_{x \in Y \cup Z} (|P_i(x) - P_i^A(x)|) = \\ &= |Y| - \frac{1}{l} \sum_{i=1}^l \sum_{x \in Y} (|P_i(x) - P_i^A(x)|) + |Z| - \frac{1}{l} \sum_{x \in Z} (|P_i(x) - P_i^A(x)|) = |Y|\Phi_A(Y) + |Z|\Phi_A(Z), \end{aligned}$$

откуда следует (4). ■

Покажем теперь, что функционалы качества алгоритмов  $A_2$  и  $A_3$  не превосходят функционала качества алгоритма  $A_C$  на множестве  $X$ .

**Теорема 8.**  $\Phi_{A_C}(X) \geq \max\{\Phi_{A_2}(X), \Phi_{A_3}(X)\}$ .

*Доказательство.* Пусть  $Y_i = \{x \mid P_i^{A_3}(x) = 1\}$ ,  $Y = \bigcup_{i=1}^l Y_i$ . Если  $x \in Y$ , то на шаге 3 алгоритма  $A_C$  получен результат  $P_i^{A_C}(x) = P_i^{A_3}(x)$ , т. е.  $\Phi_{A_C}(Y) = \Phi_{A_3}(Y)$ . Если же  $x \in X \setminus Y$ , то на шаге 6 алгоритма  $A_C$  получен результат  $P_i^{A_C}(x) = P_i^{A_2}(x)$ , т. е.  $\Phi_{A_C}(X \setminus Y) = \Phi_{A_2}(X \setminus Y)$ .

Используя свойство (4), получаем

$$|X|\Phi_{A_C}(X) = |Y|\Phi_{A_C}(Y) + |X \setminus Y|\Phi_{A_C}(X \setminus Y) = |Y|\Phi_{A_3}(Y) + |X \setminus Y|\Phi_{A_2}(X \setminus Y).$$

Покажем, что  $\Phi_{A_C}(X) \geq \Phi_{A_2}(X)$  и  $\Phi_{A_C}(X) \geq \Phi_{A_3}(X)$ :

$$|X|(\Phi_{A_C}(X) - \Phi_{A_2}(X)) = |Y|(\Phi_{A_3}(Y) - \Phi_{A_2}(Y)) \geq 0 \quad (\text{по теореме 7});$$

$$|X|(\Phi_{A_C}(X) - \Phi_{A_3}(X)) = |X \setminus Y|(\Phi_{A_2}(X \setminus Y) - \Phi_{A_3}(X \setminus Y)) \geq 0 \quad (\text{по теореме 7}).$$

Отсюда получаем, что  $\Phi_{A_C}(X) \geq \max\{\Phi_{A_2}(X), \Phi_{A_3}(X)\}$ . ■

Из доказательства теоремы 8 следует, что  $\Phi_{A_C}(X) > \max\{\Phi_{A_2}(X), \Phi_{A_3}(X)\}$  тогда и только тогда, когда одновременно  $\Phi_{A_3}(Y) > \Phi_{A_2}(Y)$  и  $\Phi_{A_2}(X \setminus Y) > \Phi_{A_3}(X \setminus Y)$ .

Главное преимущество алгоритма  $A_C$  по сравнению с  $A_3$  заключается в том, что  $A_3$  позволяет получить результат лишь для тех объектов, описание которых можно вывести из описаний множеств  $X_i^0$ , а алгоритм  $A_C$  дает оценки для всех объектов из  $X$ . Однако для объектов, принадлежность которых определена на этапе применения  $A_3$ , результат является обоснованным, в чем заключается основное преимущество  $A_C$  по сравнению с  $A_2$ .

Таким образом, подзадача  $Z_{03}$  решена: построен алгоритм, объединяющий алгоритмы объектных резолюций и распознавания и работающий на  $X$  не хуже обоих этих алгоритмов.

### Заключение

В работе рассмотрены два подхода к решению задач распознавания, имеющие дедуктивную и индуктивную природу. В задачах, решаемых при помощи этих подходов, используются соответственно логическая и прецедентная модели представления информации. Показано, что в булевом пространстве при определенных ограничениях эти модели эквивалентны. Предложен метод объектных резолюций как модификация метода резолюций для задач распознавания. Разработан комбинированный алгоритм, объединяющий метод объектных резолюций с семейством алгоритмов распознавания, который работает не хуже любого из алгоритмов, образующих комбинацию.

На текущем этапе работа носит теоретический характер. Однако предшествующие исследования по теме статьи уже нашли практическое применение. Например, результаты, полученные в [3], использовались при разработке интеллектуальной системы поддержки решений в спортивной травматологии, описанной в [9].

В качестве следующего этапа исследования предполагается обобщение рассмотренной в статье задачи для случаев, когда признаки объектов принимают произвольное число значений. Планируется также рассмотрение комбинаторных характеристик предлагаемого алгоритма, его численных оценок, особенностей практического использования.

### Список литературы

1. Журавлев, Ю.И. Об алгебраическом подходе к решению задач распознавания или классификации / Ю.И. Журавлев // Проблемы кибернетики. – 1978. – № 33. – С. 5–68.
2. Журавлев, Ю.И. Корректные алгебры над множествами некорректных (эвристических) алгоритмов / Ю.И. Журавлев // Кибернетика. – 1977. – № 4. – С. 14–21.
3. Краснопрошин, В.В. Распознавание с обучением как задача выбора / В.В. Краснопрошин, В.А. Образцов // Цифровая обработка изображений. – Минск : Ин-т техн. кибернетики НАН Беларуси, 1998. – С. 80–94.
4. Чень, Ч. Математическая логика и автоматическое доказательство теорем / Ч. Чень, Р. Ли. – М. : Наука, 1983. – 360 с.

5. Рябцев, А.В. Алгебры для представления обучающей информации в задачах распознавания образов / А.В. Рябцев // Цифровая обработка. – 2002. – № 6. – С. 80–94.
6. Мальцев, А.И. Алгебраические системы / А.И. Мальцев. – М. : Наука, 1970.
7. Яблонский, С.В. Введение в дискретную математику / С.В. Яблонский – М. : Наука, 1986. – 384 с.
8. Краснопрошин, В.В. Проблема принятия решений по прецедентности: разрешимость и выбор алгоритмов / В.В. Краснопрошин, В.А. Образцов // Выбранные научные работы Беларускага дзяржаўнага ўніверсітэта. Т. 6. Матэматыка. – Минск : БГУ, 2001. – С. 285–312.
9. Интеллектуальная система поддержки решений в спортивной травматологии / В.В. Краснопрошин [и др.] // Вестник Национального технического университета «ХПИ». Тематический выпуск: информатика и моделирование. – 2010. – № 31. – С. 106–111.

Поступила 17.04.12

*Белорусский государственный университет,  
Минск, пр. Независимости, 4  
e-mail: olgashut@tut.by*

**O.V. Shut**

### **A SOLUTION METHOD OF RECOGNITION IN BOOLEAN SPACE BASED ON LOGICAL AND PRECEDENT MODELS**

The paper considers deductive and inductive approaches to recognition problems. These approaches use logical and precedent related information encoding models, respectively. Algebra of objects, which is isomorphic to Boolean algebra, is proposed. Algorithms for conversion of information encoding models are developed. A modification of the resolution method for the precedent model is suggested. A new algorithm, which combines the resolution method and recognition algorithms, is developed.

*ПРИКЛАДНЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ  
И ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ*

УДК 681.327.11:615.8

**В.А. Лапицкий, Л.Н. Лаханько, А.Г. Батраков, В.В. Кириченко**

**РАСПРЕДЕЛЕННАЯ ТЕЛЕМЕДИЦИНСКАЯ СИСТЕМА  
ПО ЦИФРОВОЙ МАММОГРАФИИ**

*Рассматриваются принципы построения и функционирования распределенной телемедицинской системы Минска для проведения дистанционных консультаций по цифровым маммографическим исследованиям. Описываются используемый вариант хранения и передачи медицинских изображений, а также реализованные пользовательские функции для врача-консультанта.*

**Введение**

Среди стратегических механизмов реформирования системы здравоохранения важное место отводится внедрению новых медицинских технологий, способных вывести отрасль из кризисного состояния, преодолеть затратные механизмы, обеспечить выполнение государственных гарантий доступности и качества оказания медицинской помощи населению независимо от места проживания, социального статуса, уровня доходов. К технологиям, способным оказать позитивное комплексное воздействие на клинические, образовательные, научные и управленческие аспекты деятельности системы здравоохранения и обладающим высоким медицинским, социальным и экономическим потенциалом, относится телемедицина.

В комплексе диагностических методов, применяемых в настоящее время в клинической медицине, лучевая диагностика является неотъемлемой частью клинического обследования больного и одним из основных источников получения объективной информации, необходимой для современного установления правильного диагноза, точной локализации и протяженности выявленных патологических изменений, планирования оперативного вмешательства, а также динамики в процессе лечения [1].

До последнего времени основным носителем диагностической информации была рентгеновская пленка и термобумага. В настоящее время имеются широкие возможности передачи изображений по локальным и глобальным сетям. Электронный обмен медицинскими изображениями позволяет обеспечить развитие телемедицины с использованием экономически оправданных, технически и организационно оптимальных заочных сеансов удаленного консультирования [2].

В настоящее время в Минске работает около десяти маммографических аппаратов «Маммоскан», позволяющих получать и передавать изображение молочной железы в оцифрованном виде. Это обстоятельство на определенном этапе эксплуатации аппаратов, наряду с развитием коммуникационных сетей, предопределило развитие телемедицинской маммографической технологии. Возникла необходимость в создании общего механизма работы аппаратов «Маммоскан» поликлиник, а также в объединении их единой сетью с онкологическим диспансером как консультативным учреждением города.

В настоящей статье приводится описание разработанного пилотного образца распределенной телемедицинской системы Минска для проведения дистанционных консультаций по цифровым маммографическим исследованиям.

Основная задача телемедицинской системы цифровой маммографии – обеспечение общедоступности и единого высокого стандарта качества медицинского обслуживания в любом медицинском учреждении независимо от его территориального или иерархического положения.

Это обеспечивается благодаря предоставлению возможности оперативной дистанционной консультации высококвалифицированного специалиста-маммолога без его выезда к месту проведения исследования.

### **1. Краткая формулировка методов решения поставленной задачи**

Разработанная телемедицинская система реального времени является сложной корпоративной системой, обеспечивающей выполнение высоких требований как к информационной компоненте по организации доступа к большим базам данных, так и к вычислительной адаптивной компоненте, реализующей процедуры логического вывода, усвоения и мониторинга информации, а также имитационного моделирования и проигрывания сценариев.

При разработке информационной модели телемедицинской системы реального времени учитывалась выявленная особенность ее функционирования, заключающаяся в необходимости обработки и хранения большого объема информации на двух различных уровнях иерархии:

1. Обработка данных запроса лечащего врача в реальном времени. Операция подразумевает идентификацию состояния пациента по результатам сопоставления его показателей с информацией специализированных баз данных, а также с информацией, получаемой путем аналитической обработки данных этих баз. Поэтому в первую очередь используются поиск, ранжирование и шкалирование многомерных данных, а также их аналитическая обработка специальными алгоритмами.

2. Усвоение данных телемедицинских консультаций в системе. Операция необходима для постоянного пополнения встроенной базы данных информацией, накапливаемой в процессе эксплуатации системы. Поэтому на данном уровне обработки информации ведущую роль играют структуризация и каталогизация данных, а также уточнение параметров моделей в базе знаний.

Разработка телемедицинской системы цифровой маммографии (ТМС ЦМ) реального времени опиралась на следующие основные принципы построения:

– кластерная архитектура для организации ядра информационной ТМС. При этом сравнительно небольшое количество вычислительных узлов, необходимых для реализации данной задачи, не налагает кардинальных требований на топологию связи между ними;

– организация параллельного функционирования информационной ТМС на кластере. При этом организация баз данных ТМС на кластере должна предусматривать повышение надежности системы в целом, возможность вертикального и горизонтального распараллеливания обработки запросов, а также оптимизацию параллельного размещения данных.

Создание ТМС как многоуровневого проблемно-ориентированного программного комплекса для ввода, передачи, обработки, хранения и отображения медицинской информации осуществлялось в несколько этапов:

1) проектирование операционного и логического уровней компонент модели ТМС (алгоритмы обработки данных и абстрактные структуры);

2) проектирование экранного уровня и уровня хранения данных (формы диалога, структура и содержание базы данных).

Высокая производительность ТМС ЦМ достигается с помощью алгоритмов распараллеливания процессов ввода-вывода данных.

Распараллеливание ввода-вывода в сочетании с оптимальным планированием заданий позволяет осуществить весьма эффективный одновременный доступ к фрагментированным таблицам и индексам, расположенным на нескольких физических дисках в разных узлах кластера, тем самым многократно ускоряя операции с относительно медленными внешними устройствами (рис. 1).

### **2. Состав и основные функции ТМС ЦМ**

ТМС ЦМ – это распределенная многоуровневая система, функционирование которой осуществляется по технологии «клиент – сервер» в рамках корпоративной телекоммуникационной сети (VPN-сети) медицинских учреждений Минска.

В структуру аппаратно-программного и телекоммуникационного обеспечения ТМС ЦМ (рис. 2) входят следующие основные составляющие:

1) выделенный сервер кластерного типа, в котором реализована отказоустойчивая архитектура с общей подсистемой хранения данных, поддерживающая режимы резервирования вычислительных узлов и каналов доступа к системе хранения данных;

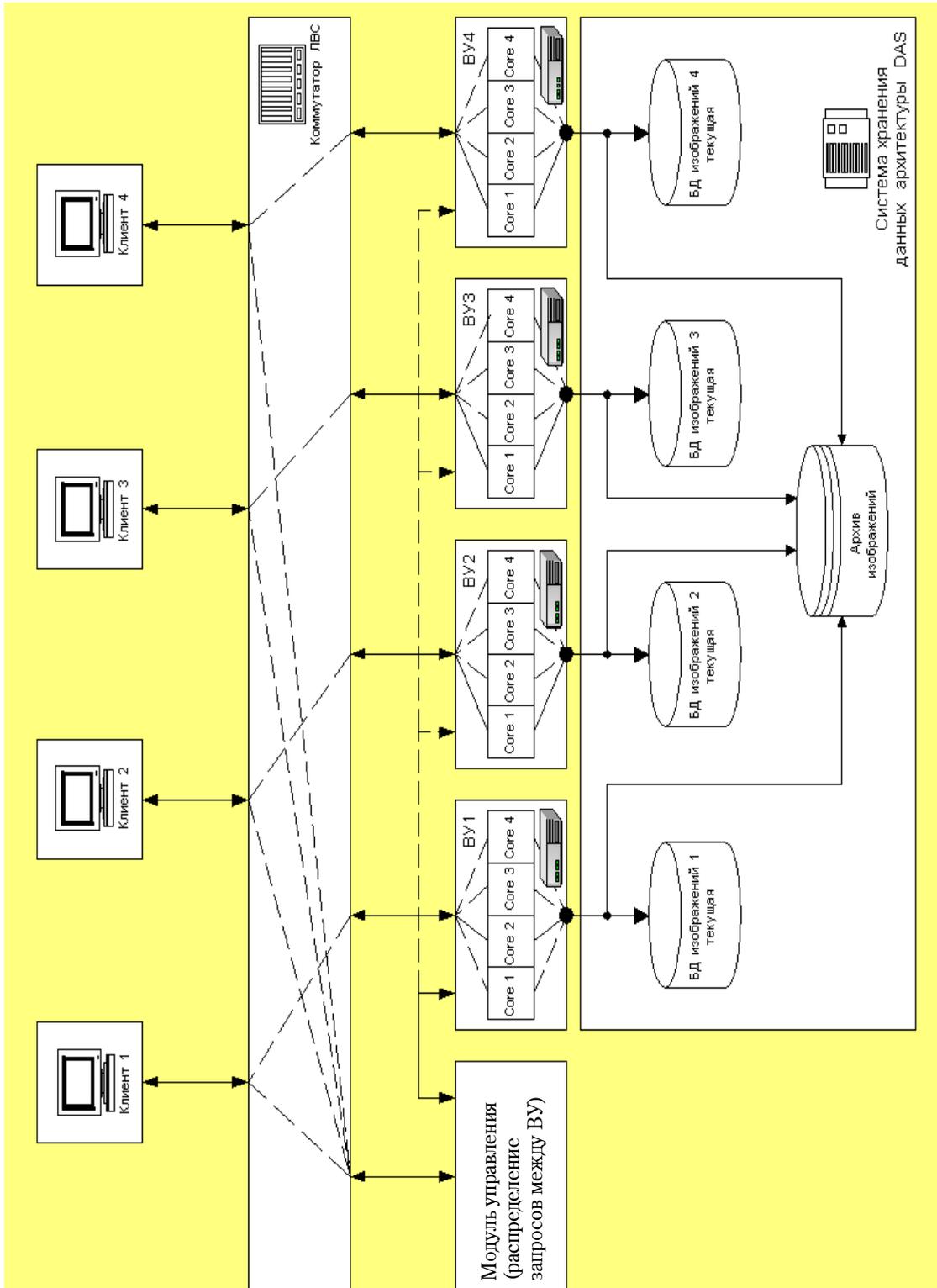


Рис. 1. Функциональная схема параллельной обработки данных на сервере ТМС ЦМ

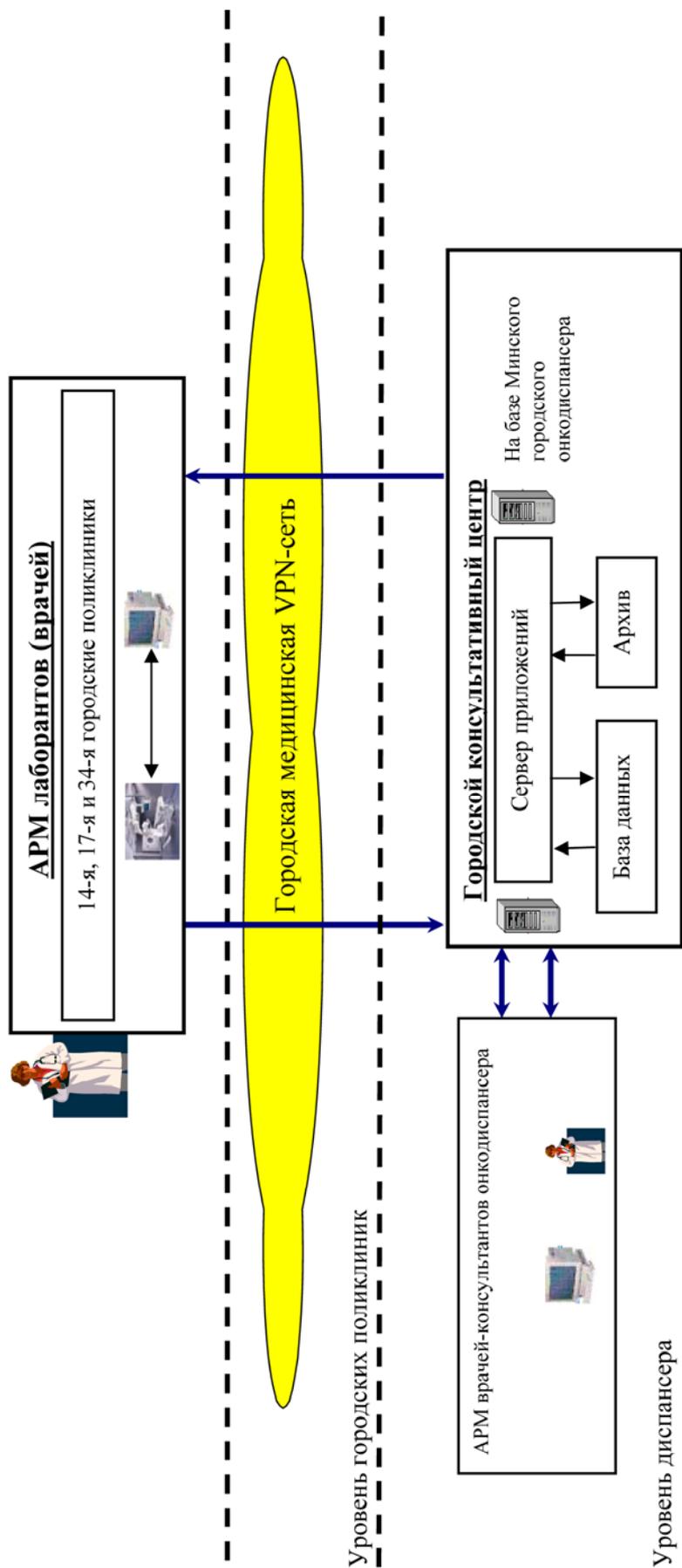


Рис. 2. Общая структура ТМС ЦМ Минска

2) автоматизированные рабочие места (АРМ) врачей-диагностов (рентген-лаборантов) в трех городских поликлиниках Минска. АРМ врача-диагноста (рентген-лаборанта) представляет собой программно-технический комплекс на базе персонального компьютера и обеспечивает аппаратно-программное сопряжение по цифровому интерфейсу (DICOM 3.0) с рабочими станциями маммографического аппарата (цифрового маммографа);

3) АРМ врача-консультанта в специализированном консультационном центре, созданном на базе Минского городского онкологического диспансера;

4) программно-технические средства передачи данных на базе VPN-сети. VPN-сеть построена на базе цифровых систем передачи данных (сети SDH) РУП «Белтелеком». Пропускная способность линии диспансера – 2 Мбит/с, пропускная способность линий поликлиник – 1 Мбит/с.

В целом ТМС ЦМ обеспечивает:

– создание и ведение единого архива электронных медицинских карт и диагностических протоколов обследования пациентов, переданных из поликлиник;

– создание и ведение единого архива медицинских рентгеновских изображений, переданных из поликлиник;

– доступ медицинских консультантов к телемедицинской базе данных;

– создание и ведение архива консультационных протоколов и автоматическую их рассылку в поликлиники;

– создание и ведение экспертного электронного атласа медицинских рентгеновских изображений и исследований и удаленный доступ к нему;

– обмен разнообразной информацией между пользователями ТМС ЦМ путем передачи диагностического изображения, текстовых и графических файлов с унифицированной тактикой диагностики и лечения;

– ведение статистики и получение различных аналитических данных за требуемый промежуток времени по проведенным телемедицинским консультациям.

### **3. Методика дистанционного консультирования по цифровым маммографическим исследованиям**

Рассмотрим блок-схему функционирования ТМС ЦМ (рис. 3).

Источником информации для распределенной базы данных ТМС ЦМ являются АРМ рентген-лаборантов, обслуживающих цифровые маммографы городских поликлиник. В данных поликлиниках осуществляется регистрация пациентов, их фактографических данных и выполняется маммографическое исследование с получением четырех, а иногда и более цифровых рентгеновских изображений. В течение рабочего дня рентген-лаборант поликлиники по мере накопления информации запускает процедуру передачи первичных диагностических данных в онкологический диспансер. АРМ лаборантов поликлиник связаны с сервером базы данных онкологического диспансера с помощью выделенной телефонной линии связи с использованием ADSL-модемов.

С целью повышения скорости передачи данных информация сжимается в 1,5–2 раза без потери качества. Время передачи одного исследования при скорости канала связи 1 Мбит/с составляет порядка 5–7 мин. Ежедневно каждая поликлиника выполняет порядка 10–25 исследований в зависимости от численности населения обслуживаемой территории и времени года.

В Минском городском онкологическом диспансере работает телемедицинский программный сервер ТМС ЦМ, который принимает информацию от поликлиник и сохраняет ее в базе данных ТМС ЦМ. В диспансере организована локальная вычислительная сеть, в которую объединены автоматизированные рабочие места врачей-рентгенологов (врачей-консультантов), подключенные к базе данных диспансера. По мере поступления информации врачи-рентгенологи выполняют анализ медицинских данных и цифровых рентгеновских изображений, с помощью специального программного модуля формируют их цифровые портреты и описывают результаты исследований в текстовом виде для поликлиник. Результаты работы врачей-рентгенологов (врачей-консультантов) сохраняются в базе данных ТМС ЦМ.

По истечении определенного времени от момента передачи данных (2–5 ч) рентген-лаборанты поликлиник подключаются к серверу ТМС ЦМ, получают от него результаты работы врачей-рентгенологов (врачей-консультантов) в виде протоколов-описаний, распечатывают их и выдают пациентам.

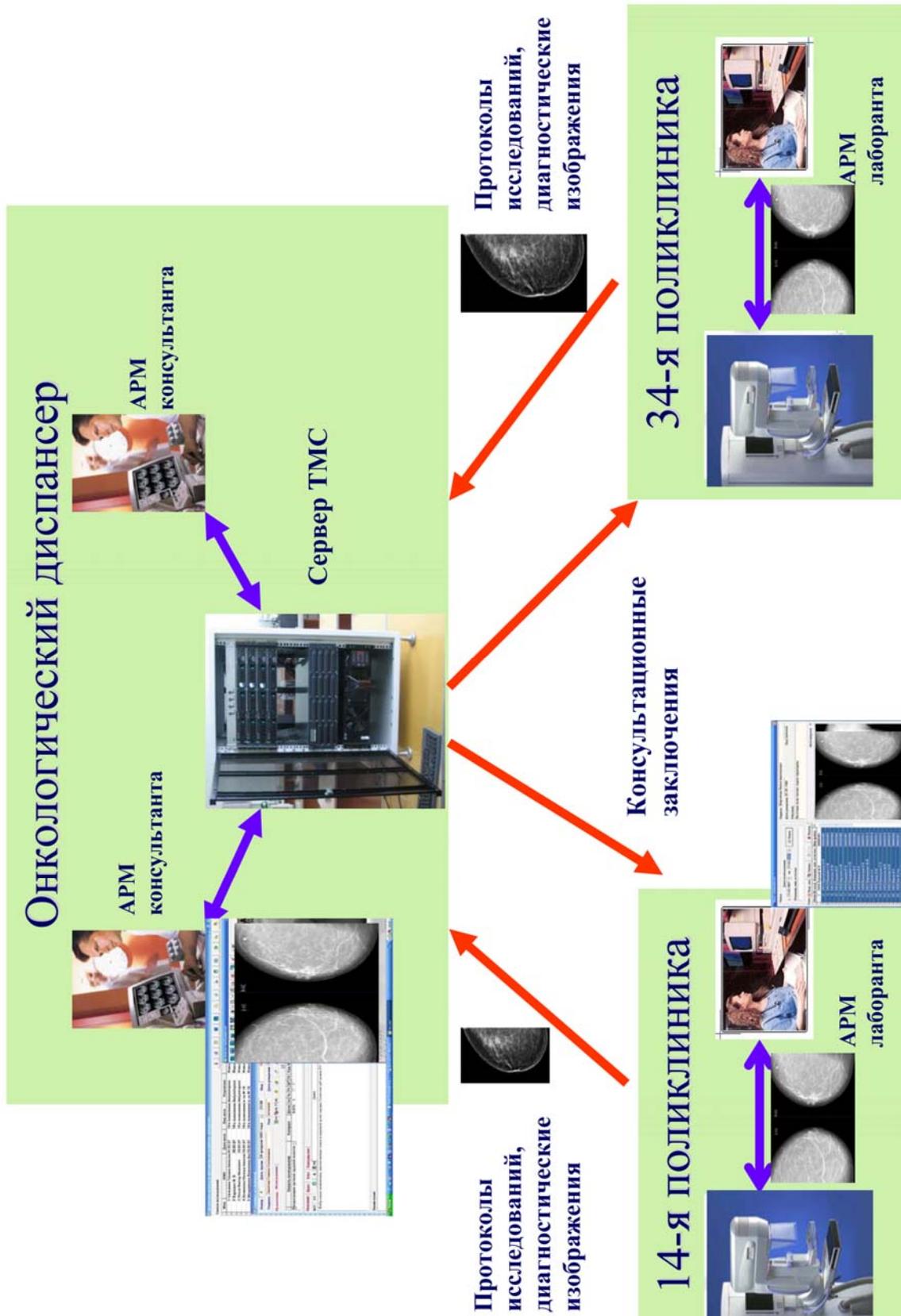


Рис. 3. Блок-схема функционирования ТМС ЦМ

#### 4. Сжатие и передача медицинских изображений в формате DICOM в рамках ТМС ЦМ

DICOM (Digital Imaging and Communications in Medicine, цифровые изображения и обмен ими в медицине) – индустриальный стандарт для передачи радиологических изображений и другой медицинской информации между компьютерами, опирающийся на стандарт Open System Interconnection (OSI), разработанный Международной организацией по стандартизации (International Standards Organization).

В настоящее время стандартом DICOM предусмотрены несколько вариантов хранения медицинских изображений. Одним из самых распространенных вариантов на территории Республики Беларусь является формат данных RAW, который подразумевает хранение графической информации в DICOM-файле в «сыром» (необработанном и несжатом) виде. Такой формат позволяет избежать потерь информации, и он наиболее пригоден для последующей обработки. Кроме того, при использовании формата RAW нет необходимости встраивать в программное обеспечение специфические алгоритмы по кодированию и декодированию графической информации. DICOM-файлы с графической информацией в формате RAW являются результатом маммографического исследования маммографами компании Adani, которые используются в большинстве поликлиник, задействованных в ТМС ЦМ.

Однако у формата RAW есть существенный недостаток – большой размер. Поэтому современные производители цифрового медицинского оборудования оснащают свои решения специальными алгоритмами, позволяющими получать медицинские изображения в сжатом виде как с потерей качества, так и без потерь. Стандартом DICOM поддерживаются следующие схемы компрессии и декомпрессии:

RLE (Run Length Encoding), или кодирование повторов, – простой алгоритм сжатия данных, который оперирует сериями данных, т. е. последовательностями, в которых один и тот же символ встречается несколько раз подряд. При кодировании строка одинаковых символов, составляющих серию, заменяется строкой, которая содержит сам повторяющийся символ и количество его повторов;

JPEG – обеспечивает сжатие графической информации с потерей качества. В основе алгоритма лежит дискретное косинусное преобразование [3];

JPEG-LS – основан на формате LOCO-I (Low Complexity Lossless Compression for Images). Этот алгоритм использует адаптивное предсказание значения текущего пиксела по окружению, включающему уже закодированные пиксели, классификацию контекста, контекстное моделирование ошибки предсказания и ее коррекцию, а также энтропийное кодирование скорректированной ошибки [4];

JPEG-2000 – использует технологию вейвлет-преобразования, основывающуюся на представлении сигнала в виде суперпозиции базовых функций – волновых пакетов.

Передача медицинской информации в формате DICOM осуществляется в два этапа. Первый – передача медицинской информации от диагностического оборудования в архив медицинских изображений и в АРМ врача-диагноста. Архив медицинских изображений поставляется вместе с диагностическим оборудованием и представляет собой отдельный блок с жесткими носителями информации. На этом этапе используется сетевой DICOM-протокол, который является частью стандарта DICOM. Врач-диагност запрашивает DICOM-файлы из архива при помощи сервиса DICOM Query, который также является обязательной частью реализации стандарта DICOM. После этого медицинские изображения, удовлетворяющие критериям запроса, передаются в АРМ врача-диагноста.

Второй этап передачи медицинской информации в формате DICOM реализуется в том случае, когда врач-диагност посылает запрос на консультацию. При этом происходит передача DICOM-файла и сопутствующей информации из АРМ врача-диагноста в АРМ врача-консультанта по VPN-сети. На втором этапе файлы передаются с использованием стандартного стека протоколов TCP/IP. Стек протоколов TCP/IP и сопутствующий ему набор системных утилит предназначены для организации сетевого взаимодействия между различными компьютерами. Семейство TCP/IP фактически состоит из двух протоколов: IP (Internet Protocol) и TCP (Transmission Control Protocol).

Реализация и использование широко распространенных и развивающихся технологий и стандартов создает большой задел на будущее. Модернизация и совершенствование стандарта DICOM, внедрение новых алгоритмов сжатия медицинских изображений позволит в дальней-

шем сократить объемы передаваемой информации, время на ее передачу, а значит, и финансовые издержки, связанные с этими показателями.

### 5. Краткое описание программного обеспечения ТМС ЦМ

В структуру комплекса программного обеспечения ТМС ЦМ входят:

- программный комплекс центрального вычислительного комплекса (сервер базы данных);
- программный комплекс АРМ рентген-лаборантов в городских поликлиниках Минска;
- программный комплекс АРМ врачей-консультантов в Минском городском онкологическом диспансере.

АРМ рентген-лаборанта, входящее в состав ТМС ЦМ, представляет собой программно-технический комплекс на базе персонального компьютера, аппаратно сопряженный по цифровому (DICOM 3.0) интерфейсу с маммографическим аппаратом.

В АРМ врачей-консультантов реализованы следующие функции просмотра исследований:

- доступ к серверу ТМС ЦМ;
  - идентификация пациентов, материалы которых присланы на консультацию;
  - идентификация медицинских специалистов и организаций, от которых поступили материалы на консультацию;
  - поиск по базе данных и просмотр исследований;
  - просмотр протокола исследования;
  - загрузка и просмотр изображений;
  - печать на Windows-принтерах;
  - увеличение-уменьшение изображения;
  - использование экранной лупы и кинопетли;
  - регулировка и предустановка диапазона рентгеновской плотности Window/Level;
  - копирование изображения в буфер обмена;
  - копирование протокола в буфер обмена;
  - измерение расстояний, углов;
  - повороты, зеркалирование;
  - шифрование данных (SSL);
  - просмотр DICOM-информации;
  - инверсия;
  - разделение рабочего стола;
  - экспорт изображений в форматы tiff, jpeg;
  - формулировка и отправка консультационного заключения на сервер ТМС ЦМ.
- Примеры рабочих экранов АРМ врачей-консультантов показаны на рис. 4 и 5.

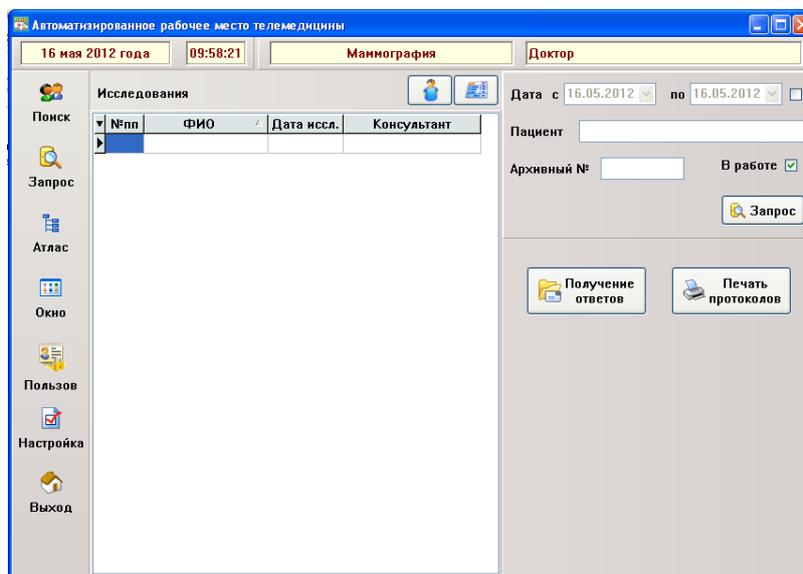


Рис. 4. Основное рабочее окно врача-консультанта

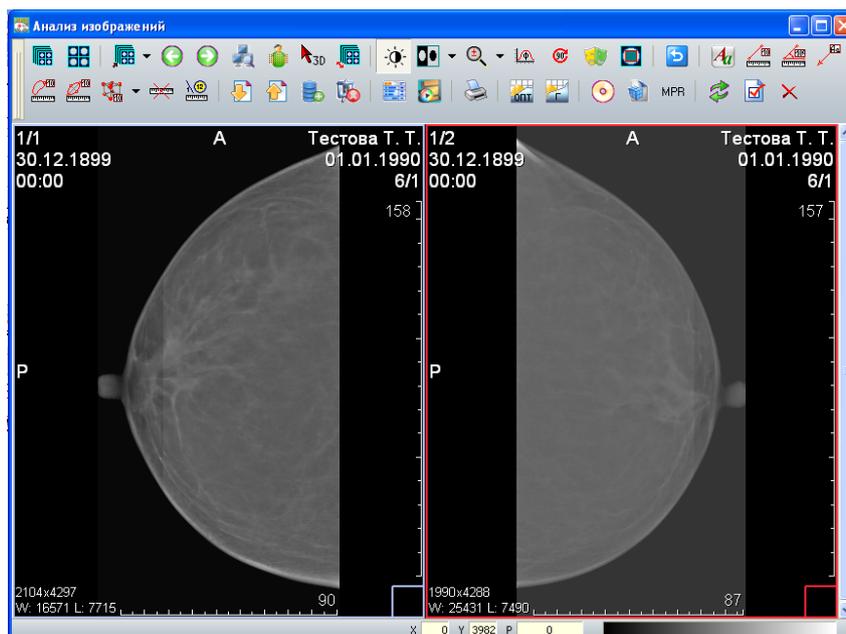


Рис. 5. Рабочее окно врача-консультанта в режиме анализа изображений

Программный продукт специализированного сервера ТМС ЦМ обеспечивает выполнение следующих функций:

- надежное хранение разнородной текстовой и графической информации, предназначенной для формулировки врачебного заключения в рамках выполнения консультации;
- эффективный доступ к информации;
- надежную защиту информации от несанкционированного доступа из локальных и глобальных сетей;
- архивирование данных и эффективный доступ к архивам.

### Заключение

В результате полугодовой эксплуатации ТМС ЦМ в 2012 г. было проведено более 600 сложных телемедицинских консультаций.

Опыт дистанционных диагностических услуг позволяет говорить о неоспоримых преимуществах использования ТМС ЦМ:

- создан уникальный механизм оперативного выявления ряда заболеваний молочной железы;
- обеспечена реализация декларируемой государством общедоступности получения высококвалифицированной специализированной медицинской помощи населением;
- обеспечено ресурсосбережение за счет уменьшения стоимости медицинского скринингового обслуживания при сокращении времени оказания медицинских услуг, экономии средств на транспортные расходы как врача, так и пациента при проведении дистанционных консультаций;
- увеличена эффективность работы медицинского персонала и снижена психологическая нагрузка на пациентов.

Значимость внедрения телемедицинской системы заключается в улучшении качества диагностики и лечения пациентов, что в итоге приводит к усилению социальной защищенности жителей республики.

Использование телемедицинской системы цифровой маммографии обеспечивает прямой экономический эффект, который складывается из следующих аспектов:

- ранняя диагностика и своевременное лечение онкологических заболеваний молочной железы позволяет сократить прямые финансовые затраты на лечебно-диагностический процесс в десятки раз;
- дистанционные консультации являются экономически и организационно эффективным мероприятием ежедневной клинической практики, позволяющим сократить стоимость меди-

цинского обслуживания за счет сокращения времени оказания медицинских услуг, экономии средств на транспортные расходы и уменьшения времени потери трудоспособности.

### Список литературы

1. Михайлов, А.Н. Средства и методы современной рентгенографии : практическое руководство / А.Н. Михайлов. – Минск : Бел. наука, 2000. – 242 с.
2. Кобринский, Б.А. Телемедицина в системе практического здравоохранения / Б.А. Кобринский. – М. : МЦФЭР, 2002. – 176 с.
3. Мушкаев, С. Применение процессора L1879BM1 для сжатия изображений / С. Мушкаев, С. Ландышев // Цифровая обработка сигналов. – 2002. – № 1. – С. 12–18.
4. The LOCO-I Lossless Image Compression Algorithm: Principles and Standardization into JPEG-LS / Hewlett-Packard Laboratories technical report, USA [Electronic resource]. – Mode of access : [www.hpl.hp.com/loco/HPL-98-193R1.pdf](http://www.hpl.hp.com/loco/HPL-98-193R1.pdf). – Date of access : 05.06.2012.

Поступила 31.07.2012

*Объединенный институт проблем  
информатики НАН Беларуси,  
Минск, Сурганова, 6  
e-mail: asc\_med@newman.bas-net.by*

**V.A. Lapitsky, L.N. Lakhanko, A.G. Batrakov, V.V. Kirichenko**

### **DISTRIBUTED TELEMEDICINE SYSTEM OF DIGITAL MAMMOGRAPHY**

The principles of construction and operation of Minsk distributed telemedicine system for carrying out remote consultations on digital mammography are considered. The used version of the storage and transfer of medical images as well as realized functions for a consultant doctor are described.

УДК 004.3

В.В. Анищенко, Л.И. Кульбак

**ОЦЕНКА УЩЕРБА РАБОТОСПОСОБНОСТИ  
ИНФОРМАЦИОННОЙ СИСТЕМЫ**

*Предлагается ввести оценку ущерба информационной безопасности от потери работоспособности информационной системы (объекта). В качестве показателя наносимого ущерба рассчитывается коэффициент возможной (потенциальной) доступности. Показатель доступности определяется методами теории надежности.*

**Введение**

Под безопасностью информации, или информационной безопасностью, принято понимать защищенность информации и поддерживающей инфраструктуры от случайных или преднамеренных воздействий естественного или искусственного характера, способных нанести ущерб владельцам и пользователям информации и поддерживающей ее структуре [1].

Не вызывает сомнения, что потеря работоспособности информационной системы наносит ущерб как владельцам, так и пользователям информации. Под работоспособным состоянием (РС) в соответствии с ГОСТ 27.002 [2] следует понимать состояние объекта, при котором значения всех параметров, характеризующих способность выполнять заданные функции, соответствуют требованиям нормативно-технической и (или) конструкторской (проектной) документации.

Условимся считать, что РС информационной системы обеспечивает возможный доступ пользователей к ее информации. В этом случае наглядным показателем работоспособности объекта может быть доля календарного времени РС объекта. Примем ее в качестве показателя возможной доступности пользователей к информации. Назовем долю календарного времени РС объекта коэффициентом возможной (потенциальной) доступности.

Можно показать, что при условии непрерывной работы комплексный показатель надежности – коэффициент готовности – численно равен коэффициенту возможной доступности.

Согласно [3] коэффициент готовности – это вероятность того, что объект окажется в РС в произвольный момент времени, кроме планируемых периодов, в течение которых применение объекта по назначению не предусматривается.

Календарное время использования объекта можно представить в виде сумм отрезков: времени РС объекта при его использовании по назначению, времени неработоспособного состояния (НРС) при использовании объекта по назначению и времени неиспользования объекта по назначению.

Статистически коэффициент возможной доступности и коэффициент готовности следует определять по формуле [4]

$$K_{\text{вд}} = \frac{T_{\text{РС}}(t_{\text{к}})}{t_{\text{к}}} = K_{\text{Г}}, \quad (1)$$

где  $K_{\text{вд}}$  – коэффициент возможной доступности;  
 $T_{\text{РС}}(t_{\text{к}})$  – суммарное время РС системы (объекта) в интервале календарного времени  $t_{\text{к}}$ ;  
 $t_{\text{к}}$  – интервал календарного времени;  
 $K_{\text{Г}}$  – коэффициент готовности.

Это выражение позволяет для оценки показателя возможной доступности системы (объекта) использовать методы теории и практики надежности.

Рассмотрим риск, связанный с потенциальной доступностью. Согласно СТБ ИСО/МЭК. Руководство 73–2005 [5] риск – это комбинация вероятности наступления вреда и тяжести последствий вреда. В качестве вреда от потери доступности к информации можно

принять потери времени, а в качестве вероятности этого события – вероятность недоступности к информации.

В качестве показателя ущерба можно дополнительно принять суммарное время НРС объекта в течение одного года. Этот показатель связан с коэффициентом возможной доступности формулой

$$t_{НР} = 525\,600(1 - K_{ВД}), \quad (2)$$

где  $t_{НР}$  – суммарное время НРС объекта в течение одного года, мин; 525 600 – количество минут в году.

### 1. Методология оценки коэффициента возможной доступности

Расчет коэффициента доступности информационной системы (объекта) следует начинать с составления структурной схемы доступности (ССД). ССД объекта можно всегда свести к последовательному соединению элементов ССД, которое отражает потерю работоспособности объекта при отказе хотя бы одного элемента (рис. 1). В качестве элементов схемы (ЭС) используются отдельные технические средства (ТС) и группы ТС, имеющие резерв.

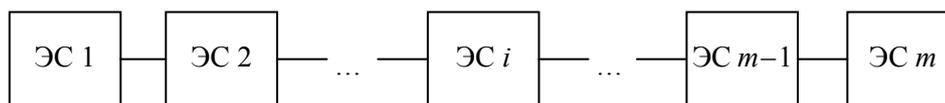


Рис. 1. Типовая ССД объекта

В соответствии с ССД на рис. 1 коэффициент возможной доступности объекта следует вычислять по формуле

$$K_{ВД.об} = \prod_{i=1}^m K_{ВД.i}, \quad (3)$$

где  $K_{ВД.об}$  – коэффициент возможной доступности объекта;  
 $K_{ВД.i}$  – коэффициент возможной доступности  $i$ -го ЭС объекта;  
 $m$  – количество ЭС в ССД объекта.

Среди ЭС объекта будем различать простые и сложные ЭС. Простые ЭС объекта – это отдельные ТС, не имеющие в своем составе резервных ТС. Сложные ЭС объекта – это группа ТС, имеющая в своем составе резервные элементы.

### 2. Расчет коэффициентов возможной доступности ЭС ССД

Любой ЭС в процессе эксплуатации находится в множестве состояний. Состояние ЭС зависит от состояния ее составных частей (СЧ). Состояние СЧ со временем изменяется, и, следовательно, появляется большое число вариантов состояний ЭС за счет комбинации состояний СЧ.

Пусть имеют место следующие допущения [6]:

- ЭС состоит из определенного количества недекомпозируемых элементов ЭС;
- ЭС может находиться в одном из двух состояний: РС или НРС;
- события, переводящие ЭС из одного состояния в другое, случайные и независимые с известными постоянными значениями интенсивности события;
- переход ЭС из одного состояния в другое совершается независимо от предыдущих переходов;
- за исходное состояние ЭС принимается состояние, при котором все элементы ЭС работоспособны.

Модель состояний ЭС принято наглядно представлять в виде размеченного графа. Узлами графа являются состояния, а ребрами со стрелками – пути перехода из одного состояния в

другое. У каждого ребра графа проставляется интенсивность перехода  $\lambda_{i,j}$ , где  $i$  – номер узла, из которого выполняется переход;  $j$  – номер узла, куда выполняется переход.

Пусть множество состояний ЭС  $W$  разделяется на два непересекающихся подмножества: подмножество РС  $W_{PC}$  и подмножество НРС  $W_{NPC}$ . В подмножестве  $W_{PC}$  выделим подмножество состояний, из которых возможен непосредственный переход в подмножество  $W_{NPC}$ , и обозначим его  $W_{PC-NPC}$ . В подмножестве  $W_{NPC}$  выделим подмножество состояний, из которых возможен непосредственный переход в подмножество  $W_{PC}$ , и обозначим его  $W_{NPC-PC}$ .

Коэффициент возможной доступности  $K_{ВД}$  по определению равен вероятности пребывания ЭС в произвольный момент времени в РС. Следовательно, его следует определять по формуле

$$K_{ВД,ЭС} = \sum_{j \in W_{PC}} P_j, \tag{4}$$

где  $P_j$  – вероятность пребывания ЭС в состоянии  $j$  подмножества  $W_{PC}$ .

Для определения вероятностей состояний ЭС, которые фигурируют в формуле (4), строится размеченный граф состояний ЭС, по которому записывается система алгебраических уравнений. При составлении системы уравнений предлагается пользоваться мнемоническим модифицированным правилом [7].

Каждому состоянию (узлу) графа соответствует уравнение. В правой части уравнения записывается произведение вероятности данного состояния и суммы интенсивностей, расставленных на стрелках, выходящих из данного состояния. В левой части уравнения записывается сумма произведений интенсивности перехода по этим стрелкам в рассматриваемое состояние и вероятности состояния, откуда осуществлялся переход. Число членов в правой части уравнения равно числу стрелок, направленных в данное состояние.

Число уравнений должно равняться числу состояний на графе. Из системы уравнений исключается одно из уравнений и вместо него добавляется уравнение нормировки

$$\sum_{j \in W} P_j = 1. \tag{5}$$

Система алгебраических уравнений решается одним из известных способов, а результаты решения подставляются в формулу (4).

Получим формулы для расчета коэффициента доступности ЭС различной структуры.

В нормативной документации по надежности, как правило, принято считать, что наработка на отказ элемента информационной системы имеет экспоненциальный закон распределения. Такой же закон распределения принимается и для времени восстановления.

Простой ЭС (рис. 2) может находиться в одном из двух состояний: РС или НРС. На рис. 2 приняты следующие обозначения:

$S_0$  – исходное РС ЭС;

$S_1$  – НРС ЭС;

$\lambda$  – интенсивность отказов ЭС;

$V$  – интенсивность восстановления ЭС.

Здесь  $\lambda = 1/T_0$ ; (6)

$V = 1/T_B$ , (7)

где  $T_0$  – средняя наработка на отказ ЭС;  
 $T_B$  – среднее время восстановления ЭС.

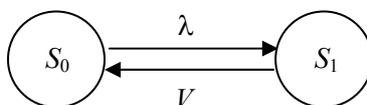


Рис. 2. Граф состояний простого ЭС

Таким образом, согласно графу на рис. 2 система уравнений примет следующий вид:

$$\begin{cases} \lambda P_0 = V P_1; \\ P_0 + P_1 = 1. \end{cases}$$

Корнями этой системы уравнений являются

$$P_0 = \frac{V}{V + \lambda}; \quad (8)$$

$$P_1 = \frac{\lambda}{V + \lambda}. \quad (9)$$

В соответствии с формулой (4) получим

$$K_{\text{вд.эс}} = \frac{V}{V + \lambda} = \frac{T_0}{T_0 + T_B}. \quad (10)$$

Сложный ЭС (например, дублирование) изображен на рис. 3.

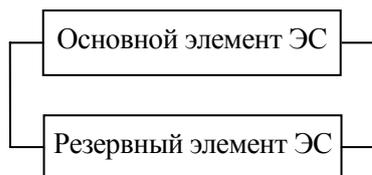


Рис. 3. ССД ЭС с однократным резервированием (дублированием)

Граф состояний резервированного ЭС в случае недостоверной реконфигурации ЭС при отказе резервного элемента ЭС показан на рис. 4.

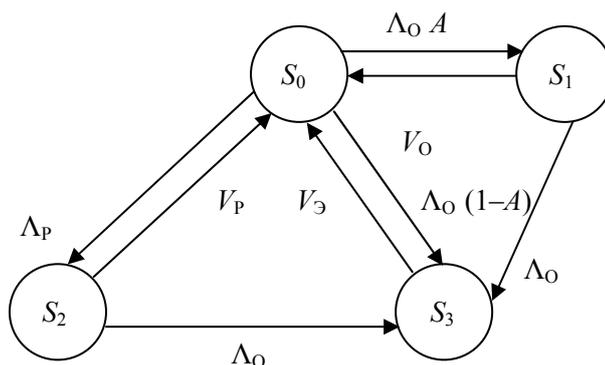


Рис. 4. Граф состояний резервированного ЭС при недостоверной реконфигурации

На рис. 4 приняты следующие обозначения:

$S_0$  – исходное состояние ЭС (основной и резервный элементы ЭС работоспособны): ЭС работоспособен;

$S_1$  – отказал основной элемент ЭС и реконфигурация прошла успешно: ЭС работоспособен;

$S_2$  – отказал резервный элемент: ЭС работоспособен;

$S_3$  – отказал основной элемент ЭС и реконфигурация не прошла успешно: ЭС неработоспособен;

$\Lambda_0$  – интенсивность отказов основного элемента ЭС;  
 $\Lambda_p$  – интенсивность отказов резервного элемента ЭС;  
 $A$  – вероятность успешной реконфигурации;  
 $V_0$  – интенсивность восстановления основного элемента ЭС в обычном случае;  
 $V_э$  – интенсивность восстановления основного элемента ЭС в экстренном случае (когда отказали оба элемента ЭС);  
 $V_p$  – интенсивность восстановления резервного элемента ЭС.  
 При принятых ранее допущениях справедливы следующие соотношения:

$$\Lambda_0 = 1/T_0, \quad (11)$$

где  $T_0$  – средняя наработка на отказ основного элемента ЭС;

$$\Lambda_p = 1/T_{0,p}, \quad (12)$$

где  $T_{0,p}$  – средняя наработка на отказ резервного элемента ЭС;

$$V_0 = 1/T_{B,0}, \quad (13)$$

где  $T_{B,0}$  – среднее время восстановления основного резервированного элемента ЭС в обычном случае;

$$V_э = 1/T_{B,э}, \quad (14)$$

где  $T_{B,э}$  – среднее время восстановления основного резервного элемента ЭС в экстренном случае;

$$V_p = 1/T_{B,p}, \quad (15)$$

где  $T_{B,p}$  – среднее время восстановления резервированного элемента ЭС.

Система уравнений в соответствии с вышеизложенным примет следующий вид:

$$\begin{aligned} (\Lambda_0 + \Lambda_p)P_0 &= V_0P_1 + V_pP_2 + V_эP_3; \\ (V_0 + \Lambda_0)P_1 &= \Lambda_0 AP_0; \\ (V_p + \Lambda_0)P_2 &= \Lambda_p P_0; \\ P_0 + P_1 + P_2 + P_3 &= 1. \end{aligned} \quad (16)$$

Корнями системы уравнений (16) являются

$$P_1 = \frac{\Lambda_0 A}{(V_0 + \Lambda_0)} P_0; \quad (17)$$

$$P_2 = \frac{\Lambda_p}{(V_p + \Lambda_0)} P_0; \quad (18)$$

$$P_3 = \frac{\Lambda_0}{V_э}(P_1 + P_2) + \frac{\Lambda_0(1-A)}{V_э} P_0; \quad (19)$$

$$P_0 = \left[ 1 + \frac{\Lambda_0(1-A)}{V_э} + \left( 1 + \frac{\Lambda_0}{V_э} \right) \left( \frac{\Lambda_0 A}{V_0 + \Lambda_0} + \frac{\Lambda_p}{V_0 + \Lambda_p} \right) \right]^{-1}. \quad (20)$$

Приведем обозначения, принятые в формуле (4), для графа на рис. 4:  $0, 1, 2, 3 \in W$ ,  $0, 1, 2 \in W_{PC}$ ,  $\Lambda_{0,1} = \Lambda_0 A$ ,  $\Lambda_{1,0} = V_0$ ,  $\Lambda_{0,2} = \Lambda_p$ ,  $\Lambda_{0,3} = \Lambda_0(1-A)$ ,  $\Lambda_{2,0} = V_p$ ,  $\Lambda_{3,0} = V_э$ ,  $\Lambda_{2,3} = \Lambda_0$ ,  $\Lambda_{1,3} = \Lambda_0$ .

В соответствии с формулой (4)

$$K_{\text{ВД.ЭС}} = P_0 + P_1 + P_2. \quad (21)$$

После подстановки в формулу (21) формул (17), (18) и (20) получим

$$K_{\text{ВД.Р.ЭС}} = \frac{1 + B_0 + B_p}{1 + \Lambda_0(1 - A)T_{\text{В.Э}} + (1 + \Lambda_0)T_{\text{В.Э}}(B_0 + B_p)}; \quad (22)$$

$$B_0 = \frac{\Lambda_0 A T_{\text{В.О}}}{1 + \Lambda_0 T_{\text{В.О}}}; \quad (23)$$

$$B_p = \frac{\Lambda_p T_{\text{В.Р}}}{1 + \Lambda_0 T_{\text{В.Р}}}, \quad (24)$$

где  $K_{\text{ВД.Р.ЭС}}$  – коэффициент возможной доступности резервированного ЭС.

Дробное (скользящее) резервирование ЭС изображено на рис. 5.

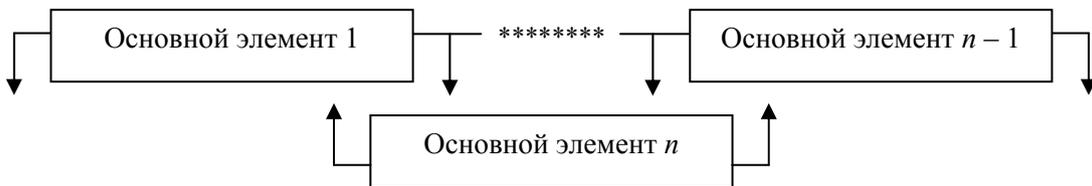


Рис. 5. ССД ЭС с дробным резервированием  $1/n$  (со скользящим резервом)

Граф состояний резервного ЭС в случае достоверной реконфигурации СЧ при отказе резервных элементов ЭС и при дробном резервировании изображен на рис. 6.

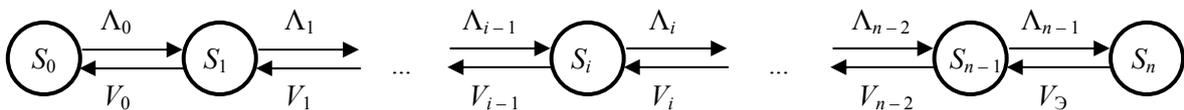


Рис. 6. Граф состояний резервированного ЭС

На рис. 6 приняты следующие обозначения:

$S_0$  – исходное состояние ЭС, когда все  $n$  элементов ЭС работоспособны;

$S_i$  – состояние ЭС, когда отказали  $i$  из  $n$  элементов ЭС;

$S_n$  – состояние ЭС, когда отказали все  $n$  элементов СЧ (предельное число отказавших элементов: ЭС неработоспособно);

$\Lambda_i, i = 0, \dots, n - 1$ , – интенсивность перехода ЭС из состояния  $S_i$  в состояние  $S_{i+1}$ ;

$V_i, i = 0, \dots, n - 2$ , – интенсивность перехода ЭС из состояния  $S_{i+1}$  в состояние  $S_i$ ;

$V_n$  – интенсивность перехода ЭС из состояния  $S_n$  в состояние  $S_{n-1}$ ;

$n$  – предельное число отказавших ЭС.

В соответствии с правилами составления системы алгебраических уравнений по размеченному графу состояний, изложенными выше, и с учетом замены одного уравнения системы уравнений на уравнение нормировки система уравнений примет следующий вид:

$$\begin{aligned} \Lambda_0 P_0 &= V_0 P_1; \\ (V_{i-1} + \Lambda_i) P_i &= \Lambda_{i-1} P_{i-1} + V_i P_{i+1}, \quad i = 1, \dots, n - 2; \end{aligned} \quad (25)$$

$$(V_{n-2} + \Lambda_{n-1})P_i = \Lambda_{n-2}P_{n-2} + V_{\text{Э}}P_n, i = 1, \dots, n-2;$$

$$\sum_{i=0}^n P_i = 1,$$

где  $P_i$  – вероятность пребывания ЭС в состоянии  $S_i$ .

Корнями системы уравнений (25) являются

$$P_0 = [1 + \frac{\Lambda_{n-1}}{V_{\text{Э}}} \prod_{j=0}^{n-2} \frac{\Lambda_j}{V_j}]^{-1}; \quad (26)$$

$$P_i = P_0 \prod_{j=0}^{i-1} \frac{\Lambda_j}{V_j}, \quad i = 1, \dots, n-1; \quad (27)$$

$$P_n = P_0 \frac{\Lambda_{n-1}}{V_{\text{Э}}} \prod_{j=0}^{i-1} \frac{\Lambda_j}{V_j}. \quad (28)$$

Для ЭС справедливо соотношение

$$\Lambda_i = (n-i)\Lambda, \quad i = 0, \dots, n-1, \quad (29)$$

где  $\Lambda$  – интенсивность отказов ЭС;

$i$  – количество накопившихся отказавших ЭС.

При неограниченном восстановлении ЭС выполняются следующие условия:

$$V_i = (i+1)V, \quad i = 0, \dots, n-2, \quad V_{\text{Э}} > nV, \quad (30)$$

где  $V$  – интенсивность восстановления отказавшего ЭС;

$i$  – количество накопившихся отказавших ЭС;

$V_{\text{Э}}$  – интенсивность экстренного восстановления отказавшего ЭС.

С учетом соотношений (29), (30) и формул (26)–(28) получим

$$K_{\text{вд.р.ЭС}}^* = P_0 (1 + \sum_{i=1}^{n-1} A^i \frac{n!}{(n-i)!i!}), \quad (31)$$

где

$$P_0 = [1 + \sum_{i=1}^{n-1} A^i \frac{n!}{(n-i)!i!} + \frac{VA^n}{V_{\text{Э}}}]^{-1}. \quad (32)$$

### 3. Пример расчета

Произведем расчет доступности в системе высокой готовности, выполненной в виде кластера [8, 9].

Кластер состоит из двух узлов (серверов), связанных между собой внутренней сетью, которая обеспечивает обмен сигналами активности. Узлы подключены к общему дисковому массиву (RAID-массиву) и к внешней локальной сети. Все подсистемы кластера имеют резервирование, поэтому при отказе любого элемента кластер в целом останется в РС. Более того, замена отказавшего элемента возможна без остановки кластера. На обоих узлах кластера устанавливается операционная система Microsoft Windows Server 2003 Enterprise, которая поддерживает технологию Microsoft Windows Cluster Service (MSCS).

Принцип работы кластера следующий. Приложение (служба), доступность которого обеспечивается кластером, устанавливается на обоих узлах. Для этого приложения создается группа ресурсов, включающая IP-адрес и сетевое имя виртуального сервера, а также один или несколько логических дисков на общем дисковом массиве. Таким образом, приложение вместе со своей группой ресурсов не привязывается жестко к конкретному узлу, а напротив, может

быть запущено на любом из этих узлов (при этом на каждом узле одновременно может работать несколько приложений).

Описанной системе соответствует ССД (рис. 7).

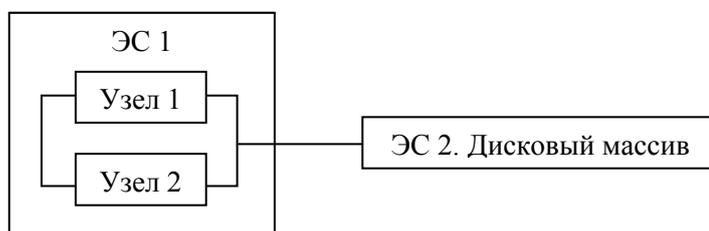


Рис. 7. ССД кластера

Примем следующие значения показателей надежности компонент кластера:

- средняя наработка на отказ узла (сервера)  $T_{O,C} = 65\,000$  ч;
- среднее время восстановления узла (сервера) в обычном случае  $T_{B,O,C} = 8$  ч;
- среднее время восстановления узла (сервера) в экстренном случае  $T_{B,\Delta,C} = 2$  ч;
- вероятность перехода на резерв  $A = 1,0$ ;
- средняя наработка на отказ дискового массива  $T_{O,D} = 100\,000$  ч;
- среднее время восстановления дискового массива  $T_{B,D} = 4$  ч.

В соответствии с формулой (4)

$$K_{ВД,к} = K_{ВД,1} K_{ВД,2}, \quad (33)$$

где  $K_{ВД,к}$  – коэффициент возможной доступности кластера;

$K_{ВД,1}$  – коэффициент возможной доступности ЭС 1;

$K_{ВД,2}$  – коэффициент возможной доступности ЭС 2.

ЭС 1 является непростым элементом ССД, и расчет показателя  $K_{ВД,1}$  следует проводить по формулам (11), (12), (14), (22)–(24). В результате расчета  $K_{ВД,1} = 0,999805$ . ЭС 2 является простым элементом ССД и в результате расчета по формуле (10)  $K_{ВД,2} = 0,999960$ . В результате расчета по формуле (33)  $K_{ВД,к} = 0,999764$ .

Суммарное время НРС объекта в течение одного года, вычисленное по формуле (2), составило 124 мин.

### Заключение

В работе обосновывается целесообразность ввода в нормативную документацию по информационной безопасности риска в виде коэффициента возможной (потенциальной) доступности и ущерба от недоступности в виде суммарного времени НРС объекта в течение одного года.

Введение такого рода риска в нормативную документацию будет способствовать повышению качества разработки информационных систем в части надежности.

Предлагается методология для расчета риска, которая иллюстрируется примером.

### Список литературы

1. Информационные технологии безопасности и защиты [Электронный ресурс]. – 2012. – Режим доступа : <http://infdis.narod.ru/it/5-6/n5.htm>. – Дата доступа : 13.03.2012.
2. Доступность информации // Википедия [Электронный ресурс]. – 2012. – Режим доступа : <http://ru.wikipedia.org>. – Дата доступа : 13.03.2012.
3. Надежность в технике. Основные понятия. Термины и определения : ГОСТ 27.002–89. – М. : Изд-во стандартов, 1990. – 37 с.
4. Половко, А.М. Основы теории надежности / А.М. Половко, С.В. Гуров. – СПб. : БХВ-Петербург, 2006. – 620 с.

5. Менеджмент риска. Термины и определения. СТБ ИСО/МЭК Руководство 73–2005. – Минск : Госстандарт, 2005. – 12 с.
6. Анищенко, В.В. Надежность и отказоустойчивость кластерных вычислительных систем / В.В. Анищенко, Л.И. Кульбак, В.К. Фисенко // Автоматика и вычислительная техника. – 2004. – № 5. – С. 32–42.
7. Овчаров, Л.А. Прикладные задачи теории массового обслуживания / Л.А. Овчаров. – М. : Машиностроение, 1969. – 324 с.
8. Сравнение кластера надежности и «обычного» сервера. Компания «Тим Компьютерс» [Электронный ресурс]. – 2012. – Режим доступа : [http://www.team.ru/server/stbl\\_compare.shtml](http://www.team.ru/server/stbl_compare.shtml). – Дата доступа : 13.03.2012.
9. Анищенко, В.В. Модели надежности кластерных вычислительных систем / В.В. Анищенко, Л.И. Кульбак, Т.С. Мартинович // Известия НАН Беларуси. Сер. физ.-техн. наук. – 2008. – № 1. – С. 89–99.

Поступила 13.03.12

*Объединенный институт проблем  
информатики НАН Беларуси,  
Минск, Сурганова, 6  
e-mail: lkulbak@yandex.ru*

**U.V. Anishchanka, L.I. Kulbak**

**ASSESSMENT OF LOSS OF INFORMATION SYSTEM  
PROCESSING CAPABILITY**

It is proposed to introduce an assessment of information security damage from processing capability loss of an information system (object). As an indicator of damage it is proposed to use a coefficient of possible (potential). The indicator is determined by the reliability theory methods.

УДК.681.2

Г.В. Фролов

## СИСТЕМНЫЙ ПОДХОД К ЗАЩИТЕ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

*Предлагается модель системы защиты информационных технологий (ИТ) с использованием сервисов безопасности. Модель выбрана на основе критериев оценки качества систем: критерия пригодности, критерия оптимальности и критерия превосходства на основе предложенной классификации угроз безопасности. Предложенная модель представляет собой совокупность сервисов безопасности, которые могут быть реализованы в среде функционирования ИТ, в самой ИТ или средствами защиты информации.*

## Введение

Решение задачи создания защищенной ИТ требует от пользователя построения модели угроз для реальной ИТ. Выстраивание же адекватной модели возможно при использовании базы данных угроз безопасности ИТ, содержащей информацию о множестве угроз безопасности ИТ, известных на момент построения защищаемой системы. В настоящее время разработан и поддерживается, в том числе и международными специализированными организациями, ряд таких баз данных. Сравнение содержащихся в существующих базах данных решений показывает значительную разницу как в подходах к классификации угроз, так и в полученных результатах. Необходимо отметить и один общий недостаток – отсутствие системного подхода к решению поставленных задач [1]. Ни в одной из известных баз данных не были использованы системные принципы построения классификации угроз.

## 1. Выбор варианта реализации системы приемлемого качества

Для выбора варианта реализации системы приемлемого качества применим методологию оценки качества систем.

Критерии оценки качества систем можно разделить на три класса: пригодности, оптимальности и превосходства.

Пусть  $\pi_{ij}$  ( $i \in \{1, \dots, m\}, j \in \{1, \dots, n\}$ ) – показатель  $i$ -го свойства  $j$ -й системы, т. е. показатель качества  $j$ -й системы есть вектор  $\Pi_j = \{\pi_{1j}, \pi_{2j}, \dots, \pi_{mj}\}$ ,  $\{\pi_{ij}^{(don)}\}$  – множество допустимых значений показателя  $\pi_{ij}$ . Тогда критерии перечисленных выше классов можно представить следующим образом:

1. Критерий пригодности:

$$K_{\text{приг}} : \left( \bigcup_{i \in M} \{\pi_{ij}\} \right) \cap \left( \bigcup_{i \in M} \{\pi_{ij}^{(don)}\} \right),$$

где  $M = \{1, \dots, m\}$  – множество всех свойств.

Данное выражение имеет следующий смысл: система  $j$  является пригодной по качеству, если значения всех показателей существенных свойств принадлежат допустимым значениям. По этому определению системы, для которых выполняются условия данного выражения, обладают одинаковыми качествами.

2. Критерий оптимальности:

$$K_{\text{опт}} : \left[ \left( \bigcup_{i \in M \setminus M_0} \{\pi_{ij}\} \right) \cap \left( \bigcup_{i \in M \setminus M_0} \{\pi_{ij}^{(don)}\} \right) \right] \cup \left[ \left( \bigcup_{i \in M_0} \{\pi_{ij}\} \right) \cap \left( \bigcup_{i \in M_0} \{\pi_{ij}^{(opt)}\} \right) \right],$$

где  $M_0$  – подмножество оптимальных свойств;

$\pi_{ij}^{(opt)}$  – оптимальное значение показателя  $i$ -го свойства  $j$ -й системы.

Данное выражение имеет следующий смысл: система  $j$  является оптимальной по  $m_0$  свойствам, если она пригодна по остальным свойствам и показатели свойств из множества  $M_0$  принимают оптимальные значения.

3. Критерий превосходства:

$$K_{прев} : [(\bigcup_{i \in M} \{\pi_{ij}\}) \cap (\bigcup_{i \in M} \{\pi_{ij}^{don}\})] \cap (\bigcup_{i \in M} \{\pi_{ij} \mid \pi_{ij} \succ \pi_{il}, l \in M, l \neq j\}).$$

Данное выражение имеет следующий смысл: система  $j$  превосходит (обозначение  $\succ$ ) по качеству остальные пригодные системы, если значения показателей ее свойств превосходят либо не хуже значений соответствующих показателей остальных систем.

Если  $\pi_{il} = \pi_{ij}$ ,  $i = 1, \dots, m$ , то качества  $l$ -й и  $j$ -й систем признаются одинаковыми. Если же хотя бы одно из условий данного выражения не выполняется, то это означает, что заданная совокупность показателей свойств не позволяет выявить систему, превосходящую по качеству все остальные.

Для оценки качества систем необходимо выделить свойства систем (точнее, существенные свойства), отсутствие которых приводит к потере того качества, которое связывалось с безопасностью ИТ. По показателям этих свойств производится оценка качества. Показатель – количественное или качественное выражение свойств системы.

В данной работе предлагается рассматривать следующие существенные свойства:

- типизацию;
- полноту;
- реализуемость.

Для характеристики свойств системы будут использованы показатели, определяющие наличие (отсутствие) существенных свойств.

Одним из вариантов реализации системы безопасности на основе технологии «перекрытия угроз» является непосредственная нейтрализация самих угроз. Это предполагает построение модели угроз, содержащей сведения о всех возможных угрозах (рис. 1).

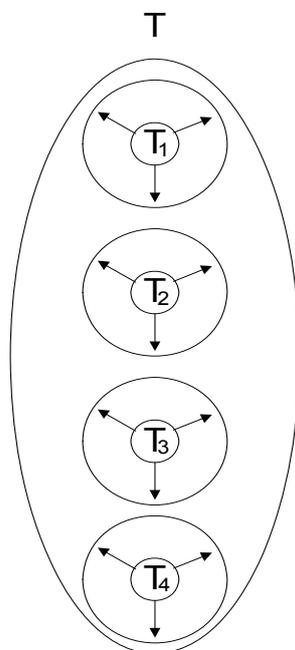


Рис. 1. Модель системы защиты по нейтрализации угроз:  $T_i$  – угроза  $i$ , круг означает операцию нейтрализации

При построении модели угроз разработчик может опираться на нормативные документы, определяющие методологию описания угроз, либо воспользоваться разработанными и поддерживаемыми, в том числе и международными специализированными организациями, базами данных угроз безопасности информации [2–4].

В первом случае количество угроз и их формулировки будут индивидуальными. Во втором случае из-за разницы подходов к классификации и идентификации угроз, которые используются при разработке баз данных, результаты их применения в каждой конкретной ситуации будут различны, и в этой ситуации нельзя говорить о типовом элементе при построении такой модели системы защиты ИТ.

С одной стороны, при отсутствии нормативно-технической базы, содержащей полный перечень угроз безопасности информации, нет гарантии полноты модели угроз в каждом конкретном случае. С другой стороны, разработчик может включить в модель такое количество угроз, что система защиты станет нереализуемой.

На основании изложенного можно сделать вывод, что вариант реализации системы защиты по принципу нейтрализации угроз является непригодным по качеству, так как в нем отсутствуют свойства типизации и будет отсутствовать либо полнота, либо реализуемость.

Другой вариант реализации системы защиты информации на основе технологии «перекрытия угроз» связан с защитой объектов. В этом случае необходимо построить модель защиты от всех угроз для каждого объекта (рис. 2). При этом разработчик неограничен как в степени детализации объектов, так и в возможности создания новых объектов и выделить типовой элемент защиты не представляется возможным.

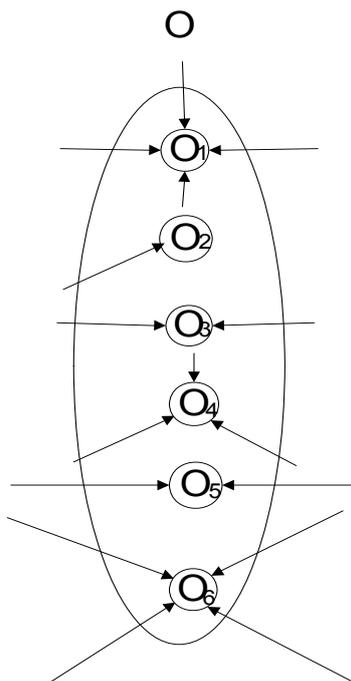


Рис. 2. Модель системы защиты объектов от угроз:  $O_i$  – объект  $i$

В отсутствие ограничения на число объектов и угроз для каждого объекта система защиты может оказаться практически нереализуемой. Реализация системы защиты только для определенной части объектов и угроз приводит к неполноте защиты. Отсюда можно сделать вывод, что, как и в предыдущем случае, рассматриваемые системы защиты не являются качественными по критерию пригодности.

Третий вариант построения системы защиты связан с полным перекрытием угроз (рис. 3). В этом случае определяется множество угроз и объектов и устанавливаются взаимосвязи (пути доступа) для каждой пары «угроза – объект». Каждый идентифицированный путь доступа угро-

зы к объекту должен быть перекрыт системой защиты. В связи с тем что множество угроз и объектов неопределено, нельзя говорить о типизации таких систем безопасности.

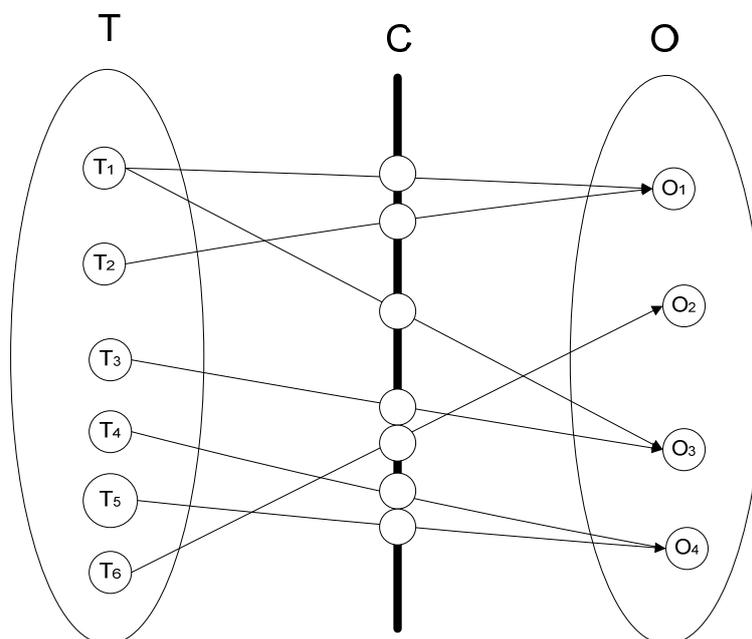


Рис. 3. Модель системы защиты с полным перекрытием угроз: С – перекрытие путей доступа угрозы к объекту

Наличие слишком большого числа взаимосвязей «угроза – объект» в данной модели приведет к ее нереализуемости. Если ограничиться рассмотрением только части взаимосвязей, система защиты будет неполной.

Следствием неопределенности числа угроз и объектов в модели защиты с полным перекрытием является невозможность типизации таких систем защиты и отсутствие реализуемости или полноты защиты. Построенные на основе данной модели системы защиты также не будут пригодными.

Однако для модели систем защиты с полным перекрытием угроз можно достичь определенности объектов защиты и определенности угроз, а при предложенном уровне детализации объектов – полноты защиты. В этом случае обеспечиваются все качественные характеристики системы: типизация, полнота и реализуемость.

## 2. Классификация угроз безопасности

Описание защищенного состояния системы ИТ  $S$  можно представить в виде следующей схемы:

$$S_{ijnm} = \{O_i, T_{ij}, C_{ijn}, F_{ijnm}\},$$

где  $O_i$  – множество структурных компонентов ИТ;

$T_{ij}$  – множество угроз  $i$ -му структурному компоненту ИТ;

$C_{ijn}$  – множество сервисов безопасности, противодействующих  $j$ -й угрозе;

$F_{ijnm}$  – множество требований по реализации  $n$ -го сервиса безопасности, противодействующего  $j$ -й угрозе на  $i$ -й структурный компонент ИТ.

Схема  $S_{ijnm}$  может быть достигнута после выполнения цепочки процедур, которые можно представить в виде операторов построения базы данных угроз ( $\{T_j, DBJ\} \Rightarrow B_j$ ), построения модели угроз ( $\{S_i, B_j\} \Rightarrow S_{ij}$ ), разработки системы мер по защите ( $\{S_{ij}, B_n\} \Rightarrow S_{ijn}$ ), построения профиля защиты ( $\{S_{ijn}, B_m\} \Rightarrow S_{ijnm}$ ). Здесь  $DBJ$  – инструментарий разработки БД, который должен позволять реализовать БД угроз на основе заданной классификации угроз  $T_j$ ;  $B_j$  – база данных угроз безопасности;  $B_n$  – база данных мер противодействия угрозам (сервисов безопасности);

$B_m$  – база данных требований безопасности;  $S_i$  – множество структурных компонентов системы;  $S_{ij}$  – модель угроз;  $S_{ijn}$  – система мер противодействия.

Задача построения защищенной ИТ может быть выражена в виде оператора

$$\{S_i, B_j, B_n, B_m, Z\} \Rightarrow S_{ijnm},$$

где  $Z$  – множество задач, которое должно быть решено для того, чтобы реализовать мероприятия по защите.

Каждую угрозу можно рассматривать как процедуру, осуществление которой приводит к ущербу ИТ. Ущерб предлагается выражать в категориях конфиденциальности, целостности и доступности. Кроме того, угрозы можно рассматривать как проявление факторов, воздействующих на защищаемую информацию по ГОСТ Р 51275–99 [5]. Согласно данному стандарту фактор – явление, действие, процесс, результатом которых может быть утечка, искажение, уничтожение, блокирование доступа к информации.

Задача построения таксономии угроз может быть выражена в виде оператора

$$\{S_i, K_{il}, Fa_y\} \Rightarrow T_j,$$

где  $K_{il}$  –  $l$ -я категория ущерба  $i$ -го структурного компонента ИТ;

$Fa_y$  – фактор, воздействующий на информацию.

Данное выражение позволяет сформулировать следующие принципы таксономии угроз:

1. Таксономия угроз должна соответствовать множеству структурных компонентов ИТ.
2. Таксономия угроз должна соответствовать категориям ущерба.
3. Таксономия угроз должна соответствовать таксономии факторов, воздействующих на защищаемую информацию.

Очевидно, что такое представление угроз закладывает возможность при реализации системы обеспечения безопасности ИТ решать задачи контроля:

- возможностей системы защиты по противодействию угрозам на структурные компоненты ИТ;
- возможного ущерба при реализации угроз;
- возможностей системы защиты по противодействию негативному влиянию факторов, действующих на объект информатизации.

### 3. Разработка баз данных сервисов безопасности и требований безопасности

Задача разработки баз данных сервисов безопасности и требований безопасности может быть сформулирована в виде операторов

$$\{T_n, DBN\} \Rightarrow B_n;$$

$$\{T_m, DBM\} \Rightarrow B_m,$$

где  $T_n$  и  $T_m$  – заданные классификации сервисов и требований безопасности;

$DBN$  и  $DBM$  – инструментарии разработки баз данных.

Принципы таксономии сервисов и требований безопасности аналогичны принципам таксономии угроз. Для требований безопасности дополнительно сформулирован еще один принцип таксономии: таксономия требований безопасности должна соответствовать множеству сервисов безопасности. В приложении к разработке базы данных требований безопасности этот принцип означает, что каждому сервису безопасности должно соответствовать множество требований по его реализации.

В качестве инструментариев при разработке баз данных использовались нормативно-методологические документы в области защиты информации [6–11].

В результате проведенных в рамках данной работы исследований был сформулирован следующий принцип: классификация угроз безопасности ИТ должна соответствовать множеству структурных компонентов ИТ и предусматривать наличие атрибутов угроз, указывающих на те структурные компоненты ИТ, в которых угроза может быть идентифицирована.

#### 4. Модель системы защиты ИТ

Таким образом, система защиты ИТ (рис. 4) представляет собой совокупность сервисов безопасности, которые могут быть реализованы:

- в среде функционирования ИТ;
- в самой ИТ (путем использования защитных свойств ИТ);
- средствами защиты информации.

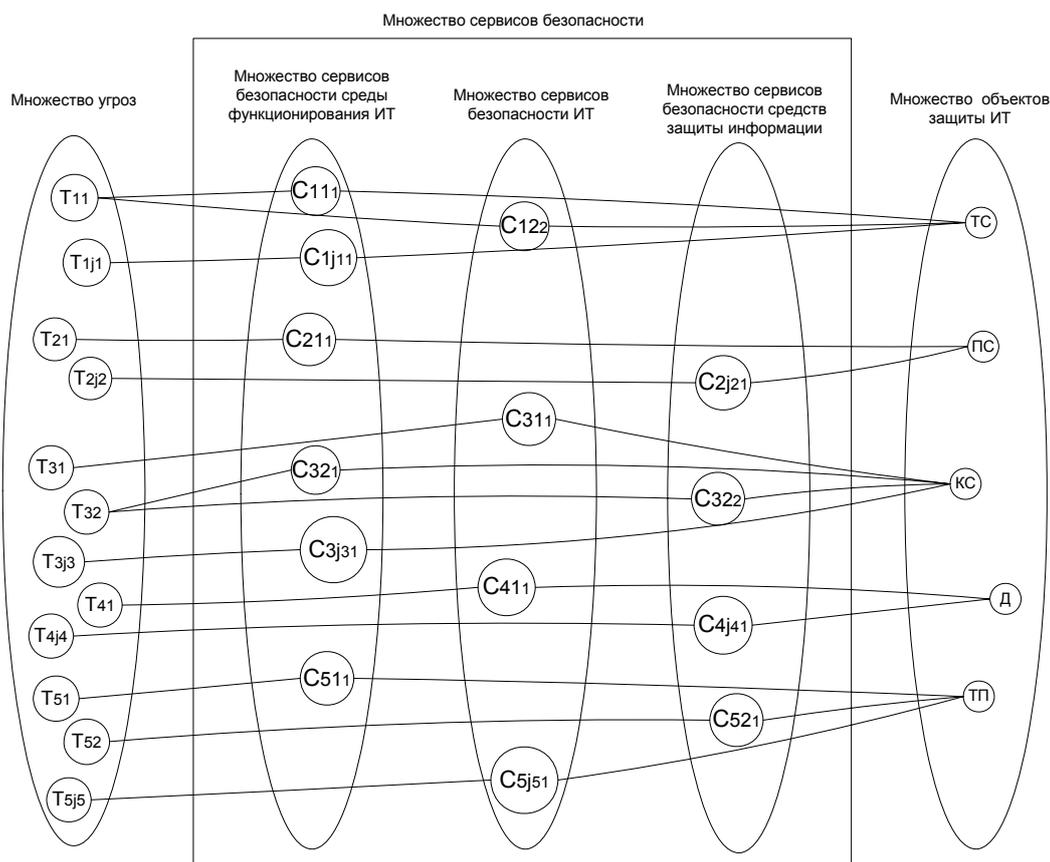


Рис. 4. Модель системы защиты, содержащей три подмножества сервисов безопасности

Модель системы обеспечения безопасности представляет собой трехслойный граф

$$S = \{T, O, C\},$$

где  $T$  – множество угроз,  $T = \{T_i\}$ ;

$O$  – множество объектов защищаемой системы,  $O = \{O_j\}$ ;

$C$  – множество сервисов безопасности,  $C = \{C_k\}$ .

Множество  $O$  разбито на взаимно непересекающиеся подмножества

$$O = \{O_1, O_2, O_3, O_4, O_5\},$$

где  $O_1$  – технические средства обработки информации;

$O_2$  – программные средства обработки информации;

$O_3$  – каналы связи;

$O_4$  – данные;

$O_5$  – технологический процесс.

Множество угроз  $T$  разбито на взаимно непересекающиеся подмножества

$$T_1 = \{T_{11}, T_{12}, \dots, T_{1j1}\};$$

$$T_2 = \{T_{21}, T_{22}, \dots, T_{2j2}\};$$

$$T_3 = \{T_{31}, T_{32}, \dots, T_{3j3}\};$$

$$T_4 = \{T_{41}, T_{42}, \dots, T_{4j4}\};$$

$$T_5 = \{T_{51}, T_{52}, \dots, T_{5j5}\},$$

где  $T_1$  – конечное множество угроз техническим средствам обработки информации;

$T_2$  – конечное множество угроз программным средствам обработки информации;

$T_3$  – конечное множество угроз каналам связи;

$T_4$  – конечное множество угроз данным;

$T_5$  – конечное множество угроз технологическому процессу.

Каждую угрозу  $T_{ij}$  из множества угроз перекрывает соответствующее множество сервисов безопасности:

$$\forall_{(T_{ij}) \in T} \exists_{(C_{ij1}, C_{ij2}, \dots, C_{ijn}) \in C},$$

где  $n$  – количество сервисов безопасности, противодействующих угрозе  $T_{ij}$ .

Это условие означает, что каждой угрозе  $T_{ij}$  из множества угроз  $T$  соответствует подмножество сервисов безопасности  $\{C_{ij1}, C_{ij2}, \dots, C_{ijn}\}$  множества сервисов  $C$ .

## 5. Реализация модели системы защиты ИТ

На основе предложенной модели разработана автоматизированная система разработки профиля защиты и задания по безопасности. Были созданы 12 таблиц справочных баз данных: «Объекты защиты», «Угрозы», «Сервисы безопасности», «Предложения безопасности», «Требования безопасности», «Правила политики безопасности», «Параметры правил политики безопасности», «Типовые прикладные решения на основе сервисов безопасности», «Правила политики безопасности, реализуемые типовыми прикладными решениями на основе сервисов безопасности», «Параметры сервисов безопасности типовых прикладных решений», «Оценочные уровни доверия», «Требования доверия».

Работа автоматизированной системы основана на использовании графического последовательного интерфейса, работа с которым заключается в определении состава структурных объектов ИТ.

После подключения справочной базы данных пользователь определяет состав структурных объектов ИТ, воспользовавшись элементом управления типа «флажок», расположенным в верхней части окна «Система поддержки принятия решения». Установка или снятие «флажка», соответствующего какому-либо объекту защиты ИТ, запускает механизм выполнения SQL-запроса к справочной базе данных. В запросе принимают участие таблицы базы данных, содержащие информацию об объектах защиты ИТ, угрозах, сервисах безопасности и правилах политики безопасности организации. Запрос осуществляет выборку объектов защиты ИТ и проверку, установлен ли для данного объекта «флажок» включения его в профиль защиты.

Использование автоматизации разработки профиля защиты задания по безопасности защищенных ИТ с применением типовых решений позволяет наглядно и эффективно представлять и манипулировать большими объемами данных, необходимыми для выполнения этапов разработки защищенных ИТ (рис. 5–8). Этапы разработки представляются в удобной графической форме. Такая общая визуализация процессов обеспечения защищенности ИТ позволяет оперативно генерировать различные варианты защиты, сравнивать их между собой с точки зрения эффективности и в результате выбирать оптимальный вариант построения или модификации системы защиты ИТ.

Мастер построения профиля защиты и задания по безопасности ИТ. Шаг 1: Описание объекта.

Проект

Наименование объекта защиты  
Система компьютерная специальная "Аккорд"

КСС Аккорд для автоматизации обработки данных при реализации товаров и услуг, для регистрации, учета, накопления и сохранения этих данных, для выдачи финансовым

Функциональные возможности ОО

Объекты защиты ОО

Сбор информации  
 Накопление информации  
 Ввод информации  
 Вывод информации  
 Прием информации  
 Передача информации  
 Запись информации  
 Хранение информации  
 Регистрация информации  
 Уничтожение информации  
 Преобразование информации  
 Отображение информации

Технические средства  
 Программные средства  
 Каналы связи  
 Технологический процесс  
 Данные

Рис. 5. Определение функциональных возможностей и состава объекта

Мастер построения профиля защиты и задания по безопасности ИТ. Шаг 2: Описание свойств среды функционирования объекта.

Проект

Предположения безопасности

№	Наименование
<input checked="" type="checkbox"/>	1 Средой функционирования ИТ осуществляется контроль физического доступа пользователей к техническим средствам обработки информации
<input type="checkbox"/>	2 Средой функционирования ИТ осуществляется контроль целостности технического состава средств обработки информации
<input type="checkbox"/>	3 Средой функционирования ИТ осуществляется регистрация действий пользователей по доступу к техническим средствам обработки информации
<input type="checkbox"/>	4 Средой функционирования ИТ осуществляется реакция на попытки нарушения целостности технического состава средств обработки информации
<input type="checkbox"/>	5 Средой функционирования ИТ осуществляется управление правами пользователей на выполнение функций администрирования технических средств
<input type="checkbox"/>	6 Средой функционирования ИТ осуществляется регистрация действий пользователей по выполнению функций администрирования технических средств
<input type="checkbox"/>	7 Средой функционирования ИТ осуществляется реакция на попытки несанкционированного выполнения пользователями функций администрирования
<input type="checkbox"/>	8 Средой функционирования ИТ осуществляется периодическое тестирование технических средств обработки информации
<input type="checkbox"/>	9 Средой функционирования ИТ осуществляется аудит функционирования технических средств обработки информации
<input type="checkbox"/>	10 Средой функционирования ИТ осуществляется восстановление функционирования системы после сбоя технических средств обработки информации
<input type="checkbox"/>	11 Средой функционирования ИТ осуществляется обеспечение способности выполнения автоматизированной системой своих функциональных возмож.
<input type="checkbox"/>	12 Средой функционирования ИТ осуществляется реакция на сбой технических средств обработки информации
<input type="checkbox"/>	13 Средой функционирования ИТ осуществляется управление правами пользователей и процессов на создание новых объектов, содержащих исполне..
<input type="checkbox"/>	14 Средой функционирования ИТ осуществляется контроль доступа пользователей и процессов к исполняемым файлам программ
<input type="checkbox"/>	15 Средой функционирования ИТ осуществляется контроль целостности программных средств обработки информации
<input type="checkbox"/>	16 Средой функционирования ИТ осуществляется регистрация действий пользователей по модификации программных средств обработки информации
<input type="checkbox"/>	17 Средой функционирования ИТ осуществляется реакция на попытки несанкционированной модификации программных средств обработки информации
<input type="checkbox"/>	18 Средой функционирования ИТ осуществляется контроль прав пользователей и процессов на запуск программ
<input type="checkbox"/>	19 Средой функционирования ИТ осуществляется регистрация запуска программ пользователями и процессами

Рис. 6. Описание свойств среды функционирования объекта

Мастер построения профиля защиты и задания по безопасности ИТ. Шаг 3: Формирование политики безопасности.

Проект

Предварительный профиль защиты

Объект защиты	Технические средства ИТ
Угроза	Несанкционированная модификация технического состава средств обработки информации в результате физического доступа
Сервис безопасности среды ИТ	Действия в случае обнаружения возможного нарушения безопасности.
Сервис безопасности ИТ	Генерация данных аудита.
Сервис безопасности ИТ	Анализ потенциального нарушения
Сервис безопасности ИТ	Просмотр аудита
Сервис безопасности ИТ	Обнаружение физического нападения
Угроза	Несанкционированная модификация конфигурации и параметров функционирования технических средств обработки информации п
Сервис безопасности среды ИТ	Действия в случае обнаружения возможного нарушения безопасности.

Правила политики безопасности

Правила, которые следует использовать для анализа журнала аудита  
Совокупность событий, подвергаемых аудиту, проявление которых (каждого в отдельности или совместно)

Параметры политики безопасности

модификация технического состава средств обработки инк  
 модификация конфигурации и параметров функционирова

Рис. 7. Формирование политики безопасности

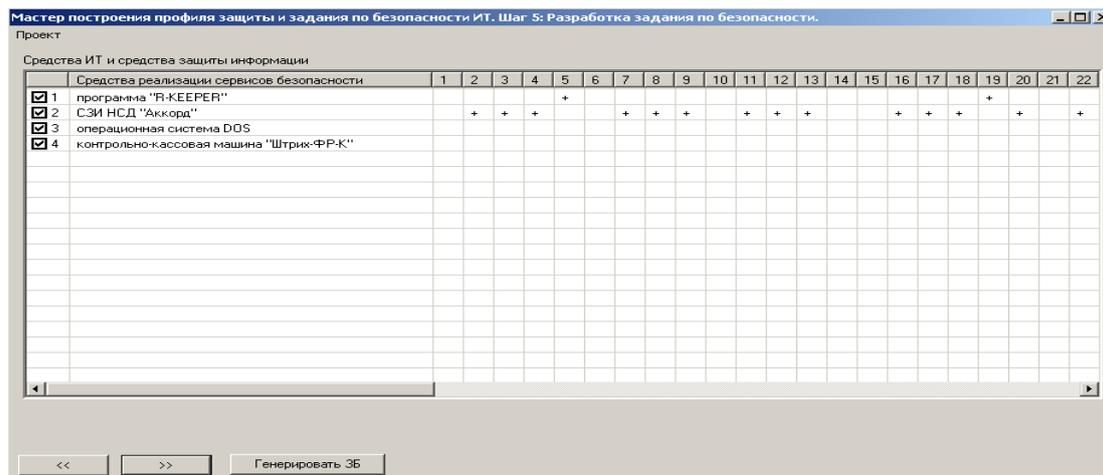


Рис. 8. Разработка задания по безопасности

### Заключение

Предлагаемая модель обеспечивает полное перекрытие угроз: каждому классу объектов соответствует весь перечень возможных угроз как для всего класса объектов, так и для каждого объекта в классе.

Так как каждой угрозе поставлен в соответствие перечень сервисов безопасности, обеспечивающих полное перекрытие конкретной угрозы, все множество сервисов безопасности обеспечивает полное перекрытие множества всех угроз.

Каждому типовому сервису безопасности сопоставлены множества требований безопасности. Эти множества разработаны на основе каталога требований безопасности, содержащегося в СТБ 34.101.-2004 (ИСО/МЭК 15408:1999) [12].

Из подхода к моделированию системы безопасности ИТ как совокупности типовых сервисов безопасности следует, что сервисный подход целесообразно применить и к описанию прикладных решений в области защиты информации. Средство ИТ или средство защиты, реализующее типовой сервис безопасности, в таком случае можно рассматривать как типовое. Для оценки способности средств ИТ или средств защиты информации реализовывать сервисы безопасности каждому сервису безопасности сопоставлены множества требований безопасности, разработанные на основании каталога функциональных требований безопасности [12].

### Список литературы

1. Липаев В.В. Программно-технологическая безопасность информационных систем / В.В. Липаев. – М. : МИФИ, 1997. – 143 с.
2. Обеспечение информационной безопасности в сфере экономики / О.В. Голосов [и др.] // Информационная безопасность : учебное пособие для студентов высших учебных заведений, обучающихся по специальностям в области информационной безопасности. – М. : МГФ «Знание», 2005. – С. 255–298.
3. Единая система программной документации. Руководство системного программиста. Требования к содержанию и оформлению : ГОСТ 19.503–79. – Введ. 01.01.80. – М. : ИПК Издательство стандартов, 1998. – С. 79–81.
4. Защита информации. Испытания программных средств на наличие компьютерных вирусов. Типовое руководство : ГОСТ Р 51188–98. – Введ. 01.07.99. – М. : ИПК Издательство стандартов, 1998.
5. Защита информации. Объект информатизации. Факторы, воздействующие на информацию : ГОСТ Р 51275–99. – Введ. 12.05.99. – М. : ИПК Издательство стандартов, 1999.
6. Единая система программной документации. Руководство программиста. Требования к содержанию и оформлению : ГОСТ 19.504–79. – Введ. 01.01.80. – М. : ИПК Издательство стандартов, 1998. – С. 81–83.

7. Единая система конструкторской документации. Эксплуатационные документы : ГОСТ 2.601–95. – Введ. 01.07.96. – М. : ИПК Издательство стандартов, 1995.
8. Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Термины и определения : ГОСТ 34.003-90. – Введ. 01.01.92. – М. : ИПК Издательство стандартов, 1990.
9. Информационная технология. Виды испытаний автоматизированных систем : ГОСТ 34.603–92. – Введ. 01.01.93. – М. : ИПК Издательство стандартов, 1992.
10. Конявский, В. А. Компьютерная преступность : в 2 т. / В.А. Конявский, С.В. Лопаткин. – М. : РФК-Имидж Лаб, 2006. – Т.1. – 560 с.
11. Конявский, В.А. Основы технологической защиты электронных документов / В.А. Конявский // Компьютерная преступность и информационная безопасность ; под общ. ред. А.П. Леонова. – Минск : АРИЛ, 2000. – С. 298–350.
12. Критерии оценки безопасности информационных технологий : СТБ 34.101-2004 (ИСО/МЭК 15408:1999) – Введ. 21.07.2004. – Минск : Госстандарт, 2004.

Поступила 02.08.12

ООО «МАРФИ»,  
Минск, Переходная, 64-б  
e-mail: 001@marfi.by

**H.V. Fralou**

## **A SYSTEM APPROACH TO INFORMATION TECHNOLOGY PROTECTION**

The suggested model of information security system works using security services. The model is chosen based on the following criteria of systems quality: suitability criterion, optimality criterion and criterion of superiority based on the proposed classification of security threats. The proposed model is a set of security services that can be realized in the IT environment or in the IT itself with the help of information security facilities.

## МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ

УДК 517.958:537.8

В.Т. Ерофеев<sup>1</sup>, Г.Ч. Шушкевич<sup>2</sup>, С.С. Грабчиков<sup>3</sup>, В.Ф. Бондаренко<sup>4</sup>МОДЕЛЬ ЭКРАНИРОВАНИЯ ПОСТОЯННЫХ МАГНИТНЫХ ПОЛЕЙ  
МНОГОСЛОЙНЫМ ЦИЛИНДРИЧЕСКИМ ЭКРАНОМ

*Предлагается алгоритм сведения решения краевой задачи проникновения постоянного магнитного поля через многослойный магнитный цилиндрический экран к решению интегрального уравнения Фредгольма второго рода. Выводится аналитическая формула для вычисления коэффициента эффективности экрана. Интегральное уравнение решается численно. Исследуются экранирующие свойства конечного и бесконечного многослойных экранов с большим числом слоев.*

**Введение**

В настоящее время развитие техники направлено на внедрение технических устройств из композитных многослойных материалов, которые обладают специальными физическими свойствами. В частности, вся космическая техника основана на использовании инновационных разработок по созданию приборов, которые, как правило, чувствительны к воздействию внешних электромагнитных полей. Хорошо известно, что на устойчивость функционирования аппаратуры влияют внешние магнитные поля, воздействие которых приводит к сбоям в ее работе. В связи с этим актуальным является создание экранов различных типов, способствующих улучшению защищенности технических устройств [1–3], а также биологических объектов [4] от воздействия магнитных полей. В работах [5, 6] указано, что анизотропия и многослойность материала экранов улучшают эффективность экранирования. В книге [7] разработаны методы расчета тонких экранов геометрически правильной формы с использованием специальных функций, теорем сложения и усредненных граничных условий на экранах, с помощью которых аналитически решены типовые задачи экранирования.

В настоящей статье методика работы [7] развита применительно к открытым в торцах конечным цилиндрическим многослойным экранам с использованием усредненных нелокальных двухсторонних граничных условий, описывающих проникновение постоянных магнитных полей через слои ортогонально экрану. Так как технология изготовления таких экранов разработана, численная реализация модели позволит оптимизировать многослойные экраны.

**1. Постановка задачи**

В однородном пространстве  $R^3$  с магнитной проницаемостью  $\mu_0 = 1$  расположен цилиндрический тонкостенный экран  $D\{R_1 < \rho < R_2, |z| < l, 0 \leq \varphi < 2\pi\}$  с высотой  $2l$  и толщиной  $h = R_2 - R_1$ ,  $\{\rho, \varphi, z\}$  – цилиндрические координаты. Цилиндр состоит из  $n$  цилиндрических слоев  $\Omega_s\{\rho_s < \rho < \rho_{s+1}, |z| < l, 0 \leq \varphi < 2\pi\}$  с относительными магнитными проницаемостями  $\mu^{(s)}$ ,  $s = \overline{1, n}$ ;  $\rho_1 = R_1$ ,  $\rho_{n+1} = R_2$ ;  $\Delta_s = \rho_{s+1} - \rho_s$  – толщина  $s$ -го слоя,  $h = \sum_{s=1}^n \Delta_s$  – толщина экрана  $D$ .

Обозначим внутреннюю и внешнюю лицевые поверхности экрана  $D$  соответственно  $\Gamma_- \{\rho = R_1, |z| < l, 0 \leq \varphi < 2\pi\}$ ,  $\Gamma_+ \{\rho = R_2, |z| < l, 0 \leq \varphi < 2\pi\}$ . Введем цилиндрическую поверхность  $\Gamma \left\{ \rho = R = \frac{1}{2}(R_1 + R_2), |z| < \infty, 0 \leq \varphi < 2\pi \right\}$ , которая разделяет внутреннюю цилиндриче-

скую область  $D_1 \{0 \leq \rho < R, |z| < \infty, 0 \leq \varphi < 2\pi\}$  и внешнюю  $D_2 \{\rho > R, |z| < \infty, 0 \leq \varphi < 2\pi\}$ . Поверхность  $\Gamma = \Gamma_0 \cup \Gamma_u$ ,  $\Gamma_0 \{\rho = R, |z| > l, 0 \leq \varphi < 2\pi\}$ ,  $\Gamma_u \{\rho = R, |z| < l, 0 \leq \varphi < 2\pi\}$  – срединная поверхность экрана  $D$ . На экран  $D$  воздействует первичное постоянное, ортогональное оси цилиндра магнитное поле  $\vec{H}_0 = -H_0 \vec{e}_x$  с магнитным потенциалом

$$u_0 = H_0 \rho \cos \varphi, \quad H_0 - \text{const.} \quad (1)$$

Обозначим магнитные потенциалы, образующиеся в результате взаимодействия первичного магнитного поля с экраном  $D$ :  $u'_1$  – вторичное поле в области  $D_1$ ;  $u'_2$  – отраженное поле в области  $D_2$ ;  $u_j = u_0 + u'_j$  – суммарное поле в областях  $D_j$ . Магнитные поля определяются формулами

$$\vec{H}_0 = -\text{grad } u_0, \quad \vec{H}_j = -\text{grad } u_j. \quad (2)$$

Сформулируем краевую задачу, описывающую взаимодействие поля  $\vec{H}_0$  с экраном  $D$ .

**Краевая задача.** Для заданного первичного потенциала  $u_0$  требуется определить потенциалы  $u'_j \in C^2(D_j) \cap C^1(\bar{D}_j)$ , которые удовлетворяют:

– уравнению Лапласа

$$\Delta u'_j = 0 \quad \text{в } D_j; \quad (3)$$

– граничным условиям

$$u_2|_{\Gamma_0} = u_1|_{\Gamma_0}, \quad \frac{\partial u_2}{\partial \rho}\bigg|_{\Gamma_0} = \frac{\partial u_1}{\partial \rho}\bigg|_{\Gamma_0}; \quad (4)$$

$$u_2(M_2)\bigg|_{\Gamma_+} = \left( B_{11}u_1(M_1) + B_{12} \frac{\partial u_1(M_1)}{\partial \rho} \right)\bigg|_{\Gamma_-}, \quad (5)$$

$$\left( \frac{\partial u_2(M_2)}{\partial \rho} \right)\bigg|_{\Gamma_+} = \left( B_{21}u_1(M_1) + B_{22} \frac{\partial u_1(M_1)}{\partial \rho} \right)\bigg|_{\Gamma_-},$$

где  $M_j = (\rho = R_j, \varphi, z)$ ,  $|z| < l$ ;

– условиям на бесконечности

$$r u'_1(M) \rightarrow 0 \quad \text{при } r \rightarrow \infty, \quad r u'_2(M) \rightarrow 0 \quad \text{при } r \rightarrow \infty, \quad (6)$$

где  $r$  – сферическая координата точки  $M$ .

Заметим, что двухсторонние граничные условия (5) моделируют проникновение магнитного поля  $\vec{H}_0$  через  $D$ .

## 2. Модель двухсторонних граничных условий

Для вывода граничных условий (5), связывающих поля по обе стороны цилиндрического экрана, рассмотрим бесконечный цилиндрический слой  $\Omega \{r_1 < \rho < r_2, |z| < \infty, 0 \leq \varphi < 2\pi\}$  с относительной магнитной проницаемостью  $\mu$ . Области  $S_1 (0 \leq \rho < r_1)$ ,  $S_2 (\rho > r_2)$  заполнены средами с магнитными проницаемостями  $\mu_1$  и  $\mu_2$ . Обозначим магнитные потенциалы через  $u_j$  в областях  $S_j$  и  $u$  в области  $\Omega$ . На поверхностях раздела сред выполнены классические граничные условия сопряжения [8]

$$(u_1 - u)|_{\rho=r_1} = 0, \quad \left( \mu_1 \frac{\partial u_1}{\partial \rho} - \mu \frac{\partial u}{\partial \rho} \right) \Big|_{\rho=r_1} = 0; \quad (7)$$

$$(u_2 - u)|_{\rho=r_2} = 0, \quad \left( \mu_2 \frac{\partial u_2}{\partial \rho} - \mu \frac{\partial u}{\partial \rho} \right) \Big|_{\rho=r_2} = 0. \quad (8)$$

Учитывая тот факт, что на экран действует поле (1), не зависящее от координаты  $z$ , представим потенциал в слое  $\Omega$  как

$$u = \left( a\rho + \frac{b}{\rho} \right) \cos \varphi, \quad r_1 < \rho < r_2, \quad (9)$$

где  $a, b$  – постоянные.

Подставляя (9) в граничные условия (7), получим

$$a + \frac{b}{r_1^2} = \frac{1}{r_1} \bar{u}_1 \Big|_{\rho=r_1}, \quad a - \frac{b}{r_1^2} = \frac{\mu_1}{\mu} \frac{\partial \bar{u}_1}{\partial \rho} \Big|_{\rho=r_1}, \quad (10)$$

где  $\bar{u}_1 = u_1 / \cos \varphi$ .

Разрешая систему (10), определим постоянные

$$a = \frac{1}{2} \left( \frac{1}{r_1} \bar{u}_1 + \frac{\mu_1}{\mu} \frac{\partial \bar{u}_1}{\partial \rho} \right) \Big|_{\rho=r_1}, \quad b = \frac{r_1^2}{2} \left( \frac{1}{r_1} \bar{u}_1 - \frac{\mu_1}{\mu} \frac{\partial \bar{u}_1}{\partial \rho} \right) \Big|_{\rho=r_1}. \quad (11)$$

Граничные условия (8) будут удовлетворены при подстановке (9) в (8). Тогда с учетом формул (11) получим соотношения

$$u_2 \Big|_{\rho=r_2} = \left( \frac{1}{2} \left( \frac{1}{\bar{r}} + \bar{r} \right) u_1 + \frac{\mu_1 r_1}{2\mu} \left( \frac{1}{\bar{r}} - \bar{r} \right) \frac{\partial u_1}{\partial \rho} \right) \Big|_{\rho=r_1};$$

$$\mu_2 \frac{\partial u_2}{\partial \rho} \Big|_{\rho=r_2} = \left( \frac{\mu}{2r_1} (1 - \bar{r}^2) u_1 + \frac{\mu_1}{2} (1 + \bar{r}^2) \frac{\partial u_1}{\partial \rho} \right) \Big|_{\rho=r_1}.$$

Запишем их в матричном виде.

**Заключение 1.** Получено векторное граничное условие, связывающее поля по обе стороны цилиндрического слоя  $\Omega$ :

$$\vec{W}_2 \Big|_{\rho=r_2} = \hat{A} \vec{W}_1 \Big|_{\rho=r_1}, \quad (12)$$

$$\text{где } \vec{W}_j = \begin{pmatrix} u_j \\ \mu_j \frac{\partial u_j}{\partial \rho} \end{pmatrix}, \quad \hat{A} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}, \quad A_{11} = \frac{1}{2\bar{r}} (1 + \bar{r}^2), \quad A_{12} = \frac{\Delta}{2\mu} (1 + \bar{r}), \quad A_{21} = \frac{\mu\Delta}{2r_1 r_2} (1 + \bar{r}),$$

$$A_{22} = \frac{1}{2} (1 + \bar{r}^2), \quad \Delta = r_2 - r_1, \quad \bar{r} = r_1 / r_2. \quad \blacksquare$$

Запишем граничное условие (12) для цилиндрического слоя  $\Omega_s$  многослойного экрана  $D$ . Воспользуемся классическим граничным условием на цилиндрической поверхности

$$\gamma_s(\rho = \rho_s): \vec{V}_{s-1} \Big|_{\rho=\rho_s} = \vec{V}_s \Big|_{\rho=\rho_s}, \quad \text{где } \vec{V}_s = \begin{pmatrix} v_s \\ \mu^{(s)} \frac{\partial v_s}{\partial \rho} \end{pmatrix}, \quad v_s - \text{магнитный потенциал в цилиндрическом}$$

слое  $\Omega_s$ ;  $\mu^{(s)}$  – относительная магнитная проницаемость слоя  $\Omega_s$ . В результате условие (12) для слоя  $\Omega_s$  имеет вид

$$\vec{V}_{s+1}\Big|_{\rho=\rho_{s+1}} = \hat{A}^{(s)} \vec{V}_s\Big|_{\rho=\rho_s}, \quad (13)$$

$$\text{где } \hat{A}^{(s)} = \begin{pmatrix} A_{11}^{(s)} & A_{12}^{(s)} \\ A_{21}^{(s)} & A_{22}^{(s)} \end{pmatrix}, \quad A_{11}^{(s)} = \frac{1}{2\bar{r}_s}(1 + \bar{r}_s^2), \quad A_{12}^{(s)} = \frac{\Delta_s}{2\mu^{(s)}}(1 + \bar{r}_s), \quad A_{21}^{(s)} = \frac{\mu^{(s)}\Delta_s}{2\rho_s\rho_{s+1}}(1 + \bar{r}_s), \quad (14)$$

$$A_{22}^{(s)} = \frac{1}{2}(1 + \bar{r}_s^2), \quad \bar{r}_s = \rho_s/\rho_{s+1}, \quad \Delta_s = \rho_{s+1} - \rho_s.$$

Воспользуемся граничным условием сопряжения на цилиндрической поверхности  $\gamma_{n+1}(\rho = R_2 = \rho_{n+1})$ :  $\vec{W}_2\Big|_{\rho=R_2} = \vec{V}_n\Big|_{\rho=\rho_{n+1}}$ , тогда для слоя  $\Omega_n$  из (13) следует  $\vec{W}_2\Big|_{\rho=R_2} = \hat{A}^{(n)} \vec{V}_n\Big|_{\rho=\rho_n}$ .

Далее, последовательно применяя соотношения (13), получим

$$\vec{W}_2\Big|_{\rho=R_2} = \hat{A}^{(n)} \hat{A}^{(n-1)} \dots \hat{A}^{(1)} \vec{W}_1\Big|_{\rho=R_1}.$$

**Заключение 2.** Получено векторное граничное условие, связывающее поля по обе стороны цилиндрического  $n$ -слойного экрана  $D$ :

$$\vec{W}_2\Big|_{\rho=R_2} = \hat{B} \vec{W}_1\Big|_{\rho=R_1}, \quad (15)$$

где  $\hat{B} = \hat{A}^{(n)} \hat{A}^{(n-1)} \dots \hat{A}^{(1)} = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}$ , матрицы  $\hat{A}^{(s)}$  определяются формулами (14). ■

Для краевой задачи (3)–(6) в областях  $D_j$  магнитные проницаемости  $\mu_j = \mu_0 = 1$ . В этом случае полученное граничное условие (15) используется в качестве модели граничных условий (5).

### 3. Преобразование граничных условий

Потенциалы магнитных полей  $u'_j = v'_j \cos \varphi$ ,  $u_0 = v_0 \cos \varphi$  в областях  $D_j$  представим в интегральном виде через цилиндрические гармонические функции [9]

$$\begin{aligned} v'_1 &= \int_0^\infty x(\lambda) \frac{I_1(\lambda\rho)}{I_1(\lambda R)} \cos \lambda z d\lambda, \quad 0 \leq \rho < R; \\ v'_2 &= \int_0^\infty y(\lambda) \frac{K_1(\lambda\rho)}{K_1(\lambda R)} \cos \lambda z d\lambda, \quad \rho > R, \end{aligned} \quad (16)$$

где  $x(\lambda), y(\lambda)$  – неизвестные функции;  $I_m(\cdot)$  – модифицированные функции Бесселя;  $K_m(\cdot)$  – функции Макдональда.

Преобразуем нелокальные двухсторонние граничные условия (4), (5) к граничным условиям для потенциалов  $v'_j$ , переводя их на срединную поверхность  $\Gamma_y$ . В результате

$$(v'_2 - v'_1)\Big|_{\Gamma_0} = 0, \quad \left( \frac{\partial v'_2}{\partial \rho} - \frac{\partial v'_1}{\partial \rho} \right)\Big|_{\Gamma_0} = 0; \quad (17)$$

$$(v'_2 - v'_1)\Big|_{\Gamma_y} = f_1(z) + c_1, \quad \left( \frac{\partial v'_2}{\partial \rho} - \frac{\partial v'_1}{\partial \rho} \right)\Big|_{\Gamma_y} = f_2(z) + c_2, \quad (18)$$

где

$$f_1(z) = \left( (B_{11} - 1)v_1' + B_{12} \frac{\partial v_1'}{\partial \rho} \right) \Big|_{\rho=R_1}, \quad |z| < l,$$

$$f_2(z) = \left( B_{21}v_1' + (B_{22} - 1) \frac{\partial v_1'}{\partial \rho} \right) \Big|_{\rho=R_1}, \quad |z| < l;$$
(19)

$$c_1 = \left( B_{11}v_0 + B_{12} \frac{\partial v_0}{\partial \rho} \right) \Big|_{\rho=R_1} - v_0 \Big|_{\rho=R_2} = H_0 (B_{11}R_1 + B_{12} - R_2),$$

$$c_2 = \left( B_{21}v_0 + B_{22} \frac{\partial v_0}{\partial \rho} \right) \Big|_{\rho=R_1} - \frac{\partial v_0}{\partial \rho} \Big|_{\rho=R_2} = H_0 (B_{21}R_1 + B_{22} - 1).$$

В дальнейшем воспользуемся интегральным преобразованием

$$f(z) = \int_0^\infty \tilde{f}(\lambda) \cos \lambda z d\lambda, \quad \tilde{f}(\lambda) = \frac{2}{\pi} \int_0^\infty f(z) \cos \lambda z dz. \quad (20)$$

Учитывая (16), запишем (19) в интегральном виде

$$f_j(z) = \int_0^\infty x(\lambda) g_j(\lambda) \cos \lambda z d\lambda, \quad |z| < l, \quad (21)$$

где

$$g_1(\lambda) = (B_{11}I_1(\lambda R_1) + B_{12}\lambda I_1'(\lambda R_1)) / I_1(\lambda R);$$

$$g_2(\lambda) = (B_{21}I_1(\lambda R_1) + B_{22}\lambda I_1'(\lambda R_1)) / I_1(\lambda R).$$

Продолжим функции (19) на полуось  $0 \leq z < \infty$ :

$$F_j(z) = \begin{cases} f_j(z), & 0 \leq z < l \\ 0, & l < z < \infty \end{cases} = \int_0^\infty \tilde{F}_j(\lambda) \cos \lambda z d\lambda. \quad (22)$$

Применяя к равенству (22) обратное интегральное преобразование (20) и учитывая (21), вычислим

$$\tilde{F}_j(\lambda) = \frac{2}{\pi} \int_0^l f_j(z) \cos \lambda z dz = \frac{2}{\pi} \int_0^\infty x(\eta) g_j(\eta) \int_0^l \cos \eta z \cos \lambda z dz d\eta = \int_0^\infty x(\eta) g_j(\eta) L(\lambda, \eta) d\eta, \quad (23)$$

где

$$L(\lambda, \eta) = \frac{1}{\pi} \left( \frac{\sin(\eta - \lambda)l}{\eta - \lambda} + \frac{\sin(\eta + \lambda)l}{\eta + \lambda} \right).$$

Рассмотрим также кусочно-постоянные функции

$$c_j(z) = \begin{cases} c_j, & 0 \leq z < l \\ 0, & l < z < \infty \end{cases} = c_j \int_0^\infty q(\lambda) \cos \lambda z d\lambda, \quad (24)$$

где  $q(\lambda) = \frac{2 \sin \lambda l}{\pi \lambda}$ .

Используя функции (22), (24), объединим граничные условия (17), (18). Тогда приходим к граничным условиям

$$\begin{aligned} (v'_2 - v'_1)|_{\rho=R} &= F_1(z) + c_1(z), \quad 0 \leq z < \infty; \\ \left( \frac{\partial v'_2}{\partial \rho} - \frac{\partial v'_1}{\partial \rho} \right)|_{\rho=R} &= F_2(z) + c_2(z), \quad 0 \leq z < \infty. \end{aligned} \quad (25)$$

В дальнейшем граничные условия (25) используются как основная модель для решения исходной задачи проникновения магнитного поля через цилиндрический экран.

#### 4. Интегральные уравнения

Используя граничные условия (25), сведем решение краевой задачи (3)–(6) к решению интегрального уравнения Фредгольма второго рода. Функции (16), (22), (24), входящие в равенства (25), записаны в виде интегралов Фурье (20). Применяя обратное интегральное преобразование к равенствам (25), получим уравнения

$$y(\lambda) = x(\lambda) + \tilde{F}_1(\lambda) + c_1 q(\lambda); \quad (26)$$

$$\lambda \left( y(\lambda) \frac{K'_1(\lambda R)}{K_1(\lambda R)} - x(\lambda) \frac{I'_1(\lambda R)}{I_1(\lambda R)} \right) = \tilde{F}_2(\lambda) + c_2 q(\lambda). \quad (27)$$

С помощью (26) исключим  $y(\lambda)$  из уравнения (27). Учитывая формулу  $I'_1(x)K_1(x) - K'_1(x)I_1(x) = \frac{1}{x}$  [10], получим равенство

$$x(\lambda) = RI_1(\lambda R) \left[ \lambda \tilde{F}_1(\lambda) K'_1(\lambda R) - \tilde{F}_2(\lambda) K_1(\lambda R) + q(\lambda) (c_1 \lambda K'_1(\lambda R) - c_2 K_1(\lambda R)) \right]. \quad (28)$$

Подставив интегралы (23) в (28), получим уравнение

$$\begin{aligned} x(\lambda) = RI_1(\lambda R) & \left( \lambda K'_1(\lambda R) \int_0^\infty x(\eta) g_1(\eta) L(\lambda, \eta) d\eta - K_1(\lambda R) \int_0^\infty x(\eta) g_2(\eta) L(\lambda, \eta) d\eta \right) + \\ & + Rq(\lambda) I_1(\lambda R) (c_1 \lambda K'_1(\lambda R) - c_2 K_1(\lambda R)). \end{aligned} \quad (29)$$

Умножим (29) на регуляризирующий множитель  $e^{-\lambda \frac{h}{4}}$  и введем новую неизвестную функцию  $z(\lambda) = x(\lambda) e^{-\lambda \frac{h}{4}}$ . Получим интегральное уравнение

$$z(\lambda) + \int_0^\infty K(\lambda, \eta) z(\eta) d\eta = f(\lambda), \quad 0 \leq \lambda < \infty, \quad (30)$$

где  $K(\lambda, \eta) = RI_1(\lambda R) (g_2(\eta) K_1(\lambda R) - \lambda g_1(\eta) K'_1(\lambda R)) e^{(\eta-\lambda) \frac{h}{4}} L(\lambda, \eta);$

$$f(\lambda) = Rq(\lambda) I_1(\lambda R) (\lambda c_1 K'_1(\lambda R) - c_2 K_1(\lambda R)) \exp\left(-\lambda \frac{h}{4}\right).$$

Для доказательства принадлежности функций интегрального уравнения (30) к пространствам  $L_2$  воспользуемся асимптотическими формулами при  $x \rightarrow \infty$  [10]:

$$I_1(x) \approx \frac{1}{\sqrt{2\pi x}} e^x, \quad I_1'(x) \approx \frac{1}{\sqrt{2\pi x}} e^x, \quad K_1(x) \approx \sqrt{\frac{\pi}{2x}} e^{-x}, \quad K_1'(x) \approx -\sqrt{\frac{\pi}{2x}} e^{-x}.$$

$$\text{Тогда } g_1(\eta) \approx (A_{11} + A_{12}\eta) \sqrt{\frac{R}{R_1}} e^{-\eta \frac{h}{2}}, \quad g_2(\eta) \approx (A_{21} + A_{22}\eta) \sqrt{\frac{R}{R_1}} e^{-\eta \frac{h}{2}}.$$

При  $\lambda \rightarrow \infty, \eta \rightarrow \infty$  асимптотическое представление функций уравнения (30) имеет вид

$$K(\lambda, \eta) \approx \frac{R}{2} \sqrt{\frac{R}{R_1}} (A_{21} + \lambda A_{11} + \eta A_{22} + \lambda \eta A_{12}) e^{-(\lambda+\eta) \frac{h}{4}} L(\lambda, \eta);$$

$$f(\lambda) \approx -\frac{q(\lambda)}{2\lambda} (c_2 + c_1 \lambda) e^{-\lambda \frac{h}{4}}.$$

Так как  $K(\lambda, \eta) \in L_2((0 < \lambda < \infty) \times (0 < \eta < \infty))$ ,  $f(\lambda) \in L_2(0 < \lambda < \infty)$ , интегральное уравнение (30) является интегральным уравнением Фредгольма второго рода.

В уравнение (30) введем безразмерные параметры интегрирования  $\bar{\lambda} = \lambda R$ ,  $\bar{\eta} = \eta R$  с известной функцией  $v(\lambda R) = \frac{z(\lambda)}{R^2 H_0}$ . Получим уравнение

$$v(\bar{\lambda}) + \int_0^\infty K_0(\bar{\lambda}, \bar{\eta}) v(\bar{\eta}) d\bar{\eta} = f_0(\bar{\lambda}), \quad 0 \leq \bar{\lambda} < \infty, \quad (31)$$

$$\text{где } K_0(\bar{\lambda}, \bar{\eta}) = l(\bar{\lambda}, \bar{\eta}) I_1(\bar{\lambda}) \left( R G_2(\bar{\eta}) K_1(\bar{\lambda}) - \bar{\lambda} G_1(\bar{\eta}) K_1'(\bar{\lambda}) \right) \exp\left( (\bar{\eta} - \bar{\lambda}) \frac{\bar{h}}{4} \right);$$

$$G_1(\bar{\eta}) = \left( B_{11} I_1(\bar{\eta} \bar{R}_1) + B_{12} \frac{\bar{\eta}}{R} I_1'(\bar{\eta} \bar{R}_1) \right) / I_1(\bar{\eta}), \quad \bar{h} = \frac{h}{R};$$

$$G_2(\bar{\eta}) = \left( B_{21} I_1(\bar{\eta} \bar{R}_1) + B_{22} \frac{\bar{\eta}}{R} I_1'(\bar{\eta} \bar{R}_1) \right) / I_1(\bar{\eta}), \quad \bar{R}_1 = \frac{R_1}{R};$$

$$l(\bar{\lambda}, \bar{\eta}) = \frac{1}{\pi} \left( \frac{\sin(\bar{\eta} - \bar{\lambda}) \bar{l}}{\bar{\eta} - \bar{\lambda}} + \frac{\sin(\bar{\eta} + \bar{\lambda}) \bar{l}}{\bar{\eta} + \bar{\lambda}} \right), \quad \bar{l} = \frac{l}{R};$$

$$f_0(\bar{\lambda}) = \frac{2}{\pi \bar{\lambda}} \sin(\bar{\lambda} \bar{l}) I_1(\bar{\lambda}) \left( \bar{c}_1 \frac{\bar{\lambda}}{R} K_1'(\bar{\lambda}) - \bar{c}_2 K_1(\bar{\lambda}) \right) \exp\left( -\bar{\lambda} \frac{\bar{h}}{4} \right);$$

$$\bar{c}_1 = B_{11} R_1 + B_{12} - R_2, \quad \bar{c}_2 = B_{21} R_1 + B_{22} - 1;$$

$$K_1'(x) = -\frac{1}{2} (K_0(x) + K_2(x)), \quad I_1'(x) = \frac{1}{2} (I_0(x) + I_2(x)).$$

Для вычисления функций Бесселя при больших аргументах воспользуемся аппроксимацией модифицированных функций Бесселя [10, с. 199]:

$$I_0(x) \approx i_0(x) e^x, \quad 3,75 < x < \infty,$$

$$i_0(x) = \frac{1}{\sqrt{x}} \left( 0,39894228 + 0,01328592\bar{x} + 0,00225319\bar{x}^2 - 0,00157565\bar{x}^3 + 0,00916281\bar{x}^4 - \right. \\ \left. - 0,02057706\bar{x}^5 + 0,02635537\bar{x}^6 - 0,01647633\bar{x}^7 + 0,00392377\bar{x}^8 \right),$$

$$\bar{x} = \frac{3,75}{x}, \quad 3,75 < x < \infty.$$

$$I_1(x) \approx i_1(x)e^x, \quad 3,75 < x < \infty,$$

$$i_1(x) = \frac{1}{\sqrt{x}} \left( 0,39894228 - 0,03988024\bar{x} - 0,00362018\bar{x}^2 + 0,00163801\bar{x}^3 - 0,01031555\bar{x}^4 + \right. \\ \left. + 0,02282967\bar{x}^5 - 0,02895312\bar{x}^6 + 0,01787654\bar{x}^7 - 0,00420059\bar{x}^8 \right).$$

$$I_1'(x) \approx i_2(x)e^x, \quad i_2(x) = i_0(x) - \frac{1}{x}i_1(x).$$

$$K_0(x) \approx k_0(x)e^{-x}, \quad 2 < x < \infty,$$

$$k_0(x) = \frac{1}{\sqrt{x}} \left( 1,25331414 - 0,07832358\bar{\bar{x}} + 0,02189568\bar{\bar{x}}^2 - 0,01062446\bar{\bar{x}}^3 + \right. \\ \left. + 0,00587872\bar{\bar{x}}^4 - 0,00251540\bar{\bar{x}}^5 + 0,00053208\bar{\bar{x}}^6 \right),$$

$$\bar{\bar{x}} = \frac{2}{x}.$$

$$K_1(x) \approx k_1(x)e^{-x}, \quad 2 < x < \infty,$$

$$k_1(x) = \frac{1}{\sqrt{x}} \left( 1,25331414 - 0,23498619\bar{\bar{x}} - 0,03655620\bar{\bar{x}}^2 + 0,01504268\bar{\bar{x}}^3 - \right. \\ \left. - 0,00780353\bar{\bar{x}}^4 + 0,00325614\bar{\bar{x}}^5 - 0,00068245\bar{\bar{x}}^6 \right).$$

$$K_1'(x) \approx k_2(x)e^{-x},$$

$$k_2(x) = - \left( k_0(x) + \frac{1}{x}k_1(x) \right).$$

Приведем формулы для правой части и ядра интегрального уравнения (31):

$$f_0(\bar{\lambda}) = \frac{2}{\pi\bar{\lambda}} \sin(\bar{\lambda}l) i_1(\bar{\lambda}) \left( \bar{c}_1 \frac{\bar{\lambda}}{R} k_2(\bar{\lambda}) - \bar{c}_2 k_1(\bar{\lambda}) \right) \exp\left(-\bar{\lambda} \frac{h}{4}\right);$$

$$K_0(\bar{\lambda}, \bar{\eta}) = l(\bar{\lambda}, \bar{\eta}) i_1(\bar{\lambda}) \left( Rg_2(\bar{\eta}) k_1(\bar{\lambda}) - \bar{\lambda}g_1(\bar{\eta}) k_2(\bar{\lambda}) \right) \exp\left(-(\bar{\eta} + \bar{\lambda}) \frac{h}{4}\right),$$

где  $g_1(\bar{\eta}) = \left( B_{11}i_1(\bar{\eta}R_1) + B_{12} \frac{\bar{\eta}}{R} i_2(\bar{\eta}R_1) \right) / i_1(\bar{\eta});$

$$g_2(\bar{\eta}) = \left( B_{21}i_1(\bar{\eta}R_1) + B_{22} \frac{\bar{\eta}}{R} i_2(\bar{\eta}R_1) \right) / i_1(\bar{\eta}).$$

## 5. Вычисление коэффициентов экранирования

Вычислим магнитное поле в центре экрана

$$\vec{H}_1 \Big|_{\substack{z=0 \\ \rho=0}} = -\text{grad} u_1 \Big|_{\substack{z=0 \\ \rho=0}} = -H_0 \vec{e}_x - \frac{1}{2} \int_0^\infty \frac{\lambda x(\lambda)}{I_1(\lambda R)} d\lambda \vec{e}_x = -H_0 \left( 1 + L(v(\bar{\lambda})) \right) \vec{e}_x,$$

где оператор

$$L(v(\bar{\lambda})) = \frac{1}{2} \int_0^\infty \frac{\bar{\lambda} v(\bar{\lambda})}{I_1(\bar{\lambda})} \exp\left(\bar{\lambda} \frac{h}{4}\right) d\bar{\lambda} = \frac{1}{2} \int_0^\infty \frac{\bar{\lambda} v(\bar{\lambda})}{i_1(\bar{\lambda})} \exp\left(-\bar{\lambda} \left(1 - \frac{h}{4}\right)\right) d\bar{\lambda}.$$

Определим коэффициент экранирования в центре цилиндрического экрана с помощью выражения

$$K_\vartheta = \frac{\left| \vec{H}_1 \right|}{\left| \vec{H}_0 \right| \Big|_{\substack{z=0 \\ \rho=0}}} = \left| 1 + L(v(\bar{\lambda})) \right|. \quad (32)$$

Коэффициент эффективности экрана определяется формулой

$$\mathfrak{D} = \frac{1}{K_\vartheta}. \quad (33)$$

Заметим, что для бесконечно протяженного цилиндрического многослойного экрана коэффициент эффективности

$$\mathfrak{D}_\infty = \frac{1}{2} \left( \frac{R_1}{R_2} B_{11} + \frac{B_{12}}{R_2} + R_1 B_{21} + B_{22} \right),$$

где  $B_{js}$  – элементы матрицы из формулы (15).

В случае однослойного однородного бесконечного экрана толщины  $h$  с магнитной проницаемостью  $\mu$  коэффициент эффективности экрана

$$\mathfrak{D}_\infty^1 = \frac{1}{2} \left[ \frac{1}{2} \left( \mu + \frac{1}{\mu} \right) \frac{h}{R_2} (1 + \bar{R}) + \bar{R}^2 + 1 \right], \quad \bar{R} = \frac{R_1}{R_2}.$$

## 6. Алгоритм вычисления эффективности бесконечного многослойного цилиндрического экрана

Рассматривается круговой цилиндрический многослойный бесконечный экран толщины  $h$ , который находится под воздействием постоянного внешнего магнитного поля, направленного ортогонально оси цилиндра. Исследуется величина, характеризующая ослабление поля во внутренней области экрана.

*Алгоритм*

1. Исходные данные:

$H_0$  – величина внешнего постоянного магнитного поля;

$n$  – число слоев в цилиндрическом экране;

$R_1$  – внутренний радиус цилиндрического экрана;

$\Delta_1, \Delta_2, \dots, \Delta_s, \dots, \Delta_n$  – толщина слоев экрана, занумерованных из внутренней области во внешнюю;

$\mu^{(1)}, \mu^{(2)}, \dots, \mu^{(s)}, \dots, \mu^{(n)}$  – относительные магнитные проницаемости слоев экрана.

2. Промежуточные величины:

$h = \Delta_1 + \Delta_2 + \dots + \Delta_n$  – толщина экрана;

$R_2 = R_1 + h$  – внешний радиус экрана;

$$\bar{R} = \frac{R_1}{R_2}, \quad r_1 = R_1, \quad r_2 = R_1 + \Delta_1, \quad \dots, \quad r_{s+1} = r_s + \Delta_s, \quad \dots, \quad r_{n+1} = r_n + \Delta_n;$$

$$p_s = \frac{r_s}{r_{s+1}}, \quad s = 1, 2, \dots, n.$$

Последовательность матриц  $2 \times 2$ :

$$\hat{A}^{(s)} = \begin{pmatrix} A_{11}^{(s)} & A_{12}^{(s)} \\ A_{21}^{(s)} & A_{22}^{(s)} \end{pmatrix}, \quad s = 1, 2, \dots, n,$$

$$A_{11}^{(s)} = \frac{1}{2p_s}(1 + p_s^2), \quad A_{12}^{(s)} = \frac{\Delta_s}{2\mu^{(s)}}(1 + p_s), \quad A_{21}^{(s)} = \frac{\mu^{(s)}\Delta_s}{2r_s r_{s+1}}(1 + p_s), \quad A_{22}^{(s)} = \frac{1}{2}(1 + p_s^2).$$

Произведение матриц:

$$\hat{B} = \hat{A}^{(n)} \hat{A}^{(n-1)} \dots \hat{A}^{(s)} \dots \hat{A}^{(2)} \hat{A}^{(1)} = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}.$$

3. Итоговые величины:

$\mathcal{E}_\infty = \frac{D}{2}$  – коэффициент эффективности экранирования;

$K_s = \frac{2}{D}$  – коэффициент ослабления поля внутри экрана (коэффициент экранирования),

где  $D = \bar{R} \left( B_{11} + \frac{B_{12}}{R_1} + R_2 B_{21} \right) + B_{22}$ .

## 7. Вычислительный эксперимент

С помощью системы компьютерной математики Mathcad [11] проведены вычисления коэффициента эффективности экранирования магнитного поля многослойным бесконечным и конечным цилиндрическими экранами.

Используя квадратурную формулу Лагерра при  $N=15$  [10], интегральное уравнение Фредгольма второго рода (31) преобразовано к системе линейных алгебраических уравнений вида

$$z_n + \sum_{k=0}^N A_k K_{nk} z_k = g_n, \quad n = 0, 1, \dots, N, \quad (34)$$

где  $A_k = w_k \exp(\bar{\lambda}_k)$ ;  $K_{nk} = K_0(\bar{\lambda}_n, \bar{\lambda}_k)$ ;  $g_n = f_0(\bar{\lambda}_n)$ ,  $\bar{\lambda}_n$  – нули многочлена Лагерра  $L_N(x)$ ;  $w_k$  – весовые коэффициенты.

Система (34) соответствует интегральному уравнению Фредгольма второго рода (31). Решение системы (34) принимается за численное решение интегрального уравнения (31) в точках  $\bar{\lambda} = \bar{\lambda}_k$ ,  $k = 0, 1, \dots, N$ :  $v(\bar{\lambda}_k) = z_k$ .

Рассмотрен случай экрана, когда магнитные и немагнитные слои чередуются:  
 $n = 2m$ , где  $m$  – число пар слоев;

$\Delta_1 = \Delta_3 = \Delta_5 = \dots = \Delta_{n-1} = \frac{h_{\text{сум}}}{m}$  – толщина магнитных слоев, где  $h_{\text{сум}}$  – суммарная толщина

магнитных слоев;

$\Delta_2 = \Delta_4 = \dots = \Delta_n = \alpha \Delta_1$  – толщина немагнитных слоев, где  $\alpha$  – параметр отношения толщины немагнитного слоя к толщине магнитного слоя для пары слоев;

$\mu^{(1)} = \mu^{(3)} = \dots = \mu^{(n-1)} = \mu^I$  – проницаемости магнитных слоев;

$\mu^{(2)} = \mu^{(4)} = \dots = \mu^{(n)} = \mu^{II} = 1$  – проницаемости немагнитных слоев.

Графики коэффициента эффективности экранирования магнитного поля многослойными бесконечным и конечным ( $l = 20$  мм) цилиндрическими экранами изображены на рис. 1–4.

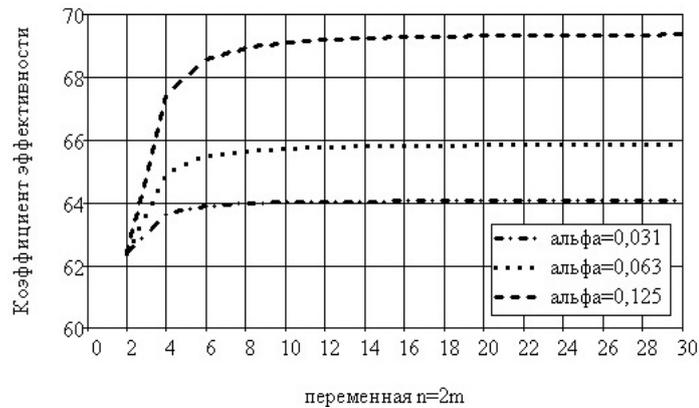


Рис. 1. Графики зависимости коэффициента эффективности экранирования магнитного поля многослойным бесконечным цилиндрическим экраном от числа слоев

для  $\mu^I = 3 \cdot 10^3$ ,  $R_1 = 11$  мм,  $h_{\text{сум}} = 0,48$  мм и  $\alpha = 0,031; 0,063; 0,125$  при  $m = 1, 2, \dots, 15$

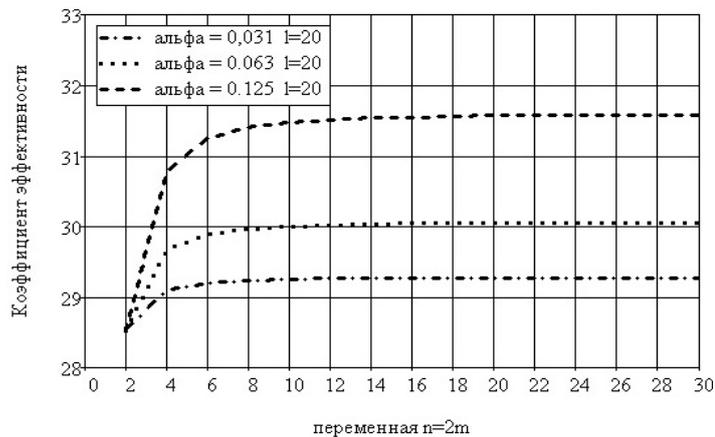


Рис. 2. Графики зависимости коэффициента эффективности экранирования магнитного поля многослойным конечным цилиндрическим экраном от числа слоев

для  $\mu^I = 3 \cdot 10^3$ ,  $R_1 = 11$  мм,  $h_{\text{сум}} = 0,48$  мм и  $\alpha = 0,031; 0,063; 0,125$  при  $m = 1, 2, \dots, 15$

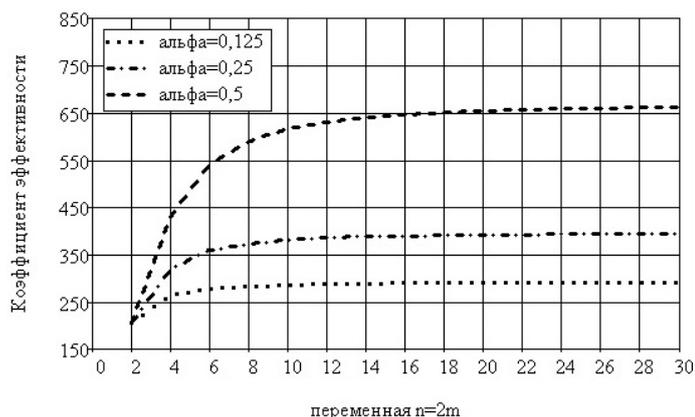


Рис. 3. Графики зависимости коэффициента эффективности экранирования магнитного поля многослойным бесконечным цилиндрическим экраном от числа слоев для  $\mu^l = 10^4$ ,  $R_1 = 11$  мм,  $h_{\text{сум}} = 0,48$  мм и  $\alpha = 0,125; 0,25; 0,5$  при  $m = 1, 2, \dots, 15$

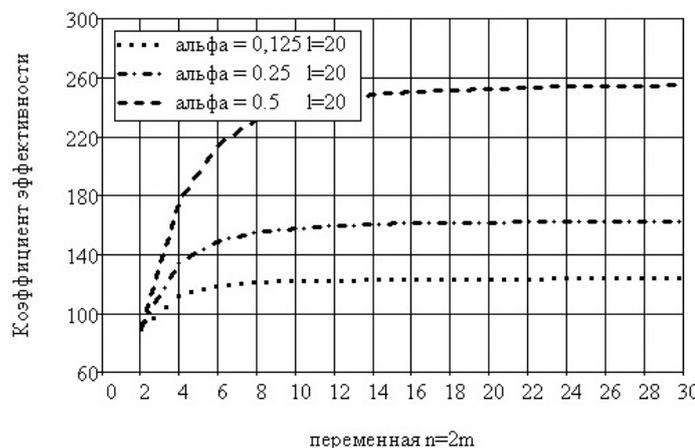


Рис. 4. Графики зависимости коэффициента эффективности экранирования магнитного поля многослойным конечным цилиндрическим экраном от числа слоев для  $\mu^l = 10^4$ ,  $R_1 = 11$  мм,  $h_{\text{сум}} = 0,48$  мм и  $\alpha = 0,125; 0,25; 0,5$  при  $m = 1, 2, \dots, 15$

Сравнение эффективностей экранирования слоистых экранов показывает, что эффективность бесконечного цилиндрического экрана в 2–2,6 раз выше эффективности конечного экрана. Кроме того, увеличение толщины немагнитных слоев экрана, расположенных между магнитными слоями, увеличивает эффективность экранирования. Анализ графиков позволяет заключить, что эффективность композитного экрана, состоящего из чередующихся магнитных и немагнитных слоев, выше сплошного экрана.

### Заключение

В работе представлена математическая модель двухсторонних нелокальных граничных условий на поверхности тонкостенного многослойного цилиндрического экрана, которые моделируют проникновение постоянных магнитных полей через стенки экрана. На основании граничных условий сформулирована краевая задача экранирования магнитного поля цилиндрическим экраном конечной длины. Разработана методика преобразования краевой задачи к решению интегрального уравнения Фредгольма второго рода.

Алгоритмы аналитических преобразований основаны на использовании специальных функций Бесселя. В результате математическое моделирование задачи сведено к численному решению интегрального уравнения и вычислению коэффициента эффективности слоистого экрана в зависимости от магнитных проницаемостей и толщины слоев.

Показано, что дробление сплошного магнитного цилиндрического экрана на магнитные и немагнитные слои увеличивает эффективность экранирования вплоть до 12 слоев (шесть магнитных и шесть немагнитных слоев). При этом считается, что суммарная толщина магнитных слоев сохраняется. Математическое моделирование с помощью предложенного в статье алгоритма может быть использовано при разработке технологий изготовления многослойных экранов с улучшенными свойствами экранирования. Разработанная методика также может быть распространена на случай низкочастотных магнитных полей [12].

### Список литературы

1. Шапиро, Д.Н. Электромагнитное экранирование / Д.Н. Шапиро. – Долгопрудный : Изд. дом «Интеллект», 2010. – 120 с.
2. Резинкина, М.М. Использование численных расчетов для выбора средств экранирования от действия магнитного поля / М.М. Резинкина // Журнал технической физики. – 2007. – Т. 77, вып. 11. – С. 17–24.
3. Глонягин, Ю.В. Элементы теории и расчета магнитостатических полей ферромагнитных тел / Ю.В. Глонягин. – Л. : Судостроение, 1967. – 180 с.
4. Холодов, Ю.А. Влияние магнитных полей на биологические объекты / Ю.А. Холодов. – М. : Наука, 1971. – 214 с.
5. Аполлонский, С.М. Анизотропия материалов – резерв повышения эффективности электромагнитных экранов / С.М. Аполлонский, В.Н. Острейко // Электротехника. – 1994. – № 11. – С. 51–53.
6. Фуфаева, Л.И. Эффективность экранирования постоянных магнитных полей многослойными экранами / Л.И. Фуфаева, А.Б. Тимофеев // Труды МАИ. – 1976. – Вып. 364. – С. 58–63.
7. Аполлонский, А.С. Электромагнитные поля в экранирующих оболочках / С.М. Аполлонский, В.Т. Ерофеев. – Минск : Университетское, 1988. – 247 с.
8. Ерофеев, В.Т. Основы математического моделирования / В.Т. Ерофеев, И.С. Козловская. – Минск : БГУ, 2002. – 195 с.
9. Ерофеев, В.Т. Теоремы сложения / В.Т. Ерофеев. – Минск : Наука и техника, 1989. – 256 с.
10. Справочник по специальным функциям с формулами, графиками и таблицами / под ред. М. Абрамовица, И. Стиган. – М. : Наука, 1979. – 830 с.
11. Шушкевич, Г.Ч. Компьютерные технологии в математике. Система Mathcad 14. Ч. 1 / Г.Ч. Шушкевич, С.В. Шушкевич. – Минск : Изд-во Гревцова, 2010. – 287 с.
12. Ерофеев, В.Т. Моделирование процессов проникновения низкочастотных магнитных полей через многослойные экраны / В.Т. Ерофеев // Информатика. – 2011. – № 3 (31). – С. 22–32.

Поступила 16.05.12

<sup>1</sup>Учреждение БГУ «НИИ прикладных проблем математики и информатики»,  
Минск, пр. Независимости, 4  
e-mail: [bsu\\_erofeenko@tut.by](mailto:bsu_erofeenko@tut.by)

<sup>2</sup>Гродненский государственный университет  
им. Янки Купалы,  
Гродно, ул. Ожешко, 22  
e-mail: [g\\_shu@rambler.ru](mailto:g_shu@rambler.ru)

<sup>3</sup>НПЦ НАН Беларуси по материаловедению,  
Минск, ул. П. Бровки, 19  
e-mail: gss@iftp.bas-net.by

<sup>4</sup>Высший государственный колледж связи,  
Минск, ул. Ф. Скорины, 8/2

**V.T. Erofeenko, G.Ch. Shushkevich, S.S. Grabchikov, V.F. Bondarenko**

**MODEL OF CONSTANT MAGNETIC FIELDS SCREENING BY  
MULTILAYER CYLINDRICAL SCREEN**

An algorithm of reducing the solution of a boundary value problem of penetration of a constant magnetic field through a magnetic multilayer cylindrical screen to the solution of a Fredholm integral equation of the second kind is offered. An analytical formula for calculation the efficiency ratio of the screen is derived. The integral equation is solved numerically. Shielding characteristics of finite and infinite multilayer screens with a large number of layers are investigated.

УДК 004.912

С.Ф. Липницкий

## ИНДЕКСИРОВАНИЕ ТЕКСТОВОЙ ИНФОРМАЦИИ НА ОСНОВЕ МОДЕЛИРОВАНИЯ ВЕРБАЛЬНЫХ АССОЦИАЦИЙ

*Предлагаются алгоритмы индексирования текстовых документов на основе математической модели представления знаний в виде вербально-ассоциативных сетей. Рассматриваются структурные особенности информационной базы, построенной с учетом результатов моделирования. Приводятся формулы для вычисления информативности слов и их вербально-ассоциативных пар при индексировании текстовой информации.*

### Введение

Под вербальными ассоциациями в лингвистике и психолингвистике понимают семантические связи между словами в языке (тексте, речи), которые соответствуют ассоциативным отношениям между обозначаемыми ими сущностями в реальном мире. В результате формализации вербальных ассоциаций в работе [1] построена модель представления знаний в информационных системах в виде вербально-ассоциативной сети, т. е. графа отношения вербально-ассоциативной связи слов в тематическом корпусе текстов, представляющем некоторую предметную область. Каждая вершина этого графа помечена значением информативности соответствующего слова, а каждое ребро – значением информативности вербально-ассоциативной связи между инцидентными ему вершинами (словами).

Как показано в статье [1], предложенная модель представления знаний может быть использована при автоматическом индексировании текстовых документов и запросов на поиск информации, реферировании и аннотировании текста, а также при информационном поиске.

На уровне моделирования индексирование текстов и запросов сводится к построению их вербально-ассоциативных сетей. При программной реализации информационной системы поисковые образы текстовых документов и запросов будем представлять в виде совокупностей вербально-ассоциативных пар слов, каждому из которых, а также самим парам соответствуют значения их информативности.

Рассмотрим алгоритмы индексирования тематических корпусов текстов, политематических и монотематических текстов, а также запросов пользователей на поиск информации с учетом следующей структуры информационной базы.

### 1. Архитектура информационной базы

Информационная база включает базу данных и базу знаний (рис. 1). В базе данных содержатся тематические корпусы текстов и архив различных полнотекстовых документов и их рефератов, а в базе знаний – лингвистические словари и вербально-ассоциативные сети тематических корпусов текстов и отдельных документов (поисковые образы).

#### 1.1. База данных

В базе данных представлены тематические и полные корпусы текстов. Тематический корпус – это совокупность текстовых документов по конкретной тематике, характеризующей соответствующую предметную область. Полный корпус представляет собой объединение всех тематических корпусов. Кроме корпусов текстов в базе данных могут храниться различные полнотекстовые документы, их рефераты и прочая актуальная информация.

*Корпусы текстов.* Формально текст  $T$  – это любое непустое подмножество входного языка информационной системы, если на этом подмножестве определена редукция  $\prec^r = \prec \setminus \prec^2$  линейного порядка  $\prec$  (транзитивного и антисимметричного бинарного отношения на множестве

$T$ , которое связано на  $T$ , т. е. для любых  $a, b \in T$  или  $a < b$ , или  $b < a$ , или  $a = b$ ). Цепочки текста  $T$  – суть предложения этого текста. Обозначим через  $Ct_i$   $i$ -й тематический корпус текстов, т. е. множество текстов, соответствующих некоторой  $i$ -й предметной области. Тогда совокупность всех тематических корпусов  $Cf = \{Ct_i \mid i = \overline{1, n}\}$  – это полный корпус текстов.

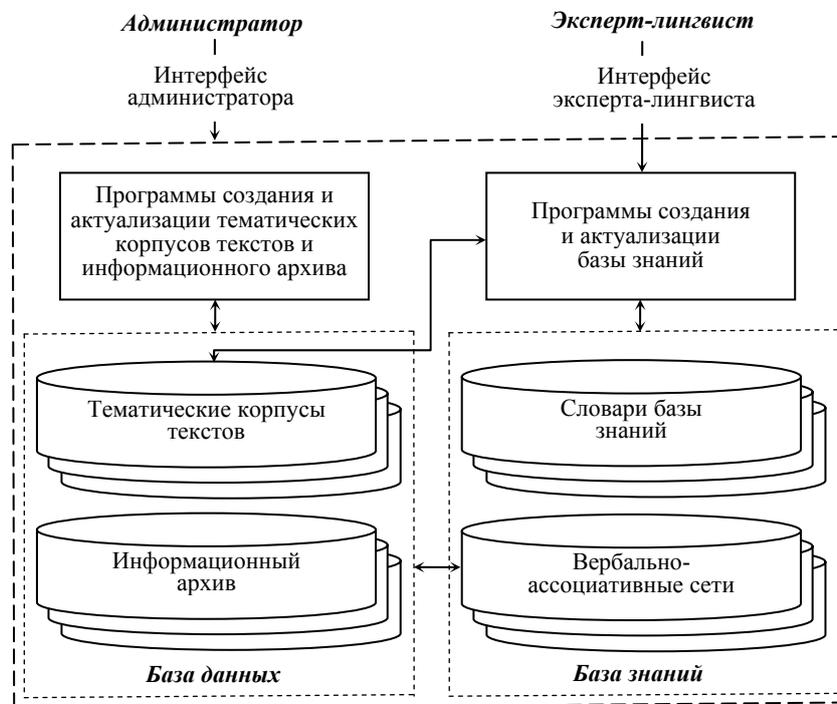


Рис. 1. Состав информационной базы

## 1.2. База знаний

Под базой знаний будем понимать совокупность лингвистических словарей и вербально-ассоциативных сетей предметных областей. Для реализации функций индексирования текстовой информации будем использовать частотные словари словоформ и вербально-ассоциативных пар слов, словарь словоизменительных парадигм и словарь синонимичных словоформ.

*Частотный словарь словоформ.* Пусть  $a$  – некоторая словоформа,  $n_{Cf}^a$  и  $n_{Ct_i}^a$  ( $i = \overline{1, n}$ ) – абсолютные частоты ее появления соответственно в полном и  $i$ -м тематическом корпусе текстов,  $W_{Cf}$  – множество всех словоформ полного корпуса текстов  $Cf$ . Тогда совокупность кортежей  $Dis_a = \{\langle a, n_{Cf}^a, n_{Ct_1}^a, n_{Ct_2}^a, \dots, n_{Ct_n}^a \rangle \mid a \in W_{Cf}\}$  будем называть частотным словарем словоформ (табл. 1).

Таблица 1

Фрагмент частотного словаря словоформ

Словоформа	$n_{Cf}^a$	$n_{Ct_1}^a$	...	$n_{Ct_n}^a$	Код (номер) парадигмы
			...		
стол	0204055	0056534	...	0014445	00000094
стола	0401657	0074526	...	0023747	00000094
			...		

*Частотный словарь вербально-ассоциативных пар слов.* Структура этого словаря аналогична структуре словаря  $Dis_a$ . Рассмотрим пару произвольных слов  $a, b \in W_{Cf}$ . Обозначим через  $n_{Cf}^{ab}$  и  $n_{Ct_i}^{ab}$  абсолютные частоты совместной встречаемости слов  $a$  и  $b$  в одном и том же предложении полного  $Cf$  и тематического  $Ct_i$  ( $i = \overline{1, n}$ ) корпусов текстов. Тогда множество  $Dis_{ab} =$

$= \{ \langle (a, b), n_{Cf}^{ab}, n_{Ct_1}^{ab}, n_{Ct_2}^{ab}, \dots, n_{Ct_n}^{ab} \rangle \mid a, b \in W_{Cf}, n_{Cf}^{ab} \neq 0, n_{Ct_i}^{ab} \neq 0, i = \overline{1, n} \}$  назовем частотным словарем вербально-ассоциативных пар слов (табл. 2).

Таблица 2

Фрагмент частотного словаря вербально-ассоциативных пар слов

Пара словоформ	$n_{Cf}^a$	$n_{Ct_1}^a$	...	$n_{Ct_n}^a$
...				
(стол, комнате)	03020	00543	...	00121
(стола, поверхности)	04023	00623	...	00242
...				

*Словарь словоизменительных парадигм.* Пусть  $a$  – произвольное слово, а  $Par_a$  – множество всех словоизменений слова  $a$  (включая  $a$ ). Тогда словарь словоизменительных парадигм представим в виде множества  $Dis_{par} = \{ (a, Par_a) \mid a \in W_{Cf}, a \in Par_a \}$  (табл. 3).

Таблица 3

Состав и структура словаря парадигм (конечное состояние)

Код (номер) парадигмы	Парадигма
...	
00000094	стол
	стола
	...
...	

Словарь парадигм создается и актуализируется в человеко-машинном режиме с использованием соответствующего инструментария эксперта-лингвиста.

*Словарь синонимичных словоформ.* Словарь состоит из групп синонимичных словоформ, которые могут быть использованы при определении информативности слов и их вербально-ассоциативных пар.

Обозначим через  $Syn_a$  множество всех синонимов слова  $a \in W_{Cf}$ . Тогда словарь синонимичных словоформ – это множество  $Dis_{syn} = \{ (a, Syn_a) \mid a \in W_{Cf}, a \in Syn_a \}$  (табл. 4).

Таблица 4

Фрагмент словаря синонимичных словоформ

Словоформа	Синонимичные словоформы
...	
языкознание	лингвистика
	языковедение
...	

Словарь синонимов создается экспертом-лингвистом с использованием специальных программных средств.

## 2. Индексирование тематических корпусов текстов

Условно объединим все тексты полного корпуса и всех тематических корпусов и будем рассматривать эти корпусы как отдельные текстовые документы.

### 2.1. Информативность слов и вербально-ассоциативных пар слов в тематическом корпусе текстов

Рассмотрим полный корпус текстов  $Cf$  и тематический корпус  $Ct_i$ , соответствующий некоторой  $i$ -й предметной области. Информативность всякой словоформы  $a$  из корпуса  $Ct_i$  при его индексировании будем вычислять как отношение абсолютной частоты встречаемости сло-

воформы  $a$  в тематическом корпусе текстов  $Ct_i$  к абсолютной частоте ее появления в полном корпусе текстов  $Cf$  [2]. На практике при вычислении информативности  $I_{Ct_i}^a$  слова  $a$  необходимо учитывать его словоизменения и синонимию, зафиксированные в словарях  $Dic_{par}$  и  $Dic_{syn}$  соответственно. В этом случае указанное отношение частот примет вид

$$I_{Ct_i}^a = \frac{n_{Ct_i}^a + \sum_{b \in Par_a, b \neq a} n_{Ct_i}^b + \sum_{c \in Syn_a, c \neq a} (n_{Ct_i}^c + \sum_{d \in Par_c, d \neq c} n_{Ct_i}^d)}{n_{Cf}^a + \sum_{b \in Par_a, b \neq a} n_{Cf}^b + \sum_{c \in Syn_a, c \neq a} (n_{Cf}^c + \sum_{d \in Par_c, d \neq c} n_{Cf}^d)}, \quad (1)$$

где  $n_{Ct_i}^a$ ,  $n_{Ct_i}^b$ ,  $n_{Ct_i}^c$  и  $n_{Ct_i}^d$  – абсолютные частоты встречаемости слов  $a$ ,  $b$ ,  $c$  и  $d$  в тематическом корпусе текстов  $Ct_i$ ;  $n_{Cf}^a$ ,  $n_{Cf}^b$ ,  $n_{Cf}^c$  и  $n_{Cf}^d$  – абсолютные частоты их появления в полном корпусе текстов  $Cf$ .

По аналогии с формулой (1) информативность вербально-ассоциативной связи слов  $a$  и  $b$  будем вычислять по формуле

$$I_{Ct_i}^{ab} = \frac{n_{Ct_i}^{ab} + \sum_{\substack{c \in Par_a, c \neq a \\ d \in Par_b, d \neq b}} n_{Ct_i}^{cd} + \sum_{\substack{r \in Syn_a, r \neq a \\ s \in Syn_b, s \neq b}} (n_{Ct_i}^{rs} + \sum_{\substack{p \in Par_r, p \neq r \\ q \in Par_s, q \neq s}} n_{Ct_i}^{pq})}{n_{Cf}^{ab} + \sum_{\substack{c \in Par_a, c \neq a \\ d \in Par_b, d \neq b}} n_{Cf}^{cd} + \sum_{\substack{r \in Syn_a, r \neq a \\ s \in Syn_b, s \neq b}} (n_{Cf}^{rs} + \sum_{\substack{p \in Par_r, p \neq r \\ q \in Par_s, q \neq s}} n_{Cf}^{pq})}, \quad (2)$$

где  $n_{Ct_i}^{ab}$ ,  $n_{Ct_i}^{cd}$ ,  $n_{Ct_i}^{rs}$  и  $n_{Ct_i}^{pq}$  – абсолютные частоты совместной встречаемости пар слов  $(a, b)$ ,  $(c, d)$ ,  $(r, s)$  и  $(p, q)$  в одном и том же предложении тематического корпуса текстов  $Ct_i$ ;  $n_{Cf}^{ab}$ ,  $n_{Cf}^{cd}$ ,  $n_{Cf}^{rs}$  и  $n_{Cf}^{pq}$  – абсолютные частоты их совместного появления в полном корпусе  $Cf$ .

Моделью поискового образа тематического корпуса текстов  $Ct_i$  является вербально-ассоциативная сеть предметной области, определяемой корпусом  $Ct_i$ .

Построим этот поисковый образ в виде множества вербально-ассоциативных пар слов, в котором словам каждой пары приписаны значения их информативности, а парам – значения информативности вербально-ассоциативных связей между их словами:

$$\omega(Ct_i) = \{(a, I_{Ct_i}^a); (b, I_{Ct_i}^b); I_{Ct_i}^{ab} \mid a \in Ct_i, b \in Ct_i, I_{Ct_i}^a > I_{Ct_i}^0, I_{Ct_i}^{ab} > I_{Ct_i}^{00}\}, \quad (3)$$

где  $\omega : Cf \rightarrow PO$  – инъективное отображение полного корпуса текстов  $Cf$  в множество  $PO$  их поисковых образов.

## 2.2. Алгоритм индексирования тематических корпусов текстов

Алгоритм 1. На входе алгоритма – тематический корпус текстов  $Ct_i$ , на выходе – поисковый образ корпуса  $Ct_i$  в виде выражения (3).

1. Сформировать множество  $W_{Ct_i}^{ab}$  всех вербально-ассоциативных пар слов вида  $(a, b)$  тематического корпуса текстов  $Ct_i$ .

2. Вычислить информативность  $I_{Ct_i}^a$  и  $I_{Ct_i}^b$  слов  $a$  и  $b$  всех вербально-ассоциативных пар слов из множества  $W_{Ct_i}^{ab}$  по формуле (1).

3. Вычислить информативность  $I_{Ct_i}^{ab}$  всех вербально-ассоциативных пар слов из множества  $W_{Ct_i}^{ab}$  по формуле (2).

4. Сформировать поисковый образ тематического корпуса текстов  $Ct_i$  в виде множества (3). Конец.

### 3. Индексирование политематических текстов

Различают монотематические и политематические тексты. Монотематический текст – это текст небольшого объема (статья, доклад), содержание которого отображает некоторую единую тематику. В политематическом тексте (например, книге) обычно представлено несколько монотематических текстов (субтекстов).

#### 3.1. Информативность слов и вербально-ассоциативных пар слов в политематическом тексте

Рассмотрим политематический текст  $T$  ( $T \in Cf$ ). Моделью поискового образа текста  $T$  является его вербально-ассоциативная сеть [1]. Представим эту сеть в виде совокупности вербально-ассоциативных пар слов

$$\omega(T) = \{ \langle (a, I_T^a); (b, I_T^b); I_T^{ab} \rangle \mid a \in T, b \in T, I_T^a > I_T^0, I_T^{ab} > I_T^{00} \}. \quad (4)$$

В этой совокупности словам  $a$  и  $b$  каждой пары соответствуют значения их информативности  $I_T^a$  и  $I_T^b$  в тексте  $T$ , а самой паре – значения информативности  $I_T^{ab}$  вербально-ассоциативных связей между этими словами.

Для вычисления информативности  $I_T^a$  произвольного слова  $a$  текста  $T$  будем использовать формулу, аналогичную формуле (1):

$$I_T^a = \frac{n_T^a + \sum_{b \in Par_a, b \neq a} n_T^b + \sum_{c \in Syn_a, c \neq a} (n_T^c + \sum_{d \in Par_c, d \neq c} n_T^d)}{n_{Cf}^a + \sum_{b \in Par_a, b \neq a} n_{Cf}^b + \sum_{c \in Syn_a, c \neq a} (n_{Cf}^c + \sum_{d \in Par_c, d \neq c} n_{Cf}^d)}, \quad (5)$$

где  $n_T^a$ ,  $n_T^b$ ,  $n_T^c$  и  $n_T^d$  – абсолютные частоты встречаемости слов  $a$ ,  $b$ ,  $c$  и  $d$  в тексте  $T$ .

По аналогии с формулой (2) информативность вербально-ассоциативной связи слов  $a$  и  $b$  в тексте  $T$  вычисляется по формуле

$$I_T^{ab} = \frac{n_T^{ab} + \sum_{\substack{c \in Par_a, c \neq a \\ d \in Par_b, d \neq b}} n_T^{cd} + \sum_{\substack{r \in Syn_a, r \neq a \\ s \in Syn_b, s \neq b}} (n_T^{rs} + \sum_{\substack{p \in Par_r, p \neq r \\ q \in Par_s, q \neq s}} n_T^{pq})}{n_{Cf}^{ab} + \sum_{\substack{c \in Par_a, c \neq a \\ d \in Par_b, d \neq b}} n_{Cf}^{cd} + \sum_{\substack{r \in Syn_a, r \neq a \\ s \in Syn_b, s \neq b}} (n_{Cf}^{rs} + \sum_{\substack{p \in Par_r, p \neq r \\ q \in Par_s, q \neq s}} n_{Cf}^{pq})}, \quad (6)$$

где  $n_T^{ab}$ ,  $n_T^{cd}$ ,  $n_T^{rs}$  и  $n_T^{pq}$  – абсолютные частоты совместной встречаемости пар слов  $(a, b)$ ,  $(c, d)$ ,  $(r, s)$  и  $(p, q)$  в одном и том же предложении текста  $T$ .

#### 3.2. Алгоритм индексирования политематических текстов

Алгоритм 2. На входе алгоритма – политематический текст  $T$ , на выходе – поисковый образ текста  $T$  в виде множества (4).

1. Сформировать множество  $W_T^{ab}$  всех вербально-ассоциативных пар слов вида  $(a, b)$  текста  $T$ .

2. Вычислить информативность  $I_T^a$  и  $I_T^b$  слов  $a$  и  $b$  всех вербально-ассоциативных пар слов из множества  $W_T^{ab}$  по формуле (5).

3. Вычислить информативность  $I_T^{ab}$  всех вербально-ассоциативных пар слов из множества  $W_T^{ab}$  по формуле (6).

4. Сформировать поисковый образ текста  $T$  в виде множества (4). Конец.

### 4. Индексирование монотематических текстов

Моделью поискового образа монотематического текста является его вербально-ассоциативная сеть, т. е. подграф сети некоторой предметной области, которой соответствует тематический (или динамический [2]) корпус текстов, релевантный индексируемому тексту. В связи с этим процесс его индексирования сводится к поиску тематического корпуса текстов или созданию динамического (рис. 2).



Рис. 2. Структурно-функциональная схема индексирования монотематического текста

Пусть  $Q$  – монотематический текст, который нужно проиндексировать, т. е. создать его поисковый образ в виде совокупности кортежей  $\{(a, I_Q^a); (b, I_Q^b); I_Q^{ab}) \mid a \in Q, b \in Q, I_Q^a > I_Q^0, I_Q^{ab} > I_Q^{00}\}$ , где  $(a, b)$  – вербально-ассоциативная пара слов;  $I_Q^a$  – информативность слова  $a$  текста  $Q$ ;  $I_Q^b$  – информативность слова  $b$ ;  $I_Q^{ab}$  – информативность вербально-ассоциативной связи между словами  $a$  и  $b$ ;  $I_Q^0$  и  $I_Q^{00}$  – пороговые значения информативности.

Для выявления статистических характеристик  $I_Q^a$ ,  $I_Q^b$  и  $I_Q^{ab}$  текста  $Q$  возможны две стратегии. Первая – поиск релевантного тематического корпуса текстов и вторая (в случае отрицательных результатов поиска) – создание динамического корпуса текстов путем отыскания релевантных текстов в полном корпусе.

#### 4.1. Поиск релевантного тематического корпуса текстов

Текст  $Q$  будем рассматривать как запрос на поиск релевантного ему тематического корпуса текстов. Исключим из всех поисковых образов документов полного корпуса текстов значения информативности слов и вербально-ассоциативных пар слов, т. е. преобразуем выражение (3) к виду

$$\omega(Ct_i) = \{(a, b) \mid a \in Ct_i, b \in Ct_i\}. \quad (7)$$

Аналогичным образом запишем поисковое предписание, т. е. поисковый образ текста  $Q$ :

$$\omega(Q) = \{(c, d) \mid c \in Q, d \in Q\}. \quad (8)$$

Поиск релевантного тематического корпуса текстов в полном корпусе  $Cf$  реализуется в два этапа. На первом этапе проводится поиск по поисковому предписанию (8). При этом используется оптимальная по релевантности (как показано в статье [2]) поисковая функция

$$\pi(Q) = \{Ct | \eta(\omega(Ct), \omega(Q)) < 0, Ct \in Cf\}, \eta(\omega(Ct), \omega(Q)) = \begin{cases} -1, & \text{если } \omega(Q) \cap \omega(Ct) \neq \emptyset; \\ 0, & \text{если } \omega(Q) \cap \omega(Ct) = \emptyset. \end{cases} \quad (9)$$

Поисковая функция – это частичное мультиотображение  $\pi : Z \rightarrow Cf \cup S_{\text{всб}}$  множества  $Z$  запросов в множество  $Cf$  документов полного корпуса текстов и веб-страниц Интернета  $S_{\text{всб}}$ . Отображение  $\eta : \omega(Cf \cup S_{\text{всб}}) \times \omega(Z) \rightarrow R$  в выражении (9) – это критерий выдачи ( $R$  – множество действительных чисел) [3], т. е. мера близости запросов и текстовых документов.

На втором этапе поиска релевантного тематического корпуса текстов из множества найденных выбирается корпус, которому соответствует наибольшее значение критерия выдачи. В большинстве известных информационных систем в качестве такого критерия применяется косинус угла между векторами поискового предписания и поискового образа документа в евклидовом пространстве  $E$ . Рассмотрим этот критерий, используя принятые выше обозначения.

Обозначим через  $W$  множество всех различных вербально-ассоциативных пар слов, входящих в поисковые образы текстов из множества  $Cf \cup S_{\text{всб}}$ , а также поисковых предписаний из множества  $\omega(Z)$ . Пусть их количество равно  $n$ . Лексикографически упорядочим все вербально-ассоциативные пары из множества  $W$ , т. е. представим  $W$  в виде кортежа  $W = \langle (a_1, b_1), (a_2, b_2), \dots, (a_n, b_n) \rangle$ . Для каждого проиндексированного тематического корпуса текстов  $Ct \in Cf$  построим вектор его поискового образа в пространстве  $E$ :  $F_{Ct} = (p_1, p_2, \dots, p_n)$ , где  $p_i = 1$ , если вербально-ассоциативная пара  $(a_i, b_i)$  входит в этот поисковый образ, в противном случае  $p_i = 0$ . Аналогично представим вектор поискового предписания, построенного для запроса  $Q$ :  $F_Q = (q_1, q_2, \dots, q_n)$ . Тогда для вычисления меры близости между векторами  $F_{Ct}$  и  $F_Q$  воспользуемся критерием выдачи

$$\cos \varphi = \frac{F_{Ct} F_Q}{|F_{Ct}| |F_Q|} = \frac{\sum_{i=1}^n p_i q_i}{\sqrt{\sum_{i=1}^n p_i^2} \sqrt{\sum_{i=1}^n q_i^2}}. \quad (10)$$

При реализации информационной системы критерий (10) целесообразно преобразовать следующим образом. Пусть  $l$  – количество совпавших вербально-ассоциативных пар поискового образа  $\omega(Ct)$  и поискового предписания  $\omega(Q)$ ,  $n_{Ct}$  – количество пар в множестве  $\omega(Ct)$ , а  $n_Q$  – их количество в множестве  $\omega(Q)$ . Тогда критерий (10) примет вид

$$\cos \varphi = \frac{l}{\sqrt{n_{Ct} n_Q}}. \quad (11)$$

Результатом поиска является тематический корпус текстов  $\pi(Q) = Ct$ , для которого значение  $\cos \varphi$  является наибольшим из всех значений, таких, что  $\cos \varphi \geq \eta_0$ . Если такой тематический корпус текстов не найден, то оперативно формируется динамический корпус.

#### 4.2. Создание динамического корпуса текстов

При формировании динамического корпуса текстов, релевантного запросу  $Q$ , в полном корпусе текстов  $Cf$  нужно найти все документы, релевантные тексту  $Q$ .

Пусть  $D \in Cf$  – произвольный документ из полного корпуса текстов. Построим вектор  $F_D$  поискового образа  $\omega(D)$  документа  $D$  по аналогии с вектором  $F_{Ct}$ :  $F_D = (d_1, d_2, \dots, d_n)$ . Компонента вектора  $d_k = 1$ , если соответствующая вербально-ассоциативная пара слов имеется в поисковом образе текста  $D$ , иначе  $d_k = 0$ . В качестве критерия выдачи будем использовать аналог критерия (10):

$$\cos \psi = \frac{F_D F_Q}{|F_D| |F_Q|} = \frac{\sum_{k=1}^n d_k q_k}{\sqrt{\sum_{k=1}^n d_k^2} \sqrt{\sum_{k=1}^n q_k^2}}. \quad (12)$$

Упростим критерий (12) и запишем его в виде, аналогичном выражению (11). Обозначим через  $r$  количество совпавших вербально-ассоциативных пар поискового предписания  $\omega(Q)$  и поискового образа  $\omega(D)$  документа  $D$ . Пусть также  $m_Q$  – количество пар в множестве  $\omega(Q)$ , а  $m_D$  – их количество в множестве  $\omega(D)$ . Тогда критерий (12) можно представить в виде

$$\cos \psi = \frac{r}{\sqrt{m_D m_Q}}. \quad (13)$$

Будем считать, что документ  $D \in Cf$  релевантен запросу  $Q$  и принадлежит создаваемому динамическому корпусу текстов  $Dt$ , если критерий (13) не меньше порогового значения  $\eta'_0$ , т. е.  $Dt = \pi(Q) = \{D \mid D \in Cf, \cos \psi \geq \eta'_0\}$ .

### 4.3. Алгоритм индексирования монотематических текстов

Обозначим через  $Kt$  ( $Kt \in \{Ct, Dt\}$ ) релевантный тексту  $Q$  тематический ( $Ct$ ) или динамический ( $Dt$ ) корпус текстов, а через  $W_Q$  – множество всех вербально-ассоциативных пар слов текста  $Q$ . Тогда поисковый образ текста  $Q$  будем строить в виде

$$\omega(Q) = \{(c, I_{Kt}^c); (d, I_{Kt}^d); I_{Kt}^{cd} \mid c \in W_Q, d \in W_Q, I_{Kt}^c > I_{Kt}^0, I_{Kt}^{cd} > I_{Kt}^{00}\}. \quad (14)$$

Алгоритм 3. На входе алгоритма – монотематический текст  $Q$ , на выходе – поисковый образ текста  $Q$  в виде множества (14).

1.  $Kt := \emptyset$ .
2. Преобразовать поисковое предписание  $\omega(Q)$  к виду (8).
3. Искать в полном корпусе текстов релевантные тематические корпуса в соответствии с поисковой функцией (9). Если хотя бы один тематический корпус найден, то перейти к п. 4, иначе – к п. 6.
4. Если в п. 3 найдено более одного тематического корпуса текстов, то выбрать один корпус с наибольшим значением критерия выдачи (11) и перейти к п. 5. Иначе перейти к п. 5.
5. Поместить найденный тематический корпус текстов в множество  $Kt$ .
6. Искать в полном корпусе текстов документы, релевантные запросу  $Q$ , в соответствии с критерием выдачи (13). Если хотя бы один документ найден, то перейти к п. 7, иначе – к п. 9.
7. Поместить все найденные в п. 6 документы в множество  $Kt$ .
8. Индексировать корпус текстов  $Kt$  (алгоритм 1). Представить поисковый образ корпуса  $Kt$  в виде выражения (14), полагая, что  $\omega(Q) = \omega(Kt)$ . Конец (поисковый образ монотематического текста  $Q$  сформирован).
9. Конец (текст  $Q$  не проиндексирован из-за недостаточной представительности полного корпуса текстов  $Cf$ , пополнить корпус текстов  $Cf$ ).

### Заключение

Предложенные в статье алгоритмы могут быть использованы при индексировании, поиске и реферировании текстовой информации в Интернете, корпоративных сетях и в локальных базах данных. При соответствующем подборе тематики и языка представления корпусов текстов возможны поиск и реферирование документов на различных входных языках. Реализация этой функции сводится к формированию корпусов текстов и созданию словарей базы знаний (без коррекции программного обеспечения системы). При наличии персональных тематических корпусов текстов обеспечивается адаптация процессов поиска и реферирования к информационным потребностям соответствующих индивидуальных и корпоративных пользователей.

### Список литературы

1. Липницкий, С.Ф. Модель представления знаний в информационных системах на основе вербальных ассоциаций / С.Ф. Липницкий // Информатика. – 2011. – № 4 (32). – С. 21–28.
2. Липницкий, С.Ф. Моделирование информационного поиска на основе динамических корпусов текстов / С.Ф. Липницкий, А.А. Мамчич // Весці НАН Беларусі. Сер. фіз.-тэхн. навук. – 2011. – № 1. – С. 72–81.
3. Липницкий, С.Ф. Моделирование информационного мониторинга Интернета на основе тематических корпусов текстов / С.Ф. Липницкий // Весці НАН Беларусі. Сер. фіз.-тэхн. навук. – 2011. – № 3. – С. 92–99.

Поступила 18.05.12

*Объединенный институт проблем  
информатики НАН Беларуси,  
Минск, Сурганова, 6  
e-mail: lipn@newman.bas-net.by*

**S.F. Lipnitsky**

**TEXT INFORMATION INDEXING  
ON THE BASIS OF VERBAL ASSOCIATIONS MODELING**

Algorithms of indexing text documents based on a mathematical model of knowledge representation in the form of verbal-associative networks are developed. Structural features of the knowledge base built by taking into account the simulation results are considered. Formulas for calculating words informativity and their verbal-associative pairs are proposed.

## АВТОМАТИЗАЦИЯ ПРОЕКТИРОВАНИЯ

УДК 658.512.011.56 : 621.9.06

Д.Н. Свирский

**ФОРМАЛИЗАЦИЯ РАННИХ СТАДИЙ ПРОЕКТИРОВАНИЯ  
МЕТАЛЛОРЕЖУЩЕГО ОБОРУДОВАНИЯ НА ОСНОВЕ  
QFD-МОДЕЛИРОВАНИЯ**

*Рассматривается проблема компьютерной поддержки ранних стадий проектирования металлообрабатывающих станков. Предлагается методическое обеспечение ее решения. Теоретические положения иллюстрируются примером макропроектирования заточного станка.*

**Введение**

Объективной основой конкурентоспособности продукции отечественного станкостроения является ее качество, понимаемое как точное соответствие изделия требованиям потребителя (целевого сегмента рынка). В свою очередь, качество выпускаемых станков закладывается на стадии проектирования путем выбора определенных инженерных решений. В связи с этим исключительно важную роль играют первые стадии творческого процесса, где формулируются принципиальные технические решения, далее подвергаемые детализации по каждому компоненту (узлу, блоку и т. п.) изделия. С конца прошлого века накоплен большой опыт автоматизации конструкторского труда в станкостроении [1, 2]. В настоящее время в процессе конструирования широко применяются мощные средства компьютерной графики и инженерного анализа. Однако техническое творчество на ранних стадиях проектирования до сих пор остается без реальной компьютерной поддержки. Поиску решения указанной проблемы посвящена эта работа.

**1. Постановка задачи**

В табл. 1 представлена временная организация ранних стадий проектирования. Исходя из нормативной модели конструкторской подготовки производства новой техники, обычно выполняются этапы предпроектных исследований, разработки технического задания и технического предложения, результаты которых документально оформляются в соответствии с государственными стандартами. В русле системного подхода, как правило, выделяют внешнее (или макро) проектирование, заключающееся в формулировании служебного назначения изделия и определении его основных технических характеристик, и внутреннее (или микро) проектирование, результатом которого является рабочая документация, удовлетворяющая требованиям технического задания [3]. В начале детальной разработки конструкции изделия на этапе внутреннего проектирования лежит формирование его концепции – «общего замысла», т. е. принципиальных инженерных решений, предопределяющих совершенство и экономичность технической реализации требований технического задания. Эти процедуры, результат которых отражается в техническом предложении, в настоящее время объединены понятием «концептуальное проектирование» [4].

Для эффективной компьютерной поддержки выполнения процедур ранних стадий проектирования станочного оборудования требуется решить ряд задач, основной из которых является разработка методологии формализации процедур концептуального проектирования технологического оборудования.

Таблица 1

Ранние стадии проектного процесса

<i>Ранние стадии проектирования</i>			Время →
Научно-технические и маркетинговые исследования	Техническое задание	Техническое предложение	
Макропроектирование		Концептуальное проектирование	
Функционально-квалиметрический анализ		Функционально-операторный синтез	Структурный синтез
Функционально-параметрический метод		Структурно-параметрический метод	
QFD-моделирование		Функционально-структурное моделирование	...

## 2. Предлагаемая методология

Результатом последовательного решения задач ранних стадий проектирования, в частности концептуального проектирования, является синтезированная структура станка в целом, оформленная в документации технического предложения. В ходе структурного синтеза определяют состав и взаимосвязи компонентов проектируемого объекта, а в качестве основы выступает функциональная модель (схема) изделия. Проектирование структуры новой машины проходит в следующей последовательности: цель→функции→структура. Поэтому для построения системы компьютерной поддержки принятия решений на ранних стадиях проектирования технологического оборудования целесообразно использовать функционально-структурный подход (см. строку 4 табл. 1), а из известных способов формализации процедур структурного синтеза (перебор, последовательный синтез и т. п.) рационально выбрать метод трансформации описаний. Он основан на преобразовании по определенным правилам описаний (моделей) объекта. Ключевым моментом формализации операции перехода от одной модели к другой является их параметризация. Цель, функции и структура системы (станка) могут быть описаны параметрически.

При системном подходе к объекту проектирования выделяют его внешние и внутренние функции. Внешние функции (главная и второстепенные) выполняются объектом в целом и отражают отношения между ним и сферой применения. Главная функция изделия синонимична служебному назначению, т. е. цели его создания. Второстепенные функции, например, способствуют его ликвидности. Внутренние функции определяются составом действий внутри объекта, выполняются его компонентами и обусловлены особенностями технологических и конструктивных решений. В зависимости от роли в рабочем процессе среди внутренних функций различают основные и вспомогательные. Основные функции служат для обеспечения работоспособности машины – осуществляют главную функцию, а вспомогательные способствуют их реализации. Так, при разработке станка для заточки концевых фрез внешними функциями являются обеспечение заданной величины биения на задней и передней поверхности; реализация заданного качества точности и т. п. В качестве внутренних выступают передача режущему инструменту и заготовке согласованного поступательно-вращательного движения, обеспечение минимального радиального и торцевого биения шпинделя и др.

Внешние функции, особенно главная, определяют качество и конкурентоспособность станка непосредственно, напрямую отвечая требованиям потребителя. Для параметрической оценки качества давно и успешно используется формальный аппарат квалиметрии. Так, внешние функции и потребительские требования относительно легко выражаются количественно с помощью функционально-квалиметрического анализа. При проектном структурировании станка внутренние функции как действия (операции) его компонентов также численно характеризуются параметрами производимого процесса, особенно конечными показателями его результата, путем функционально-операторного синтеза. Наконец, реализующие внутренние функции компоненты машины обязательно должны быть параметризованы значениями физических величин. Таким образом, трансформация функционального описания проектируемого объекта в его структурную модель должна быть формализована с помощью отображения станка в  $N_f$ -мерных пространствах их попарно общих параметров.

Формализацию перехода от внешних функций к внутренним можно эффективно осуществить QFD-моделированием [5, 6]. Методика Quality Function Deployment – развертывания (структурирования) функции качества – заключается в заполнении полей так называемого «дома качества» (рис. 1).

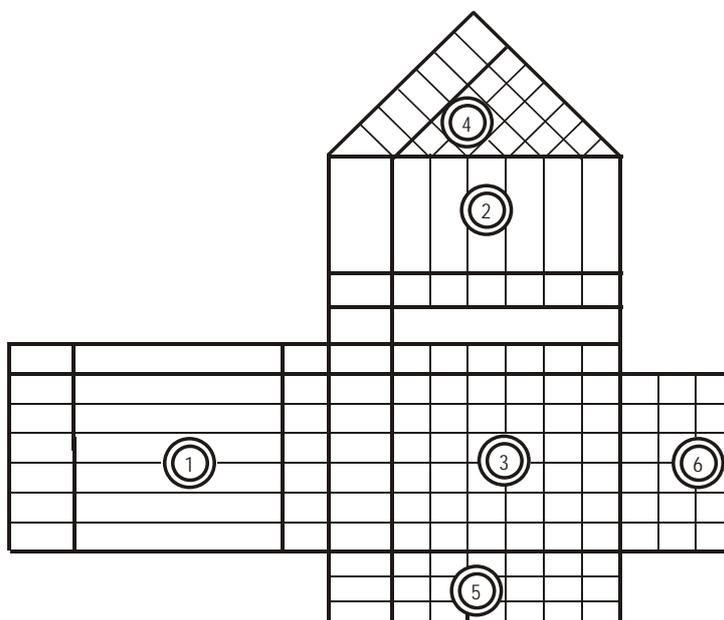


Рис. 1. Дом качества

В табл. 2 представлено поле 1 «дома качества», заполненное потребительскими требованиями (ПТ), ранжированными по результатам маркетинговых исследований. Сумма рангов ПТ равна единице.

Таблица 2  
Анализ потребительских требований (пример условный)

№ п/п	Потребительские требования	Ранг
1	Точность обработки	0,6
2	Уровень шума	0,2
3	Универсальность	0,09
4	Компактность	0,08
5	Эстетичность	0,03
...	...	
N		

В поле 2 располагаются показатели технического уровня станка – его технические характеристики (ТХ). Для металлорежущих станков регламентированы восемь групп показателей технического уровня станка. Поэтому общее их количество в матрице может достигать нескольких десятков.

Поле 3 отображает связь ПТ и ТХ. В частном случае известной зависимости ПТ от ТХ влияние ТХ на ПТ находится аналитически, но чаще приходится применять универсальный метод экспертных оценок. Удобно оценивать парную корреляцию числом из ряда  $-1; -0,5; 0; +0,5; +1$ , занося их в соответствующие ячейки поля 3. По тому же принципу заполняется поле 4 – «чердак дома». В эти ячейки записываются числа, определяющие взаимовлияние ТХ, так как изменение одной ТХ обычно ведет к изменению в ту или иную сторону значений других ТХ. При

заполнении полей 3 и 4 необходимо учитывать «скрытые воздействия» (инциденции) [7, 8] ПТ на ТХ и ТХ на ТХ.

Поле 5 отображает важность ТХ. Для расчета весовых коэффициентов используется формула

$$R_{ju} = \sum_{u=1}^N r_u \cdot x_{ju} ,$$

где  $j$  – порядковый номер технической характеристики (изменяется от 1 до  $M$ );  $u$  – порядковый номер потребительского требования (изменяется от 1 до  $N$ );  $x_{ju}$  – коэффициент, соответствующий зависимости ИХ от ПТ в поле 3;  $r_u$  – ранг  $u$ -го потребительского требования.

В ходе заполнения поля 6 выполняется основная процедура так называемого бенчмаркинга, а именно оценка конкурентоспособности своего продукта на целевом рынке. Из анализа предпочтения потребителей по каждому потребительскому требованию и степени его удовлетворенности качеством продукции складывается картина, которую можно назвать профилем производителя. Его целесообразно сравнивать с профилями ближайших по доли рыночного сегмента конкурентов.

Таким образом, информация, содержащаяся в полях 5 и 6, составляет важнейшую часть технического задания на разработку нового станка – ранжированные технические требования к нему. Если QFD-моделирование использовалось для анализа выпускаемой продукции, то его результаты могут быть применены для определения главных направлений повышения качества изделий и их конкурентоспособности.

Внутренние функции станка реализуются его узлами и механизмами. На каждом предприятии или в разрабатывающей организации есть их отработанные конструкторские решения, освоенные в производстве и используемые в качестве элементной базы выпускаемой продукции. Конструкция любого компонента станка параметрически описывается его технической характеристикой. Если значения показателей того или иного варианта конструкции узла соответствуют результатам QFD-моделирования, то он включается в разрабатываемую структурную схему станка. В противном случае необходимо создавать оригинальную конструкцию или покупать узел на стороне. Ориентиром всегда служат результаты QFD-моделирования.

При проектировании оригинальных узлов на стадии детальной разработки конструкции станка методику QFD-моделирования можно применять для определения параметров их ТХ. При этом роль ПТ исполняют соответствующие ТХ станка в целом.

Необходимо особо отметить важность экономических показателей при принятии принципиальных конструктивных решений. При этом учитывается современная мировая тенденция в ценообразовании – «target costing». Кратко отличие этой рыночно ориентированной методики от традиционного назначения цены на новую продукцию можно сформулировать следующим образом. В затратном варианте ценообразования рассчитывается себестоимость изделия, назначается норма прибыли, исходя из этого получается цена продажи: себестоимость + прибыль = цена. При управлении целевой себестоимостью расчетная формула имеет вид «цена – прибыль = себестоимость» [9]. Таким образом по результатам исследования рынка и бенчмаркинга выявляют актуальную и прогнозную цены на планируемую к выпуску продукцию, затем, отняв желаемую прибыль, получают требуемую себестоимость. Приняв за основу значение себестоимости станка методами функционально-стоимостного анализа и обратных вычислений [10] далее можно определить допустимые затраты на любой из узлов исходя из его важности в станке.

Изложенные выше теоретические соображения легли в основу автоматизированной системы поддержки принятия решений при проектировании универсального заточного станка, функционально-структурная модель которого показана на рис. 2. Станок обладает развитой кинематической структурой (рис. 3), реализуемой модульными компонентами (рис. 4). Компонировка станка (рис. 5) иллюстрирует его техническое предложение.



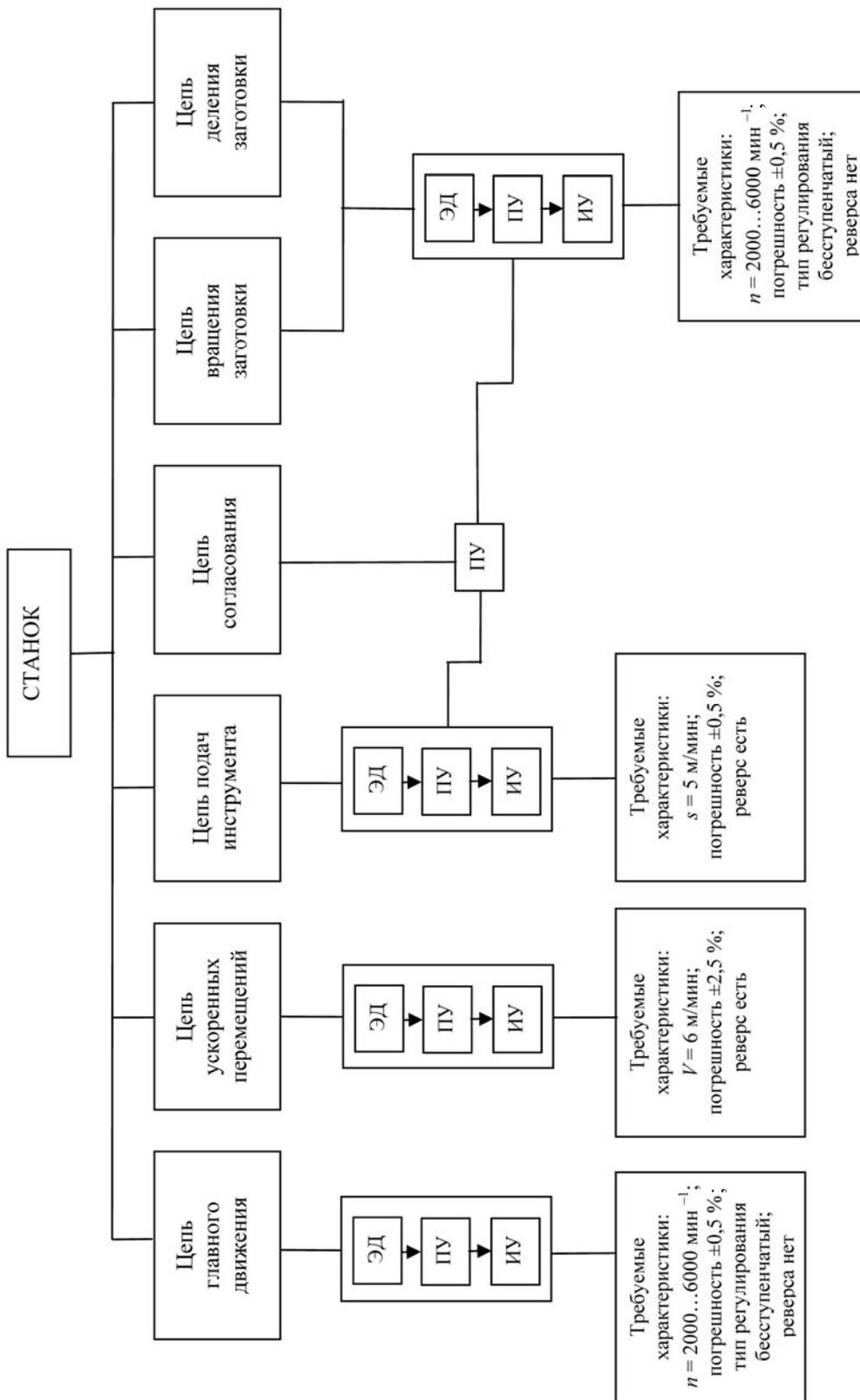


Рис. 3. Структурная модель кинематики станка: ЭД – электродвигатель; ПУ – передаточное устройство; ИУ – исполнительное устройство

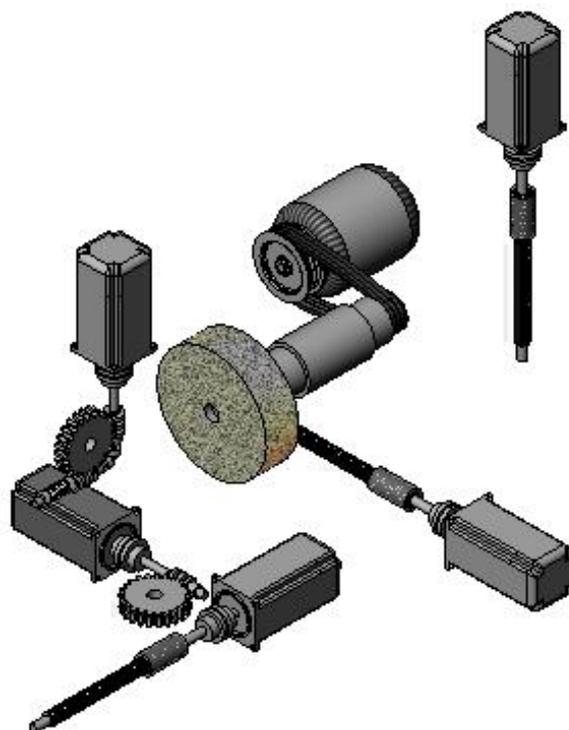


Рис. 4. Синтез кинематической структуры станка из компонентов элементной базы

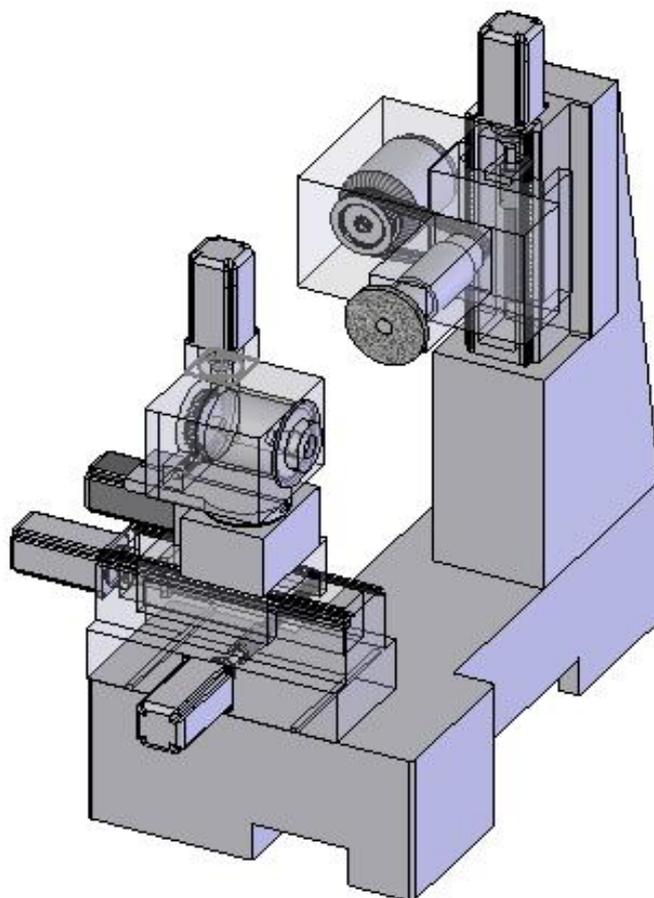


Рис. 5. Компоновка заточного станка

### Заключение

Разработанное программно-методическое обеспечение компьютерной поддержки принятия решений на ранних стадиях проектирования металлорежущего оборудования используется в учебном процессе при подготовке инженеров соответствующей специальности [11]. В настоящее время система автоматизации ранних стадий проектирования проходит отладку для ее предложения станкостроительным предприятиям.

### Список литературы

1. Васильев, Г.Н. Автоматизация проектирования металлорежущих станков / Г.Н. Васильев. – М. : Машиностроение, 1987. – 280 с.
2. Прохоров, А.Ф. Конструктор и ЭВМ / А.Ф. Прохоров. – М. : Машиностроение, 1987. – 272 с.
3. Вязгин, В.А. Математические методы автоматизированного проектирования / В.А. Вязгин, В.В. Федоров. – М. : Высш. школа, 1989. – 184 с.
4. Ивахненко А.Г. Методология концептуального проектирования металлорежущих систем / А.Г. Ивахненко, А.В. Пуш // СТИН. – 1998. – № 4. – С. 3–6.
5. Ficalora, J.P. Quality Function Deployment & Six Sigma: A QFD Handbook / J.P. Ficalora, L. Cohen. – Prentice Hall, 2009. – 480 p.
6. Svirsky, D. QFD-Method Integration in tensor concept of collective intellect support system for compact product organisation / D. Svirsky // Proc. 8th Intern. Symp. on QFD. – Munich : German QFD Institute, 2002. – P. 135–146.
7. Кофман, А. Модели для исследования скрытых воздействий / А. Кофман, Х. Хил Алуха. – Минск : Выш. школа, 1993. – 160 с.
8. Svirsky, D. Decision making process formalizing for the competitive machine tool conceptual design / D. Svirsky, A. Firsov // Proc. 9th Int. conf. on Pattern Recognition and Information Processing. – Minsk : UIIP NAS Belarus, 2007. – Vol. 2. – P. 188–192.
9. Мерзликина, Г.С. Концепция формирования и управления целевой себестоимостью (target costing) / Г.С. Мерзликина, С.В. Волков // Управленческий учет. – № 8. – 2008. – С. 34–39.
10. Одинцов, Б.Е. Обратные вычисления в формировании экономических решений / Б.Е. Одинцов. – М. : Финансы и статистика, 2004. – 192 с.
11. Свирский, Д.Н. Определение технических характеристик проектируемого станка. Методические указания к лабораторным работам по курсу «Конструирование и расчет станков» / Д.Н. Свирский, А.С. Фирсов. – Витебск : ВГТУ, 2005. – 44 с.

Поступила 24.04.12

*Объединенный институт проблем  
информатики НАН Беларуси,  
Минск, Сурганова, 6  
e-mail: svirsky@newman.bas-net.by*

**D.N. Svirsky**

### **EARLY STAGES DESIGN FORMALIZATION OF THE METAL-CUTTING EQUIPMENT ON THE BASE OF QFD-MODELING**

The problem of computer support of early stages design of machine-tools is considered. A solution methodological base is offered. A case study of tool-grinding machine macro-design illustrates the theoretical developments.

УДК 621.374.3; 004.4:004.9

Д.И. Черемисинов

## ГЕНЕРАЦИЯ ВЫПОЛНИМЫХ СПЕЦИФИКАЦИЙ ЦИФРОВЫХ СИСТЕМ ИЗ СТРУКТУРНЫХ ОПИСАНИЙ FPGA-ПРОЕКТОВ

*Описывается алгоритм конвертации формата XDL в промежуточный формат описания реализованного на FPGA проекта. Моделью промежуточного представления служит двудольный граф. Алгоритмическим базисом операции конвертации является подстановка графов, задающая распаковку элемента FPGA в примитивные логические элементы. Проводится методика испытаний этого алгоритма, для реализации которой разработана программа конвертации промежуточного представления в формат EDIF.*

### Введение

Проектировщик цифрового устройства должен построить его функциональное и структурное описание. Обычно задается только одна форма спецификации, другая строится в ходе проектирования с применением САПР. Как правило, исходными данными для процесса проектирования является функциональное описание, по которому во время логического синтеза и технологического проектирования строится структурное описание. Нужда в преобразовании структурного описания в функциональное возникает при верификации результатов проектирования.

Структурная модель (netlist) представляет систему в терминах взаимосвязей составляющих компонентов и не содержит ничего кроме компонентов и цепей с их атрибутами. В своем основном значении структурное описание есть описание *внутреннего устройства* чего-либо. Структурное описание устройства является взглядом наблюдателя изнутри системы, характеризующим ее организацию через элементы (части) и их взаимозависимости. Функциональное описание системы является взглядом внешнего наблюдателя, определяющим его взаимодействие с окружающей средой через соотношения между входами и выходами. Структурное описание выполнимо (может моделироваться), если известны функциональные модели компонентов; функциональное описание – это описание законов функционирования, эволюции системы, алгоритмов ее поведения, оно моделируемо само по себе по определению. Таким образом, описания поведения при помощи понятий или структуры или функции содержат фундаментальные различия.

Для представления структурных моделей в современных САПР используются специальные текстовые языки описания данных, называемые форматами структурных описаний. Формальной моделью структурных описаний являются графы.

### 1. Форматы структурных описаний FPGA

Использование элемента в описании называют экземпляром элемента (instance). Структурное описание содержит (или ссылается на) описания элементов или блоков устройства, и, таким образом, каждый экземпляр имеет «определение». В этих определениях обычно перечисляются точки связи, используемые при подключении элемента к другим элементам устройства. Точки подключения связей называют выводами (port) элемента. Выводы считаются частью структуры экземпляра элемента. Цепи (net) символизируют «провода», которыми подключаются экземпляры элементов в устройстве.

Структурная модель может быть задана в виде списка экземпляров (instance based netlist) или в виде списка цепей (net based netlist).

Список экземпляров задает структуру соединений, используемых в устройстве, указанием для каждого экземпляра списка пар, содержащих вывод экземпляра и цепь, с которой связан этот вывод, – списка подключения выводов элементов (port map). В этом виде описания список

цепей нужно собирать, анализируя списки подключения выводов элементов. Примером задания структуры соединений списком экземпляров является структурный стиль VHDL [1].

Список цепей задает структуру соединений, используемых в устройстве, указанием, во-первых, всех экземпляров элементов и, во-вторых, всех цепей, которые задаются перечислением выводов экземпляров, связанных этой цепью. Примером задания структуры соединений списком экземпляров является формат EDIF [2].

EDIF – формат netlist общего назначения, принятый в качестве стандарта в области САПР интегральных схем. EDIF имеет Lisp-подобный синтаксис. В настоящее время наиболее широко используется версия 2.0.0 этого формата, позволяющая описывать структуру проектируемых устройств. Известные производители САПР электронных устройств, такие как Cadence, Mentor Graphics, Altera и др., в своих САПР предусматривают возможность использования EDIF.

Предложения языка EDIF имеют иерархическую структуру и построены из конструкций, называемых секциями. Все секции имеют стандартную форму. Текст секции заключен в скобки и состоит из названия секции, обычно заданного ключевым словом языка EDIF, и тела секции. Тело чаще всего представляет собой последовательность секций более низкого уровня иерархии. Тела секций нижнего уровня иерархии содержат непосредственно данные.

Содержательно описание проектного решения на EDIF состоит из множества библиотек (library), которые, в свою очередь, состоят из элементов (cell). Каждый элемент представлен в библиотеке в одном или нескольких видах (view). Обычно используется один тип view – netlist. Задание view состоит из интерфейса (interface) и содержания (contents). Секция интерфейса задает определение элемента – список его выводов, а секция содержания представляет собой структурное описание элемента в виде списка цепей. Вначале перечисляются экземпляры, а затем цепи.

Для описания устройств с большим количеством экземпляров общепринятой практикой является использование иерархии. Иерархическое описание разбито на блоки. Каждый блок становится «определением», экземпляры которого используются в структурном описании. Определение, которое не включает экземпляров, называют *примитивом* в данном структурном описании, тогда как определение, которое включает экземпляры других элементов, является иерархическим – детализованным. В «свернутом» (folded) иерархическом описании допускается использование нескольких экземпляров одного детализованного определения. В «развернутом» (unfolded) иерархическом описании не допускается больше одного экземпляра детализованного определения. Структурное описание называется «плоским» (flat), если все экземпляры являются примитивами.

В процессе проектирования устройств на основе Xilinx FPGA результирующее структурное описание представлено в формате NCD (Native Circuit Description) [3], описание которого является технологическим секретом Xilinx. Для доступа к структурному представлению на этом уровне Xilinx предлагает текстовый формат XDL, частично описанный в [4]. Формат XDL является плоским структурным описанием в виде списка цепей.

Обобщение (generalization) – это вид отношений между общим описанием и специфическим, которое основывается на общем и детализирует его. Специфическое описание полностью согласуется с общим (имеет то же поведение), но обладает также и некоторой дополнительной информацией. Обобщение связывает общую сущность (родителя) с ее конкретным воплощением (потомком). Обобщения иногда называют отношениями типа «является», имея в виду, что одна сущность является частным (детализованным) выражением другой, более общей. Обобщение означает, что специфическое описание может заменить общее описание, но не наоборот. Другими словами, потомок может быть подставлен вместо родителя. Операция замены общего структурного описания более детальным структурным описанием в соответствии с заданным отношением обобщения – это *распаковка* исходного описания.

Смысл распаковки состоит в построении описания, которое, оставаясь структурным, позволяет получить функциональное (выполнимое) описание. В детализованном описании названия типа элементов позволяют определить функцию – отношение вход-выход – элемента. Поэтому такое описание может быть превращено в выполнимое путем суперпозиции этих функций в соответствии со структурой соединений.

Информация об отношении обобщения структурного описания [5], соответствующего размещенной в FPGA схеме, содержится в файле отчета для соответствующего типа (part)

FPGA Xilinx. Программа *xdl* в САПР Xilinx имеет опции, позволяющие построить файлы отчета (с расширением .XDLRC). Отношение обобщения задано в файле отчета секцией *primitive\_defs*, содержащей зависимости для каждого типа блока родительского описания (рис. 1).



Рис. 1. Зависимость отношения обобщения для блока BUFGMUX из файла отчета

Описание примитива формата XDL в файле отчета – это плоское структурное описание в виде списка экземпляров. Иллюстрация отношения обобщения для блока SLICEL приведено в виде схемы [5, рис. 4-2] в руководстве пользователя FPGA Xilinx семейства Spartan-3.

## 2. Модель структуры устройства для операции распаковки

Моделью структурного описания в формате XDL выбран двудольный граф, одной долей которого являются порты (выводы) элементов FPGA, а второй долей – цепи, соединяющие порты.

В качестве примера на рис. 2 изображен фрагмент логической схемы секции SLICEL, а на рис. 3 – модель этого фрагмента в виде неориентированного двудольного графа. В данном фрагменте XORG – логический элемент сумма по модулю 2, GYMUX – трехвходовый коммутационный элемент, DYMUX – двухвходовый коммутационный элемент.

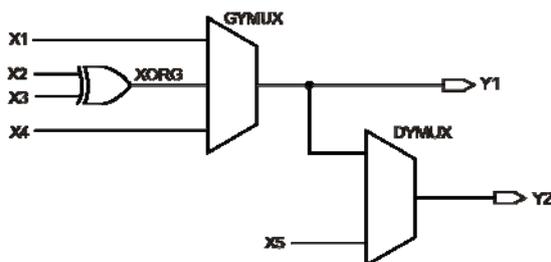


Рис. 2. Логическая схема (фрагмент секции SLICEL)

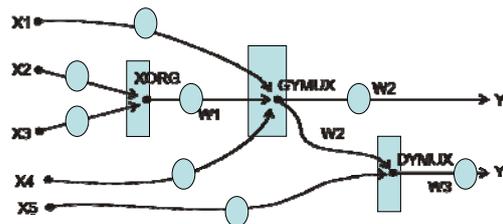


Рис. 3. Неориентированный двудольный граф, соответствующий логической схеме

На рисунках графов вершины, являющиеся выводами элемента, сгруппированы в прямоугольники (сами прямоугольники не являются элементами графа). Обозначения выводов предваряются именем элемента, показанным на рисунках в фигуре элемента (см. рис. 3). Вершины цепей показаны кружками. Обозначения (имена) выводов и некоторых цепей для упрощения рисунка опущены.

### 3. Распаковка блоков FPGA Spartan 3

В качестве алгоритмического базиса для решения задачи распаковки формата XDL удобно использовать подстановки графов, основой которых являются операции поиска и замены подграфов. Подстановка графов задает правило переписывания графа в форме «образец – действие» [6].

Алгоритмические системы на основе правил переписывания в форме «образец – действие» позволяют описывать любые алгоритмически представимые вычисления, так как в их основе лежит алгоритмически полная система нормальных алгоритмов Маркова [7]. Очевидно, свойства этой системы сохраняются при ее использовании для вычислений на графах. Функциональные возможности этой системы ограничиваются использованием единственной операции – сопоставления с образцом для построения любых алгоритмов. В теории алгоритмов Маркова эта операция называется *подстановкой слов*. Подстановка состоит из двух частей: левой и правой. Подстановка применяется к слову следующим образом: в слове ищется подслово, совпадающее с левой частью подстановки (образцом). Если такое подслово находится, то оно заменяется правой частью подстановки.

В основе алгоритма преобразования структурного описания, построенного на базе операций сопоставления с образцом, лежит та же идея, что и в алгоритме Маркова, когда берется список образцов (подсхем) с действиями замены и выполняются действия везде, где соответствующие образцы встречаются в некотором большом структурном описании.

*Граф  $g$*  – неориентированный двудольный граф, построенный в результате синтаксического анализа структурного описания в формате XDL. Вершины графа  $g$  помечены. Граф  $H$  называется *подграфом* (или *частью*) графа  $G$ , если  $VH \subseteq VG$ ,  $EH \subseteq EG$ . Если множество вершин подграфа  $H$  есть  $U$ , а множество его ребер совпадает с множеством всех ребер графа  $G$ , концы которых принадлежат  $U$ , то  $H$  называется *подграфом, порожденным множеством  $U$* . Определим двуместную операцию *склейки* вершин как отождествление двух заданных вершин пары графов, не имеющих общих элементов. Можно обобщить эту операцию как склейку графов, указывая множество пар склеиваемых вершин.

Число ребер, инцидентных некоторой вершине  $v$ , называется *степенью* вершины и обозначается  $\deg v$ , вершина степени 1 называется *концевой* (*висячей*). Ребро, инцидентное концевой вершине, также называется *концевым*. В графе  $g$ , построенном в результате синтаксического анализа структурного описания в формате XDL, все вершины-выводы являются висячими, а вершины-цепи имеют  $\deg v \geq 2$  (см. рис. 3).

*Подстановка графов* – это правило, заданное с помощью  $g_j \rightarrow g_r$ , где  $g_j$  и  $g_r$  – графы. Граф  $g_j$  называется образцом подстановки, а  $g_r$  – правой частью подстановки. Подграф графа  $g$ , изоморфный графу  $g_j$ , должен быть заменен графом, изоморфным  $g_r$ . Эта операция называется применением подстановки. *Операция применения подстановки* использует модель приклеивания, в которой механизм применения подстановки задается отображением склеивания. *Функция трансформации меток* вычисляет значения меток при применении операции подстановки. Задает процедуру вычисления меток в подграфе  $g_r^{host}$  по значениям меток в подграфе  $g_j^{host}$ .

Введем следующие определения:

$g$  – *исходный граф*, к которому применяется правило подстановки;  
 $g_j^{host}$  – подграф графа  $g$ , изоморфный  $g_j$ . *Остаток графа* – это граф  $g - g_j^{host}$  (оператор « $\rightarrow$ » обозначает удаление из  $g$  всех вершин и ребер  $g_j^{host}$  и всех ребер, одна или обе конечные точки которых принадлежат  $g_j^{host}$ );  
 $g_r^{host}$  – подграф, изоморфный  $g_r$  и используемый при подстановке вместо  $g_j^{host}$ . *Исходное множество ребер связи* – множество ребер, соединяющих  $g_j^{host}$  с остатком графа (эти ребра

«подключают»  $g_i^{host}$  к  $g$ ). Множество ребер связи после подстановки – множество ребер, соединяющих  $g_r^{host}$  с остатком графа.

Чередующаяся последовательность  $v_1, e_1, v_2, e_2, \dots, v_l, e_l, v_{l+1}$  вершин и ребер графа, такая, что  $e_i = v_i v_{i+1}$  ( $i = 1, 2, \dots, l$ ), называется *путем, соединяющим вершины  $v_i$  и  $v_{i+1}$*  (или  $(v_i, v_{i+1})$ -путем). Очевидно, что путь можно задать последовательностью  $v_1, v_2, \dots, v_l, v_{l+1}$  его вершин, а также последовательностью его ребер  $e_1, e_2, \dots, e_l$ . Граф называется *связным*, если любые две несовпадающие вершины в нем соединены путем. В графе  $g$ , построенном в результате синтаксического анализа структурного описания в формате XDL, число компонент связности равно числу цепей.

### 3.1. Подстановка распаковки

Выводы элементов при распаковке удобно считать висячими вершинами. Образцом подстановки распаковки является граф, множество вершин которого состоит из выводов элемента заданного типа, а множество ребер пустое. Правая часть подстановки является двудольным графом, построенным в результате синтаксического анализа соответствующей подсекции секцией *primitive\_defs* из файла отчета (с расширением .XDLRC). Для иллюстрации на рис. 4 приведена подстановка распаковки элемента BUFGMUX [5].

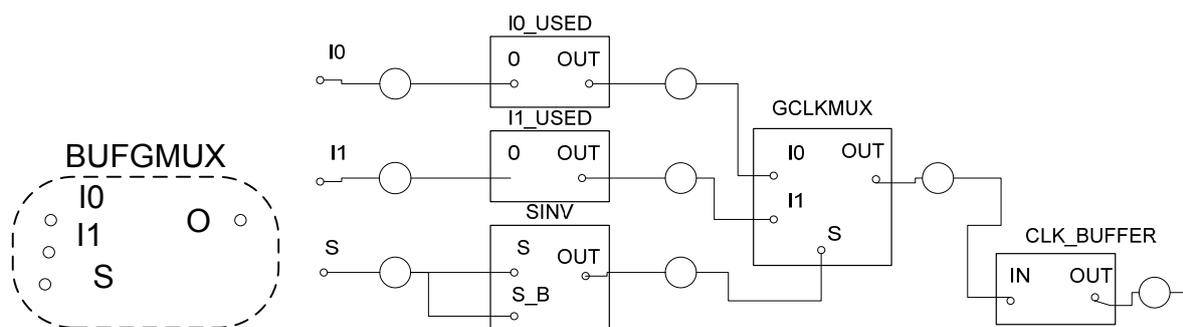


Рис. 4. Подстановка распаковки элемента BUFGMUX

Граф правой части подстановки строится в ходе анализа подсекции BUFGMUX секцией *primitive\_defs* из файла отчета. Все выводы элементов включаются в долю выводов графа. Списки соединений преобразуются в долю цепей и задают ребра, связывающие выводы и цепи.

Операция применения подстановки распаковки выполняется в два этапа. На первом этапе все вершины подграфа в исходном графе  $g$ , выделенного образцом подстановки, склеиваются с одноименными внешними выводами графа  $g$ , правой части подстановки. Не подсоединенные в графе  $g$  выводы элемента  $g$ , не включаются в результат подстановки. Цепи с  $\deg v < 2$  тоже не включаются в результат подстановки. На втором этапе склеиваются вершины-цепи  $g$ , инцидентные некоторой вершине и полученные в результате склеивания на предыдущем этапе, а сама эта вершина удаляется из графа  $g$ . Данная операция выполняется для каждого вывода распаковываемого элемента.

Процесс применения подстановки распаковки элементов типа BUFGMUX к фрагменту схемы (рис. 5) состоит в следующем. Единственный подграф, соответствующий образцу подстановки распаковки элементов этого типа, в этом фрагменте выделен пунктиром. После приклеивания вершин-выводов из правой части подстановки к вершинам, выделенным образцом, граф фрагмента принимает вид, показанный на рис. 6. Затем одна за другой удаляются вершины-выводы и склеиваются инцидентные удаляемой вершине вершины-цепи. Для вывода I0 склеиваемые цепи показаны жирной дуговой стрелкой.



Анализируя строки конфигурации в элементах FPGA, можно заметить, что значительная часть составляющих элементов имеет значение #OFF в подстроке конфигурации, т. е. находится в выключенном состоянии. С точки зрения экономии вычислительной работы выгодно при реализации подстановки использовать инверсный процесс формирования правой части подстановки. В прямом процессе реализация распаковки начинается с полного графа элементов примитива и их связей с исключением в дальнейшем ненужного. В инверсном процессе правая часть подстановки вначале содержит используемые в данной конфигурации элементы, которые можно найти, анализируя строку конфигурации, затем эти элементы дополняются цепями в соответствии с описанием структуры примитива в файле отчета.

Покажем этот процесс формирования правой части подстановки распаковки на примере элемента circ3/Q типа SLICEL [5]. Для этого элемента задана следующая строка конфигурации:

```
cfg " BXINV::BX BYINV::#OFF CEINV::#OFF
CLKINV::CLK COUTUSED::#OFF CY0F::#OFF
CY0G::#OFF CYINIT::#OFF CYSELF::#OFF
CYSELG::#OFF DXMUX::1 DYMUX::#OFF
F:circ0/circ1/d31:#LUT:D=(A1*(A4*(~A3+~A2)))
F5USED::#OFF FFX:circ3/Q:#FF
FFX_INIT_ATTR::INIT0 FFX_SR_ATTR::SRLOW
FFY::#OFF FFY_INIT_ATTR::#OFF
FFY_SR_ATTR::#OFF FXMUX::F5
FXUSED::#OFF
G:circ0/circ1/d32:#LUT:D=((~A3*(A4*A1))+A3*((~A2*(A4*A1))+(A2*(A4+A1))))
GYMUX::#OFF REVUSED::#OFF SRINV::#OFF
SYNC_ATTR::ASYNC XBUSED::#OFF
XUSED::#OFF YBUSED::#OFF YUSED::#OFF
F5MUX:circ0/circ1/d3_f5: "
```

Подстроки активных элементов перечислены ниже. Всего в этом элементе используется восемь компонентов: BXINV, CLKINV, DXMUX, F, FFX, FXMUX, G, F5MUX.

```
BXINV::BX
CLKINV::CLK
DXMUX::1
F:circ0/circ1/d31:#LUT:D=(A1*(A4*(~A3+~A2)))
FFX:circ3/Q:#FF
FXMUX::F5
G:circ0/circ1/d32:#LUT:D=((~A3*(A4*A1))+A3*((~A2*(A4*A1))+(A2*(A4+A1))))
F5MUX:circ0/circ1/d3_f5: "
```

Эти компоненты в файле отчета имеют следующие описания. Цепи соединений с активными компонентами или с выводами элемента выделены жирным шрифтом.

```
(element BXINV 3
  (pin BX_B input)
  (pin BX input)
  (pin OUT output)
  (cfg BX_B BX)
  (conn BXINV OUT ==> CYINIT BX)
  (conn BXINV OUT ==> DXMUX 0)
  (conn BXINV OUT ==> CY0F BX)
  (conn BXINV OUT ==> F5MUX S0)
  (conn BX BX ==> BXINV BX_B)
  (conn BX BX ==> BXINV BX)
)
(element CLKINV 3
  (pin CLK_B input)
  (pin CLK input)
  (pin OUT output)
  (cfg CLK_B CLK)
  (conn CLKINV OUT ==> FFX CK)
  (conn CLKINV OUT ==> FFY CK)
  (conn CLK CLK ==> CLKINV CLK_B)
  (conn CLK CLK ==> CLKINV CLK)
)
```

```

(element DXMUX 3
    (pin 0 input)
    (pin 1 input)
    (pin OUT output)
    (cfg 0 1)
    (conn DXMUX OUT ==> FFX D)
    (conn BXINV OUT ==> DXMUX 0)
    (conn FXMUX OUT ==> DXMUX 1)
)
(element F 5
    (pin A1 input)
    (pin A2 input)
    (pin A3 input)
    (pin A4 input)
    (pin D output)
    (cfg <eqn>)
    (conn F D ==> XORF 0)
    (conn F D ==> CYSELF F)
    (conn F D ==> FXMUX F)
    (conn F D ==> F5MUX F)
    (conn F1 F1 ==> F A1)
    (conn F2 F2 ==> F A2)
    (conn F3 F3 ==> F A3)
    (conn F4 F4 ==> F A4)
)
(element FFX 6
    (pin CK input)
    (pin CE input)
    (pin D input)
    (pin Q output)
    (pin SR input)
    (pin REV input)
    (cfg #FF #LATCH)
    (conn FFX Q ==> XQ XQ)
    (conn CLKINV OUT ==> FFX CK)
    (conn CEINV OUT ==> FFX CE)
    (conn DXMUX OUT ==> FFX D)
    (conn SRINV OUT ==> FFX SR)
    (conn REVUSED OUT ==> FFX REV)
)
(element FXMUX 4
    (pin F5 input)
    (pin F input)
    (pin FXOR input)
    (pin OUT output)
    (cfg F5 F FXOR)
    (conn FXMUX OUT ==> XUSED 0)
    (conn FXMUX OUT ==> DXMUX 1)
    (conn F5MUX OUT ==> FXMUX F5)
    (conn F D ==> FXMUX F)
    (conn XORF 0 ==> FXMUX FXOR)
)
(element G 5
    (pin A1 input)
    (pin A2 input)
    (pin A3 input)
    (pin A4 input)
    (pin D output)
    (cfg <eqn>)
    (conn G D ==> XORG 0)
    (conn G D ==> CYSELG G)
    (conn G D ==> F5MUX G)
    (conn G D ==> GYMUX G)
    (conn G1 G1 ==> G A1)
    (conn G2 G2 ==> G A2)
    (conn G3 G3 ==> G A3)
    (conn G4 G4 ==> G A4)
)

```

```

(element F5MUX 4
  (pin F input)
  (pin G input)
  (pin OUT output)
  (pin S0 input)
  (conn F5MUX OUT ==> F5USED 0)
  (conn F5MUX OUT ==> FXMUX F5)
  (conn F D ==> F5MUX F)
  (conn G D ==> F5MUX G)
  (conn BXINV OUT ==> F5MUX S0)
)

```

Сформированный из выводов элементов и выделенных цепей двудольный граф показан на рис. 7.

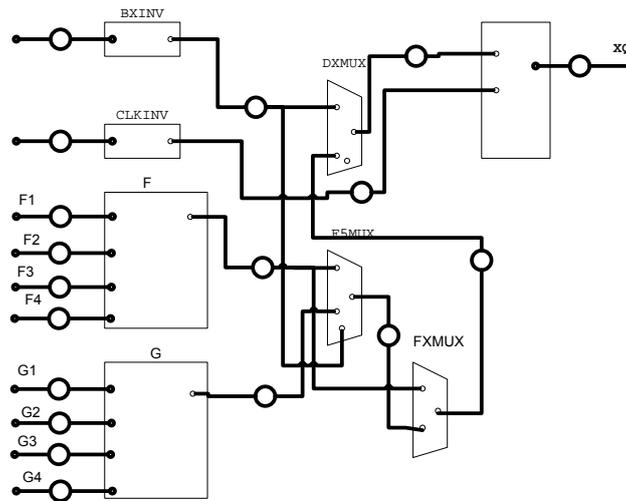


Рис. 7. Структура элемента sig3/Q

Непосредственно при построении графа на рис. 7 можно учесть программирование элементов. Вид правой части подстанции, используемой при распаковке элемента sig3/Q, изображен на рис. 8.

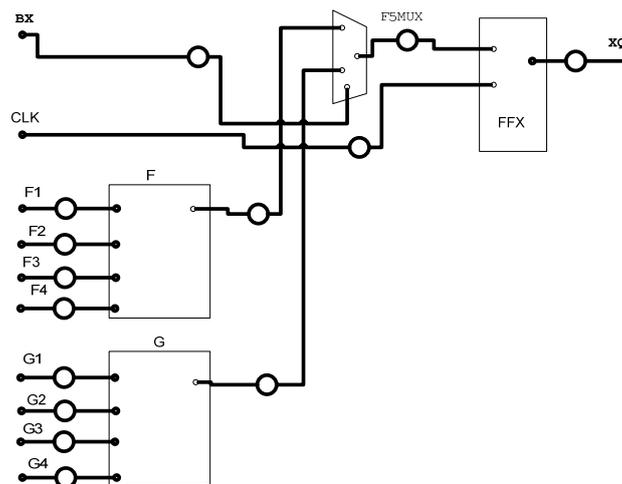


Рис. 8. Правая часть подстанции распаковки элемента sig3/Q

### 3.3. Алгоритм распаковки

1. Просматриваются предложения net из исходного описания в формате XDL и строится двудольный граф структурного описания.

2. Для каждого предложения *inst* из исходного описания выполняются следующие действия.

2.1. Используя строку конфигурации рассматриваемого предложения *inst*, с помощью инверсного процесса строится с учетом программирования двудольный граф правой части подстановки.

2.2. К выводам экземпляра с именем рассматриваемого предложения *inst* (к подграфу, составленному из этих выводов) применяется построенная на шаге 2.1 подстановка графов.

2.3. Подставленные при распаковке экземпляры элементов имеют имена типов, являющиеся именами составляющих элементов из описания примитива в файле отчета, используемого для построения правой части подстановки. В качестве имени экземпляра используется имя исходного экземпляра, дополненное именем типа. Например, распаковывается экземпляр *circ2/Q*, именем экземпляра LUT *G* будет *circ2/Q:G*. Если в строке конфигурации для элемента указан второй компонент параметра, то эта строка тоже добавляется к имени элемента. Например, LUT *G* *circ2/Q* имеет параметр программирования

$$G:circ0/circ1/d2\_F:\#LUT:D=(\sim A2*(A4@A1))+(A2*(A4*A3)).$$

Полное имя экземпляра, полученного в результате подстановки, будет иметь вид *circ2/Q:G:circ0/circ1/d2\\_F*. Имена вновь образуемых цепей строятся на основе имени элемента, являющегося источником в цепи. Это имя дополняется именем вывода выхода.

Уточненное структурное описание, полученное в результате распаковки исходного структурного описания, может быть дополнено функциональными описаниями составляющих элементов, что превращает структурное описание в функциональное. Соответствующее преобразование – это оценка (*evaluation* – вычислить числовое значение, выразить в цифровой форме).

Этап оценки состоит в назначении логических функций элементам структурного описания, полученного в результате распаковки. Для комбинационных элементов логическая функция определяется названием элемента. Элементы *F* и *G* являются комбинационными, если значение параметра программирования *value* из подстроки конфигурации начинается с «*#LUT*». Например, элемент *G* в конфигурации элемента *circ3/Q* задан следующей строкой:

$$G:circ0/circ1/d32:\#LUT:D=((\sim A3*(A4*A1))+A3*((\sim A2*(A4*A1))+(A2*(A4+A1))))).$$

Здесь *circ0/circ1/d32* – имя экземпляра элемента, а значением параметра программирования является строка, в которой логическое выражение задает функцию элемента

$$\#LUT:D=((\sim A3*(A4*A1))+A3*((\sim A2*(A4*A1))+(A2*(A4+A1))))).$$

Секции (слайсы) CLB могут служить источником логических констант, например

```
inst "XDL_DUMMY_CLKB_VCC_X22Y0" "VCC",placed CLKB VCC_X22Y0,
cfg "_NO_USER_LOGIC:: _VCC_SOURCE::VCCOUT ".
```

Подстрока *\_NO\_USER\_LOGIC::* указывает, что эта секция не относится к рассматриваемому устройству и служит источником константы 1 (0), если именем элемента в следующей подстроке конфигурации является *\_VCC\_SOURCE* (*\_GND\_SOURCE*).

#### 4. Методика проверки работоспособности алгоритма распаковки

Алгоритм распаковки работает с неофициальным (недокументированным) описанием, правильность его работы основывается на предположении, что по названиям типов элементов можно правильно установить их функции. Смысл проверки заключается в том, что она позволяет подтвердить (опровергнуть) данное предположение. Очевидно, эта проверка не может быть формальной. Так как алгоритм распаковки работает со структурой в памяти, для оценки его работы требуется преобразование этой структуры в форму для вывода результатов. Моделью обрабатываемого структурного описания во внутреннем представлении служит двудольный граф, форма вывода которого должна допускать ручное и автоматическое сравнение с эталонным представлением.

В качестве эталонного представления можно использовать структурное описание, являющееся результатом синтеза. Инструментом синтеза в САПР FPGA Xilinx является програм-

ма XST (Xilinx Synthesis Technology) [3]. Результатом работы XST является файл с расширением NGC, представляющий структурное описание, предназначенное для использования другими инструментами САПР FPGA Xilinx. Структура формата NGC не опубликована в документации Xilinx, и поэтому непосредственно результат синтеза не может быть использован в качестве эталона. Однако в САПР FPGA Xilinx включена программа NGC2EDIF, предназначенная для перевода формата NGC в EDIF. Ее назначением является представление результата синтеза в форме, воспринимаемой инструментами из других САПР. Программа NGC2EDIF по файлу с расширением NGC строит файл, содержащий структурное описание в формате EDIF и имеющий расширение NDF. Описание в файле NDF не предназначено для использования в процессе проектирования FPGA Xilinx. С учетом этих соображений удобно, чтобы форматом вывода для оценки работоспособности алгоритма тоже являлся EDIF (рис. 9).

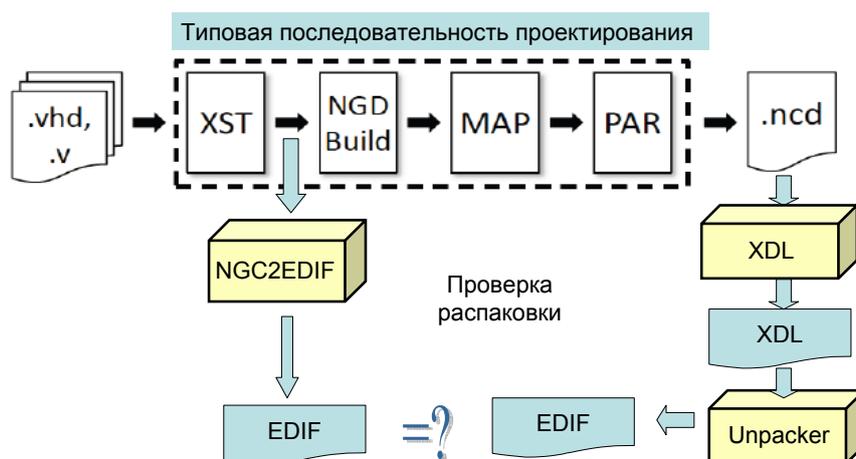


Рис. 9. Тест при оценке работоспособности алгоритма распаковки

Формат EDIF может быть преобразован в формат SF [8], для этого имеются соответствующие программы [9]. Описание в формате EDIF невыполнимо, но может стать выполнимым после преобразования в VHDL и замены элементов их функциональными эквивалентами.

Сравнение текстов в формате EDIF, полученных программой NGC2EDIF и программой распаковки, обнаруживает разницу. Прежде всего сравниваемые представления отличаются по элементам. В EDIF, полученном программой NGC2EDIF (файл NDF), присутствуют буферные элементы IBUF и OBUF. Так как цель распаковки состоит в том, чтобы служить промежуточным результатом при построении функционального описания, то включать в структуру элементы, реализующие только транспорт данных, нецелесообразно. Также отличаются имена типов элементов и имена их выводов. Описание в файле NDF ориентировано на моделирование, основанное на библиотеке UNISIMS [5], и использует имена, определенные в этой библиотеке. Распакованный EDIF содержит имена типов элементов и имена их выводов, заданные в файле отчета. Отличаются и наименования экземпляров элементов и цепей. Таким образом, текстовое сравнение этих двух файлов не позволяет выявить эквивалентность структуры соединений.

Оценка теста работоспособности требует ручного сравнения для выявления эквивалентности описаний. Работа по построению алгоритма преобразования внутреннего представления в формат EDIF, который эквивалентен EDIF, построенный программой NGC2EDIF, нецелесообразна, так как на суть самого алгоритма распаковки не влияет. Кроме того, когда сформировано функциональное описание, заменяющее структурное, тест на работоспособность можно проводить проверкой эквивалентности результатов моделирования.

Методика ручной проверки состоит в установлении соответствия имен типов элементов, имен их выводов, имен экземпляров элементов (instance) и имен цепей. Если эти соответствия представляют собой взаимно-однозначные отображения, тогда структуры, задаваемые в обоих представлениях, эквивалентны.

### Заключение

Проблема распаковки формата XDL рассматривается в [10] в связи с задачей разработки инструментов трассировки и размещения элементов FPGA Xilinx, альтернативных «фирменным». В этой задаче включение альтернативных инструментов в маршрут проектирования Xilinx происходит после выполнения «фирменных» инструментов. Трассированное и размещенное структурное описание распаковывается до уровня логических элементов. Формой представления распакованного описания является снова формат XDL, в такой форме оно обрабатывается (возможно, после ручной корректировки с целью устранения ошибок) альтернативными инструментами. Следствием выбора данного маршрута проектирования является требование «реверсивности» операции распаковки. Результат распаковки должен быть представлен в форме, допускающей его обратное преобразование программой *xdl* в формат NCD. Главной проблемой обеспечения реверсивности является проблема именования типов распакованных элементов, потому что именование, предлагаемое файлом отчета, для этого не годится.

В работе рассмотрена проблема конвертации структурного формата XDL проекта цифрового устройства, реализованного на FPGA семейства Spartan 3 (микросхема XC3S1000), в структурное промежуточное описание. По промежуточному описанию далее будет строиться моделируемое описание на языке VHDL. Описаны алгоритмы, разработанная программа конвертации и методика тестирования результата конвертации.

### Список литературы

1. Суворова, Е.А. Проектирование цифровых систем на VHDL / Е.А. Суворова, Ю.Е. Шейнин. – СПб. : БХВ-Петербург, 2003. – 576 с.
2. Stanford, P. EDIF: Electronic Design Interchange Format Version 2 0 0 / P. Stanford, P. Mancuso. – Electronic Industries Association, ANSI/EIA-548-1988, 1988. – 484 p.
3. Зотов, Ю.В. Проектирование цифровых устройств на основе ПЛИС фирмы XILINX в САПР WebPack ISE / Ю.В. Зотов. – М. : Горячая линия – Телеком, 2003. – 624 с.
4. Beckhoff, C. The Xilinx Design Language (XDL): Tutorial and Use Cases / C. Beckhoff, D. Koch, J. Torresen // Reconfigurable Communication-centric System-on\_Chip (ReCoSoC'2011), 6th International Workshop. – Montpellier, 2011.
5. Xilinx Inc., Spartan-3 Generation FPGA User Guide – Extended Spartan-3A, Spartan-3E, and Spartan-3 FPGA Families UG331 (v1.5). – USA, 2009. – 522 p.
6. Черемисинов, Д.И. Анализ и преобразование структурных описаний СБИС / Д.И. Черемисинов. – Минск : Белорусская наука, 2006. – 275 с.
7. Марков, А.А. Теория алгоритмов / А.А. Марков, Н.М. Нагорный. – М. : Наука. Гл. ред. физ.-мат. лит., 1984. – 376 с.
8. Бибило, П.Н. Кремниевая компиляция заказных СБИС / П.Н. Бибило. – Минск : Ин-т техн. кибернетики АН Беларуси, 1996. – 268 с.
9. Черемисинов, Д.И. Конвертор структурных описаний для интеграции САПР // Междунар. конф. «Теория и практика логического управления» / Москва, 10–11 ноября 2003 г. – М. : Институт проблем управления им. В.А. Трапезникова, 2003. – С. 114–118.
10. Couch, J.D. Applications of TORC: An Open Toolkit for Reconfigurable Computing / J.D. Couch // Virginia Polytechnic Institute and State University [Electronic resource]. – 2011. – Mode of access : [http://scholar.lib.vt.edu/theses/available/etd-08182011-165440/unrestricted/Couch\\_JD\\_T\\_2011.pdf](http://scholar.lib.vt.edu/theses/available/etd-08182011-165440/unrestricted/Couch_JD_T_2011.pdf). – Date of access : 17.04.2012.

Поступила 23.05.12

Объединенный институт проблем  
информатики НАН Беларуси,  
Минск, Сурганова, 6  
e-mail: cher@newman.bas-net.by

**D.I. Cheremisinov**

**GENERATING EXECUTABLE CIRCUIT SPECIFICATIONS  
FROM STRUCTURAL FPGA DESCRIPTIONS**

The problem of transformation of structural description Xilinx FPGA to the description which is possible to simulate without use of libraries of elements is considered. The algorithm of converting XDL format description of FPGA in an intermediate format is described. The model of intermediate representation is a bipartite graph. The algorithmic basis of the conversion operation is the graph substitution which unpacks an element of FPGA into the primitive logic elements. A technique is developed to verify this algorithm. This technique is based on the program that converts the intermediate format into the EDIF format.

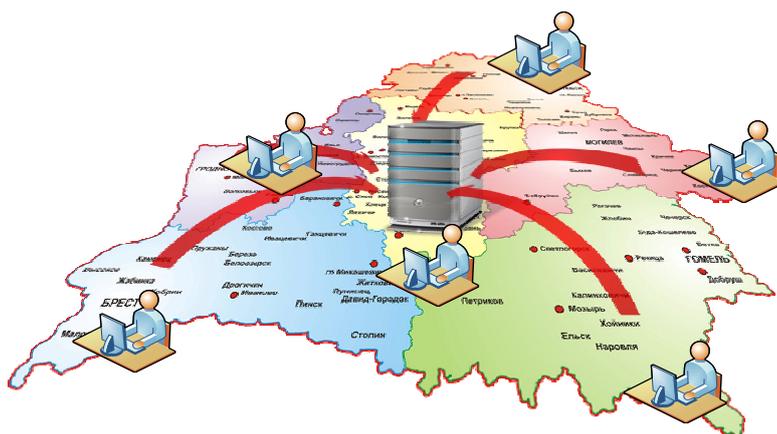
### Лаборатория биоинформатики

#### ***Разработка и развитие интеллектуальных методов обнаружения знаний в данных и создание на их основе информационных технологий поддержки принятия решений в медицине и других областях***

Республиканская информационно-аналитическая система учета и планирования лечебных мероприятий для больных коагулопатиями на основе регистра больных гемофилией (РИАС РГ) обеспечивает:

- ведение регистра пациентов с коагулопатиями;
- учет пациентов с коагулопатиями, нуждающихся в лечебных мероприятиях, которые финансируются за счет средств республиканского бюджета;
- предоставление отчетности о пациентах с коагулопатиями по различным формам;
- расчет необходимых расходов препаратов заместительной терапии для коррекции патологических состояний пациентов с гемофилиями.

РИАС РГ автоматизирует ведение регистра больных коагулопатиями и проведение учета и планирования лечебных мероприятий. Выполняется индивидуализированный расчет потребности в факторах свертывания для пациентов с гемофилией в зависимости от вида коагулопатии, степени тяжести (уровня фактора), наличия ингибитора в соответствии с вероятностью возникновения и частотой различных видов кровотечений, а также проводимого вида помощи с учетом возрастных особенностей. Автоматизированный расчет количества препаратов позволяет повысить достоверность и обоснованность необходимого объема препаратов заместительной терапии для оказания индивидуальной помощи пациентам с гемофилией и в связи с этим рационально расходовать бюджетные средства.



*Общая схема работы специалистов различных уровней оказания медицинской помощи с РИАС РГ*

## ПРАВИЛА ДЛЯ АВТОРОВ

1. Статьи принимаются в редакцию через электронную систему подачи по адресу <http://jinfo.bas-net.by> в формате файлов текстовых редакторов Microsoft Word 97 и Word 2000 для Windows. Основной текст статьи набирается с переносами шрифтом Times New Roman 11 пт, интервал между строками – одинарный, абзацный отступ 1 см, поля по 2,5 см со всех сторон.

2. Объем статьи не должен превышать 12 страниц (включая таблицы, иллюстрации, список литературы), количество иллюстраций – не больше пяти. Допускаются краткие сообщения до трех страниц.

3. Статья должна иметь индекс УДК (универсальная десятичная классификация).

4. Название статьи, фамилии всех авторов и аннотация должны быть переведены на английский язык. Для каждого из авторов приводится развернутое название учреждения с полным почтовым адресом, а также номер телефона и электронный адрес (e-mail) для связи с редакцией.

5. Формулы, иллюстрации, таблицы, встречающиеся в статье, должны быть пронумерованы в соответствии с порядком цитирования в тексте. Ссылки на рисунки и таблицы в тексте обязательны. Необходимо избегать повторения одних и тех же данных в таблицах, графиках и тексте статьи.

Рисунки должны быть представлены в виде файлов формата .cdr, .ai, .wmf, .psd, .jpg, .tif (.tiff) и выполнены с хорошим разрешением в масштабе, позволяющем четко различать надписи и обозначения. Подрисовочные подписи с расшифровкой всех позиций, представленных на рисунке, набираются шрифтом гарнитуры основного текста, размер символов 9 пт. Цветные иллюстрации печатаются только в том случае, когда это необходимо для понимания излагаемого материала.

6. Набор формул выполняется в формульных редакторах Microsoft Equation или Math Type и должен быть единообразным по применению шрифтов и знаков по всей статье.

Прямо (□) набираются: греческие и русские буквы; математические символы ( $\sin$ ,  $\lg$ ,  $\infty$ ); символы химических элементов (C, Cl,  $\text{CHCl}_3$ ); цифры (римские и арабские); векторы; индексы (верхние и нижние), являющиеся сокращениями слов.

Курсивом (~) набираются: латинские буквы – переменные, символы физических величин (в том числе и в индексе).

7. Сокращения в тексте статьи (за исключением единиц измерения) могут быть использованы только после упоминания полного термина. Единицы измерения физических величин следует приводить в Международной системе СИ.

8. Литература приводится автором общим списком в конце статьи. Ссылки на литературу в тексте идут по порядку и обозначаются цифрой в квадратных скобках. Ссылаться на неопубликованные работы не допускается. С примерами оформления библиографического описания в списке литературы можно ознакомиться в приложении 2 к *Инструкции по оформлению диссертации, автореферата и публикаций по теме диссертации* на сайте Высшей аттестационной комиссии Республики Беларусь <http://vak.org.by>.

9. Поступившие в редакцию статьи направляются на рецензирование специалистам. Основным критерием целесообразности публикации является новизна и информативность статьи. Если по рекомендациям рецензента статья возвращается автору на доработку, а переработанная рукопись вновь рассматривается редколлегией, датой поступления считается день получения редакцией ее окончательного варианта. Статьи не по профилю журнала возвращаются авторам после заключения редколлегии.

10. Статьи, направляемые на доработку, должны быть возвращены в исправленном виде с ответами на все вопросы.

11. Редакция журнала предоставляет возможность первоочередного опубликования статей, представленных лицами, которые осуществляют послевузовское обучение (аспирантура, докторантура, соискательство) в год завершения обучения.

12. Авторы несут ответственность за направление в редакцию статей, уже опубликованных ранее, или статей, принятых к публикации другими изданиями.

13. Редакция оставляет за собой право на редакционные изменения, не искажающие основное содержание статьи.

***Журнал «Информатика» включен Высшей аттестационной комиссией Республики Беларусь в список научных изданий для опубликования результатов диссертационных исследований.***

## **Индексы**

**00827**

для индивидуальных  
подписчиков

**008272**

для предприятий и  
организаций