

# ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

УДК 004.032.26

В. И. ПОТАПОВ

Омский государственный  
технический университет

## ВЫЧИСЛЕНИЕ ПОКАЗАТЕЛЕЙ НАДЕЖНОСТИ НЕЙРОННОЙ СИСТЕМЫ С ЧЕЛОВЕКО-МАШИНЫМ КОМПЛЕКСОМ КОНТРОЛЯ И УПРАВЛЕНИЯ ПРОЦЕССОМ НАСТРОЙКИ И ВОССТАНОВЛЕНИЯ ПОСЛЕ ОТКАЗОВ НЕЙРОННОЙ СЕТИ\*

Рассматриваются три математические модели для вычисления показателей надежности нейронной системы с человеко-машинным комплексом контроля и управления процессом настройки и восстановления после отказов нейронной сети при абсолютной и конечной надежности человека-оператора. Приводятся формулы для вычисления вероятности безотказной работы, коэффициента готовности и среднего времени безотказной работы нейронной системы.

**Ключевые слова:** математическая модель, нейронная сеть, нейронная система, надежность, обучение нейронной сети.

\* Данная работа выполнена при финансовой поддержке гранта РФФИ, проект № 11-08-00271.

Практически во всех известных работах, посвященных исследованию надежности искусственных нейронных сетей и нейронных систем, влияние человека на надежность таких систем либо не рассматривалось, либо молчаливо полагалось, что человек, участвующий в процессе настройки (обучении) и вос-

становлении после отказов (переобучении) нейронной сети, является абсолютно надежным [1, 2].

На самом деле любая нейронная (нейрокомпьютерная) система с программно-аппаратным управлением процесса обучения и переобучения после отказов нейронной сети представляет собой человеко-

машинный комплекс, в котором человек может выполнять различные функции: оператора (настройщика), ремонтника, оператора и ремонтника, эргатического резерва (резервного компонента в системе) и другие, в зависимости от поставленных целей и специфики решаемых нейронной системой (НС) задач. Поэтому логично рассматривать две модели надежности нейронной системы: одну — когда человек-оператор абсолютно надежный, а вторую — когда человек-оператор обладает конечной надежностью.

Будем полагать, что человек является составной частью нейронной системы и в той или иной степени оказывает влияние на ее надежность. Такую систему условно назовем «НС-человек». При этом будем считать, что в идеальном случае человек абсолютно надежный, а в реальных условиях человек обладает конечной надежностью и в силу физиологических особенностей или иных обстоятельств может ошибаться (временно отказывать) с интенсивностью  $\nu$  и восстанавливать работоспособность с интенсивностью  $\nu$ , при условии абсолютной надежности человека. При конечной надежности человека  $\nu = 0$ .

Поведение системы «НС-человек» будем рассматривать в предположении марковского процесса и простейших потоках отказов и восстановлений как человека, так и технической части системы, в состав которой включим автоматизированную систему контроля состояния нейронной сети и управления процессом восстановления функциональных свойств сети после отказов. При этом будем полагать, что нейронная сеть избыточна и имеет соответствующий ресурс для конечного числа восстановлений ее работы после отказов, интенсивность отказов системы контроля и восстановления нейронной сети системы равна  $\lambda$ , а интенсивность ее восстановления после отказов равна  $\mu$ . Автоматизированную систему контроля и управления восстановлением после отказов нейронной сети (АСКУ), в состав которой входит аппаратно-программная составляющая и человек, назовем «АСКУ-человек» и в дальнейшем при построении моделей поведения соответствующей НС будем рассматривать показатели надежности именно этой системы, так как она является важнейшей составной частью системы «НС-человек» и в существенной степени определяет ее надежность в целом.

#### Невосстанавливаемая система «АСКУ-человек» при конечной надежности человека

С учетом сделанных выше предположений поведение рассматриваемой системы «АСКУ-человек» может быть описано следующим вероятностным графом состояний.

Обозначение вершин  $E_{ij}$  ( $i = 0, 1; j = 0, 1$ ) графа имеет следующий смысл. Индексом  $i$  обозначено работоспособное ( $i = 1$ ) и неработоспособное (отказовое) ( $i = 0$ ) состояние человека, а индексом  $j$  — работоспособное ( $j = 1$ ) и неработоспособное состояние (состояние отказа) ( $j = 0$ ) аппаратно-программной составляющей рассматриваемой системы «АСКУ-человек».

Очевидно, что система дифференциальных уравнений, соответствующая графу на (рис. 1), имеет следующий вид:

$$\begin{cases} p'_{11}(t) = -\zeta p_{01}(t) - \lambda p_{10}(t), \\ p'_{01}(t) = \zeta p_{01}(t) - \lambda p_{00}(t), \\ p'_{10}(t) = \lambda p_{11}(t) - \zeta p_{00}(t), \\ p'_{00}(t) = \lambda p_{01}(t) + \zeta p_{10}(t), \end{cases} \quad (1)$$

где  $p_{ij}$  — вероятность нахождения системы в состоянии  $E_{ij}$ .

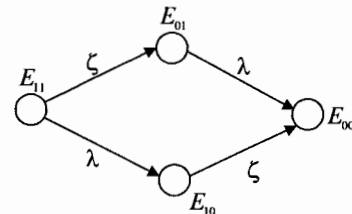


Рис. 1. Граф состояний невосстанавливаемой системы «АСКУ-человек»

Легко видеть, что в невосстанавливаемой системе «АСКУ-человек», модель которой представлена на рис. 1, в зависимости от функций выполняемых человеком (оператор, ремонтник, резервный компонент в системе и т.д.), а также от степени автоматизации процесса контроля и управления настройкой (обучением) нейронной сети и восстановлением заданных функциональных свойств нейронной системы после отказов, возможны следующие три работоспособные состояния, соответствующие вершинам рассматриваемого графа: 1- $E_{11}$ ; 2- $(E_{11}, E_{01})$ ; 3- $(E_{11}, E_{10})$ .

Для каждого из работоспособных состояний, используя известный прием [3] при решении системы уравнений (1), нетрудно получить аналитические выражения для вычисления вероятности безотказной работы  $P_i(t)$  ( $i = 1, \dots, 3$ ) системы в каждом работоспособном состоянии, которые для невосстанавливаемой системы совпадают с нестационарным значением коэффициента готовности в этом состоянии, и выражения для вычисления среднего времени безотказной работы  $T_i$  ( $i = 1, \dots, 3$ ) системы в каждом из работоспособных состояний, определяемого из равенства  $T_i(t) = \int_0^{\infty} P_i(t) dt$ .

В результате решения системы уравнений (1) с помощью преобразования Лапласа имеем:

$$P_1(t) = e^{-(\lambda+\zeta)t}; \quad T_1 = \frac{1}{\lambda+\zeta};$$

$$P_2(t) = e^{-\lambda t}; \quad T_2 = \frac{1}{\lambda};$$

$$P_3(t) = e^{-\zeta t}; \quad T_3 = \frac{1}{\zeta};$$

В связи с тем, что рассматриваемые работоспособные состояния системы несовместны, функция готовности всей системы  $K_r(t)$  может быть определена как сумма вероятностей нахождения системы в каждом из работоспособных состояний [4]

$$K_r(t) = \sum_{i=1}^3 P_i(t).$$

Очевидно, что вероятность безотказной работы всей системы определяется по формуле

$$P(t) = e^{-\lambda t} + e^{-\zeta t} - e^{-(\lambda+\zeta)t},$$

а среднее время безотказной работы системы в целом —

$$\text{по формуле } T = \frac{1}{\lambda} + \frac{1}{\zeta} - \frac{1}{\lambda+\zeta}.$$

Теперь не представляет сложности с помощью компьютера вычислять характеристики надежности рассматриваемой системы для любого интервала времени, если известны значения  $\lambda$  и  $\zeta$ .

#### Восстанавливаемая система «АСКУ-человек» при абсолютной надежности человека

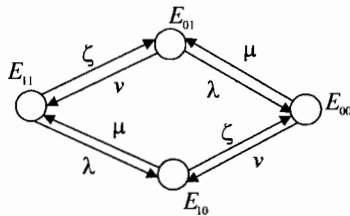


Рис. 2. Граф состояний восстанавливаемой системы «АСКУ-человек» при абсолютной надежности человека

Поведение рассматриваемой системы при сделанных выше предложениях описывается графом несовместных состояний, представленным на рис. 2.

Система дифференциальных уравнений, соответствующая графу на рис. 2, при учете сделанных выше обозначений имеет вид:

$$\begin{aligned} p'_{11}(t) &= \nu p_{01}(t) + \mu p_{10}(t) - \zeta p_{01}(t) - \lambda p_{10}(t), \\ p'_{01}(t) &= \zeta p_{11}(t) + \mu p_{00}(t) - \nu p_{01}(t) - \lambda p_{00}(t), \\ p'_{10}(t) &= \lambda p_{11}(t) + \nu p_{00}(t) - \mu p_{10}(t) - \zeta p_{00}(t), \\ p'_{00}(t) &= \lambda p_{01}(t) + \zeta p_{10}(t) - \mu p_{01}(t) - \nu p_{10}(t). \end{aligned} \quad (3)$$

Проведя аналогичные выше сделанным рассуждения для рассматриваемой восстанавливаемой системы «АСКУ-человек» при абсолютной надежности человека и учитывая, что состояния системы  $E_{ij}$  ( $i, j \in \{0,1\}$ ) несовместны, не трудно получить выражения для вероятностей нахождения системы в каждом работоспособном состоянии  $P_{ij}(t)$  и выражения для коэффициентов готовности системы  $K_{ij}(t)$ , которые в этом случае равны сумме вероятностей нахождения системы в каждом из работоспособных состояний

$$K_{ij}(t) = \sum_{E_{ij} \in \Omega} P_{ij}(t),$$

где  $\Omega$  – подмножество работоспособных состояний множества возможных состояний рассматриваемой системы;  $l$  – номер подмножества работоспособного состояния.

Очевидно, что в рассматриваемой системе работоспособные состояния аналогичны четырем работоспособным состояниям, рассмотренным выше, в невосстанавливаемой системе (рис. 1) и включают еще одно работоспособное состояние.

С учетом сказанного, решая систему уравнений (3) с помощью преобразования Лапласа, имеем:

$$\begin{aligned} K_{\Gamma 1}(t) &= \frac{1}{(\lambda + \mu)(\zeta + \nu)} \left[ \mu\nu + \lambda\nu e^{-(\lambda + \mu)t} + \right. \\ &\quad \left. + \zeta\mu e^{-(\zeta + \nu)t} + \lambda\zeta e^{-(\lambda + \mu + \zeta + \nu)t} \right]; \\ K_{\Gamma 2}(t) &= \frac{\mu}{\lambda + \mu} + \frac{\lambda}{\lambda + \mu} e^{-(\lambda + \mu)t}; \\ K_{\Gamma 3}(t) &= \frac{\nu}{\zeta + \nu} \left[ 1 + \frac{\zeta}{\nu} e^{-(\zeta - \nu)t} \right]; \\ K_{\Gamma 4}(t) &= 1 - \frac{\lambda\zeta}{(\lambda + \mu)(\zeta + \nu)} \left[ 1 - e^{-(\lambda + \mu)t} - e^{-(\zeta + \nu)t} + e^{-(\lambda + \mu + \zeta + \nu)t} \right]. \end{aligned}$$

При этом, среднее время  $T_i$  ( $i = 1, \dots, 4$ ) пребывания рассматриваемой системы в  $l$ -ом подмножестве работоспособных состояний можно получить по приведенной в [5] формуле, учитывающей вероятность

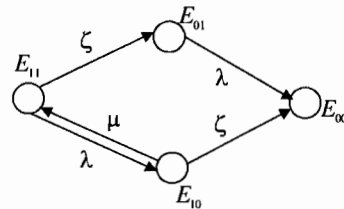


Рис. 3. Граф состояний восстанавливаемой системы «АСКУ-человек» при конечной надежности человека

нахождения системы в  $ij$ -ом работоспособном состоянии, коэффициент готовности и соответствующие интенсивности переходов.

В результате имеем:

$$T_1^* = \frac{1}{\lambda + \zeta}; \quad T_2^* = \frac{1}{\lambda}; \quad T_3^* = \frac{1}{\zeta}; \quad T_4^* = \frac{1}{\lambda + \zeta}.$$

### Восстанавливаемая система «АСКУ-человек» при конечной надежности человека

При конечной надежности человека  $\nu = 0$  и, следовательно, граф состояний такой системы «АСКУ-человек» примет вид, показанный на рис. 3.

Соответствующая графу на рис. 3 система дифференциальных уравнений может быть записана следующим образом

$$\begin{aligned} p'_{11}(t) &= \mu p_{10}(t) - \zeta p_{01}(t) - \lambda p_{10}(t), \\ p'_{01}(t) &= \zeta p_{11}(t) - \lambda p_{00}(t), \\ p'_{10}(t) &= \lambda p_{11}(t) - \mu p_{10}(t) - \zeta p_{00}(t), \\ p'_{00}(t) &= \lambda p_{01}(t) + \zeta p_{10}(t). \end{aligned} \quad (4)$$

Очевидно, что решение системы уравнений (4) не представляет особых трудностей и легко может быть получено с помощью преобразования Лапласа для всех работоспособных состояний системы в следующем виде:

$$\begin{aligned} P_1(t) &= \frac{\mu}{\lambda + \mu} e^{-\alpha_1 t} \left( 1 + \frac{\lambda}{\mu} e^{-(\lambda + \mu)t} \right), \\ P_2(t) &= \alpha e^{-\beta t} + \beta e^{-\gamma t} + \gamma e^{-(\lambda + \mu + \zeta)t}, \\ P_3(t) &= e^{-\zeta t}, \end{aligned}$$

$$P_4(t) = \alpha_2 e^{-\beta t} + \beta_2 e^{-\gamma t} - \gamma_2 e^{-(\lambda + \mu + \zeta)t},$$

где

$$\begin{aligned} \alpha_1 &= \frac{\zeta(\mu + \zeta - \lambda)}{(\mu + \zeta)(\zeta - \lambda)}, \quad \beta_1 = \frac{\mu(\lambda + \mu + \zeta)}{(\lambda + \mu)(\lambda - \zeta)}, \quad \gamma_1 = \frac{\mu(\lambda - \zeta)}{(\lambda + \mu)(\mu + \zeta)}, \\ \alpha_2 &= \frac{\zeta(\mu + \zeta - \lambda)}{(\mu + \zeta)(\zeta - \lambda)}, \quad \beta_2 = \frac{\lambda(\lambda + \mu - \zeta)}{(\lambda + \mu)(\lambda - \zeta)}, \quad \gamma_2 = \frac{\lambda - \zeta}{(\lambda + \mu)(\mu + \zeta)}. \end{aligned}$$

Имея аналитические выражения для  $P_i(t)$  ( $i = 1, \dots, 4$ ), нетрудно получить формулы для среднего времени безотказной работы  $T_i$  ( $i = 1, \dots, 4$ ) рассматриваемой системы [6] для всех работоспособных состояний:

$$T_1 = \frac{\zeta + \mu}{\zeta(\lambda + \zeta + \mu)}, \quad T_2 = \frac{\alpha_1}{\lambda} + \frac{\beta_1}{\zeta} + \frac{\gamma_1}{\lambda + \mu + \zeta},$$

$$T_3 = \frac{1}{\zeta}, \quad T_4 = \frac{\alpha_2}{\lambda} + \frac{\beta_2}{\zeta} - \frac{\gamma_2}{\lambda + \mu + \zeta}.$$

Следует отметить, что при увеличении детализации модели системы «НС-человек» возрастает

сложность уравнений, описывающих эту систему, что усложняет аналитическое решение. В этом случае следует использовать численные методы и моделировать поведение системы на компьютере.

#### Библиографический список

1. Потапов, И. В. Надежность нейрокомпьютерных систем. Модели и задачи / И. В. Потапов. — Омск: Изд-во ОмГТУ, 2007. — 240 с.
2. Потапов, В. И. Математические модели, методы и алгоритмы оптимизации надежности и технической диагностики искусственных нейронных сетей / В. И. Потапов, И. В. Потапов. — Омск: Изд-во ОГУП. Омская областная типография, 2004. — 220 с.
3. Вентцель, Е. С. Исследование операций / Е. С. Вентцель. — М.: Сов. радио, 1972. — 550 с.
4. Методические рекомендации по оценке надежности и эффективности «человек — техника». Научный совет по компле-

ксной проблеме «Кибернетика». АН СССР. — М.: Изд-во ЦНИИТЭИ легкой промышленности, 1971. — 172 с.

5. Контроль функционирования больших систем / Г. П. Шибанов [и др.]. — М.: Машиностроение, 1977. — 330 с.

6. Козлов, Б. А. Справочник по расчету надежности аппаратуры радиоэлектроники и автоматики / Б. А. Козлов, И. А. Ушаков. — М.: Сов. Радио, 1975. — 472 с.

**ПОТАПОВ Виктор Ильич**, доктор технических наук, профессор (Россия), профессор, заведующий кафедрой «Информатика и вычислительная техника». Адрес для переписки: e-mail: ivt@omgtu.ru

Статья поступила в редакцию 24.02.2011 г.  
© В. И. Потапов

УДК 004.052.42

**М. А. БОГАНЕЦ**

Омский государственный  
технический университет

## МЕТОДЫ И АЛГОРИТМЫ ДИАГНОСТИКИ ИСКУССТВЕННОГО НЕЙРОНА С НЕЛИНЕЙНОЙ ФУНКЦИЕЙ АКТИВАЦИИ

В статье представлены методы и алгоритмы диагностики искусственного нейрона с нелинейной функцией активации на наличие в нем отказов. Представлена модель искусственного нейрона, обладающего возможностью диагностики.

**Ключевые слова:** искусственный нейрон, нелинейная функция активации, отказ, методы диагностики, алгоритмы диагностики.

### 1. Введение

Проведенные библиографические исследования показали слабую проработанность темы технической диагностики искусственных нейронов (ИН) с нелинейными функциями активации (НФА), что, учитывая применимость сетей, состоящих из нейронов данного типа, для построения экспертных систем, распознавания образов и работы с нечеткой логикой является существенным недостатком. Кроме того, существующие методы технической диагностики [1] обладают рядом серьезных недостатков. Так, метод построения проверяющих тестов существенно зависит от типа функции активации и не предполагает возможность изменения значений весовых коэффициентов ИН. Схемотехнические методы применимы для узкого круга аппаратных реализаций нейрона. В рамках данной работы сделана попытка исследования принципиальной возможности диагностики ИН с НФА и разработки методов и алгоритмов для ее осуществления.

**Отказом (неисправностью)** будем считать событие, заключающееся в нарушении работоспособного состояния объекта (ИН или ИНС). **Перебегающие отказы** (которые в большинстве случаев являются следствием обратимых случайных изменений режимов работы и параметров системы; при возвращении ре-

жима работы в допустимые пределы системы сама, обычно без вмешательства человека, возвращается в работоспособное состояние.) рассматривать не будем. **Константным отказом** ИН будем считать отказ, заключающийся в возникновении короткого замыкания или обрыва на входах или выходе нейрона. **Параметрическим отказом** будем считать событие, заключающееся в изменении весового коэффициента ИН вследствие возникновения неисправности. **Компенсирующей неисправностью** будем считать неисправность, при которой выходное значение нейрона на заданных входных наборах соответствует выходному значению аналогичного исправного нейрона, что возможно только в избыточных структурах.

### 2. Модель ИН

Рассмотрим модель ИН, представленную на (рис. 1). Нейрон содержит  $n$  входов, на каждый отдельный вход  $x_i$  подается непрерывный сигнал произвольной формы. Каждому входу  $x_i$  соответствует свой весовой коэффициент  $\omega_i$  (множитель входного сигнала), назначение которого аналогично назначению усилителя сигнала в электронных схемах. Совокупность весовых коэффициентов образует вектор весов  $W$ . Значения весовых коэффициентов могут быть как положительными, так и отрицательными. Все входы соединены с сумматором, реализующим функцию

Возможные одиночные неисправности модели ИН с НФА

Тип неисправности	Наименование неисправности	Требования к тестовому набору	OUT	Примечание
1	Обрыв на выходе ИН с НФА	OUT $\neq$ 0, OUT $\neq$ 0	OUT=0	Для обнаружения требуется один набор с $x_i \neq 0$ и $\omega_i \neq 0$
2	Обрыв на i-м входе ИН с НФА	NET=0, OUT=0	OUT $_{\min} \leq$ (OUT $\neq$ 0) $\leq$ OUT $_{\max}$	Проверяются входы ИН с НФА, для которых $x_i \neq 0$
3	Уменьшение весового коэффициента $\omega_i$	NET=0, OUT=0	OUT $_{\min} \leq$ (OUT $\neq$ 0) $\leq$ OUT $_{\max}$	
4	Короткое замыкание на выходе ИН с НФА	OUT=0	OUT>0	Требуется один тестовый набор
5	Короткое замыкание на i-м входе ИН с НФА	NET=0, OUT=0	OUT $_{\min} \leq$ (OUT $\neq$ 0) $\leq$ OUT $_{\max}$	Проверяются входы ИН с НФА, для которых $x_i \neq 1$
6	Увеличение весового коэффициента $\omega_i$	NET=0, OUT=0	OUT $_{\min} \leq$ (OUT $\neq$ 0) $\leq$ OUT $_{\max}$	

суммирования произведения значения входного сигнала на соответствующий ему весовой коэффициент по всем входам:  $NET = \sum x_i \omega_i$ . Блок вычисления, в свою очередь, рассчитывает  $OUT = f(NET)$ , где  $f(NET)$  – функция активации нейрона. Различают следующие нелинейные функции активации:

1. Сигмоидальная функция (функция Ферми):

$$OUT = \frac{1}{1 + e^{-NET}} = F(NET)$$

2. Гиперболический тангенс:

$$OUT = th(NET) = \frac{e^{NET} - e^{-NET}}{e^{NET} + e^{-NET}}$$

3. Экспоненциальная функция:

$$OUT = e^{-NET}$$

4. SOFTMAX-функция:

$$OUT = \frac{e^{NET}}{\sum_i e^{NET_i}}$$

5. Участки синусоиды:

$$OUT = \sin(NET), NET \in (-\pi, \pi)$$

6. Гауссова кривая:

$$OUT = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(NET-m)^2}{2\sigma^2}}$$

### 3. Ограничения модели ИН

Представленная в данной работе модель ИН обладает следующими ограничениями:

1. Вычисления значения OUT нейрона предполагаются мгновенными, не вносящими задержки. Непосредственно моделировать динамические системы, имеющие «внутреннее состояние» (например, системы с памятью), с помощью таких нейронов нельзя.

2. За время тестирования вероятность возникновения одной и более неисправностей пренебрежительно мала. Рассматриваемые в данной работе алгоритмы должны иметь высокое быстродействие.

### 4. Классификация возможных однократных неисправностей ИН

Определим возможные однократные неисправности ИН с НФА ((табл. 1) составлена на основе таблицы неисправностей для ИН с пороговой логикой [2]).

OUT $_{\min}$  будем считать минимальное возможное значение (левую границу диапазона значений OUT), OUT $_{\max}$  по аналогии будем считать максимальное возможное значение OUT (правую границу диапазона значений OUT).

Из анализа (табл. 1) видно, что для обнаружения неисправностей типа 2 – 6 необходимым требованием к диагностическому набору является NET=0, OUT=0. Для обнаружения неисправности типа 1 необходимо и достаточно выполнение условия NET $\neq$ 0, OUT=0, то есть при ненулевом векторе входных значений **X** и весов **W** значение выхода OUT=0 явно указывает на отказ типа короткого замыкания на выходе нейрона. Следует отметить, что неисправности типа 3, 6 описываемыми в данной работе методами и алгоритмами не локализируются. Возможно лишь установление факта наличия отказа в нейроне (без определения его типа), чего вполне достаточно для диагностики ИНС в целом.

Необходимое и достаточное условия диагностируемости ИН с НФА приведены в [3].

### 5. Методы и алгоритмы диагностики ИН с НФА

Для решения задачи диагностики нейронов с нелинейными функциями активации на наличие константных и параметрических неисправностей на его входах и выходе модель формального нейрона (описанная, например, в [4, 5]) была изменена. Был добавлен диагностический вход  $S_0$  с соответствующим весовым коэффициентом  $\omega_s$  (фактически, для диагностики используется один из информационных

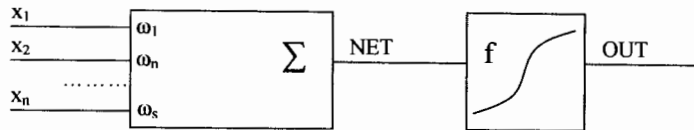


Рис. 1. Модель формального нейрона

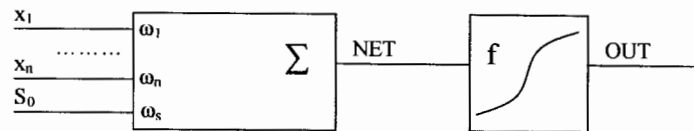


Рис. 2. Предлагаемая модель нейрона с нелинейной функцией активации с возможностью диагностики неисправностей

входов нейрона). Модель такого нейрона представлена на рис. 2.

Работу рассматриваемого ИН можно описать следующими выражениями:

$$OUT = F(NET) - D_{dg}, \quad (1)$$

$$OUT = F(\sum x_i \omega_i - S_0 \omega_s) - D_{dg} \quad | \quad i = 1, \dots, n \quad (2)$$

где  $x_1, \dots, x_n$  – информационные входы, на которые подается непрерывный сигнал произвольной формы;

$S_0$  – диагностический вход ИН, на который подается константа из диапазона  $[0; \max(x_i)] \quad | \quad i = 1, \dots, n$ , причем при работе нейрона в обычном режиме (не в диагностическом)  $S_0 = 0$ ;

$\omega_1, \dots, \omega_n$  – весовые коэффициенты соответствующих им входов  $x_1, \dots, x_n$ ;

$\omega_s$  – весовой коэффициент диагностического входа  $S_0$ . При диагностике ИН на наличие отказов  $\omega_s = -\sum \omega_i \quad | \quad i = 1, \dots, n$ .

NET, OUT – то же, что и в случае (рис. 1)

$D_{dg}$  – диагностическая константа, обеспечивающая сходимость функции OUT в точке (0,0), вычисляется по формуле (1) при условии  $OUT = 0$ , то есть  $D_{dg} = F(NET = 0)$ . Так, для сигмоидальной функции активации  $D_{dg} = 0,5$ .

Для диагностики нейрона на наличие неисправностей необходимо рассчитать значение  $\omega_s$  по формуле, приведенной выше. Тогда:

при  $S_0 = x_i \neq 0 \quad | \quad i = 1, \dots, n$ :

$$OUT = F(0) - D_{dg} = 0 \quad (3)$$

при  $S_0 = 0$ :

$$OUT = F(\sum x_i \omega_i) - D_{dg} \quad (4)$$

Для проверки работоспособности нейрона необходимо на вход нейрона подать тестовые наборы, представленные в табл. 2.

Константу, подаваемую на все входы нейрона при диагностике, на основании опытных данных (апробации приведенных методов и алгоритмов на различных моделях ИН) предлагается принимать равной

$$D_{dg} = \min(\max x_i) \quad | \quad i = 1, \dots, n \quad (5)$$

Рассмотрим процесс диагностики ИН с НФА на наличие константных неисправностей на входах или выходе нейрона.

При подозрении наличия отказа типа логической константы для проверки корректности работы нейрона принимают  $\omega_s = -\sum \omega_i \quad | \quad i = 1, \dots, n$  и подают

последовательно диагностические наборы, представленные в табл. 2.

Возможны следующие случаи:

$$x_1 = \dots = x_n = (const \neq 0); S_0 = 0; OUT = 0 \quad OUT = (const = 0) \quad (6)$$

$$\begin{cases} x_1 = \dots = x_n = S_0 = 0; & OUT \neq 0 \\ x_1 = \dots = x_n = S_0 = (const \neq 0); & OUT = 0 \quad x_i = (const \neq 0) \end{cases} \quad (7)$$

$$\begin{cases} x_1 = \dots = x_n = S_0 = 0; & OUT = 0 \\ x_1 = \dots = x_n = S_0 = (const \neq 0); & OUT \neq 0 \quad x_i = (const = 0) \end{cases} \quad (8)$$

$$\begin{cases} x_1 = \dots = x_n = S_0 = 0; & OUT \neq 0 \\ x_1 = \dots = x_n = S_0 = (const \neq 0); & OUT \neq 0 \quad OUT = (const \neq 0) \end{cases}$$

В случае (6) отказ носит характер обрыва на выходе ИН; в (7) – короткого замыкания на входе нейрона; в (8) – обрыва на входе; в (9) – короткого замыкания на выходе ИН.

В случае, если обнаруженная неисправность не является константной неисправностью на выходе нейрона, возможна дальнейшая локализация отказа с точностью до входа (случаи (7) и (8), если неисправность однократная). Для этого следует применить следующие алгоритмы:

1) Алгоритм 1 (для случая (7)):

1.  $i = 0, lconst = (const \neq 0)$ ;

2.  $\omega_s = 0$ ;

3.  $i = i + 1$ ;

4.  $\omega_s = \omega_s - \omega_i$ ;

5.  $S_0 = lconst; x_i = lconst \quad | \quad j = 1, \dots, i; x_k = 0 \quad | \quad k = i + 1, \dots, n$ ;

6. если  $OUT \neq 0$ , идти к п.3;

7.  $x_i = x_{i, \max} (const \neq 0)$  и есть искомый вход.

2) Алгоритм 2 (для случая (8)):

1.  $i = n + 1, lconst = (const \neq 0)$ ;

2.  $\omega_s = -\sum \omega_i \quad | \quad i = 1, \dots, n$ ;

3.  $i = i - 1$ ;

4.  $\omega_s = \omega_s + \omega_i$ ;

5.  $S_0 = lconst; x_j = 0 \quad | \quad j = i, \dots, n; x_k = lconst \quad | \quad k = 1, \dots, i - 1$ ;

6. если  $OUT \neq 0$ , идти к п.3;

7.  $x_i = const \neq "0"$  и есть искомый вход.

Алгоритм (1) назовем алгоритмом последовательного включения, а (2) по аналогии алгоритмом последовательного выключения.

Среднее количество тестовых наборов, необходимых для выявления неисправностей рассматриваемых типов, приведено в табл. 3.

Следует отметить, что отличия предлагаемой модели нейрона с нелинейной функцией активации с возможностью диагностики неисправностей типа логических констант на его входах/выходе от модели формального нейрона, описанной в [4, 5], не носят характер критических и не ведут к существенному уве-

Таблица 2

Диагностические наборы, подаваемые на вход нейрона при подозрении наличия неисправности типа логической константы на входах/выходе нейрона

№ набора	Диагностические наборы	NET	OUT
1	$x_1 = \dots = x_n = (\text{const} \neq 0); S_0 = 0$	$\neq 0$	$\neq 0$
2	$x_1 = x_2 = \dots = x_n = S_0 = 0$	0	0
3	$x_1 = x_2 = \dots = x_n = S_0 = (\text{const} \neq 0)$	0	0

Таблица 3

Среднее количество тестовых наборов, необходимых для локализации однократной константной неисправности с точностью до входа/выхода

Наименование неисправности	Среднее количество тестовых наборов для локализации неисправности
Обрыв на выходе ИН с НФА	1
Короткое замыкание на выходе ИН с НФА	3
Короткое замыкание на выходе ИН с НФА	$n/2 + 3$
Короткое замыкание на $i$ -м входе ИН с НФА	$n/2 + 3$

личению сложности. Фактически, в качестве диагностического в данном случае используется один из его информационных входов.

Рассмотрим результаты применения приведенных выше методов и алгоритмов в случае возникновения параметрического отказа.

1. Пусть  $\omega_{i0}$  – весовой коэффициент входа  $i$ , для которого имеет место отказ типа увеличения весового коэффициента (случай б (табл. 1)), то есть  $\omega_{i0} > \omega_i$ , или  $\omega_{i0} = \omega_i + (\text{const} \neq 0)$ , где  $\omega_i$  – весовой коэффициент входа  $i$  исправного нейрона. Тогда при подаче тестовых наборов 1 – 3 (табл. 2) будем иметь следующее:

- $x_1 = \dots = x_n = (\text{const} \neq 0); S_0 = 0$  NET  $\neq 0$  OUT  $\neq 0$
- $x_1 = x_2 = \dots = x_n = S_0 = 0$  NET = 0 OUT = 0
- $x_1 = x_2 = \dots = x_n = S_0 = (\text{const} \neq 0)$  NET  $\neq 0$  OUT  $\neq 0$

2. Аналогично приведенному выше, примем  $\omega_{i0} < \omega_i$ , или  $\omega_{i0} = \omega_i + (\text{const} \neq 0)$ .

- $x_1 = \dots = x_n = (\text{const} \neq 0); S_0 = 0$  NET  $\neq 0$  OUT  $\neq 0$
- $x_1 = x_2 = \dots = x_n = S_0 = 0$  NET = 0 OUT = 0
- $x_1 = x_2 = \dots = x_n = S_0 = (\text{const} \neq 0)$  NET  $\neq 0$  OUT  $\neq 0$

Как видно из представленного выше, параметрический отказ выявляется на третьем тестовом наборе из табл. 2. Дальнейшая его локализация алгоритмами последовательного включения и выключения невозможна. Другими словами, если внутренняя структура ИН не является абсолютно надежной (то есть в нем возможно возникновение параметрического отказа), тестовыми наборами устанавливается сам факт неисправности, но не ее тип или место возникновения в нейроне. На практике для ИНС с НФА, поддерживающих аппаратное или программное резервирование, важен сам факт обнаружения неисправности для дальнейшей замены отказавшего нейрона резервным, а не ее тип.

В результате апробации представленных в работе методов и алгоритмов (с помощью программного моделирования ИН и внесения случайных неисправностей) на моделях ИН с двукратными неисправностями было установлено, что приведенные выше тестовые наборы обнаруживают наличие кратных некомпен-

сирующих неисправностей любого типа. На основании этого можно сделать вывод о применимости методов диагностики для выявления любых кратных неисправностей рассматриваемых типов. Локализация неисправности с точностью до входа, выхода или весового коэффициента при этом невозможна.

#### Выводы

В заключение следует отметить, что приведенные методы и алгоритмы диагностики, обладая некоторыми достоинствами:

- высоким быстродействием методов диагностики ИН на наличие константных отказов на его входах или выходе;
- возможностью диагностики кратных неисправностей нейрона;
- использованием имеющихся входов нейрона в качестве диагностических не лишены недостатков, среди которых можно выделить следующие:
  - невозможность локализации отказа нейрона с точностью до его входа, выхода или весового коэффициента для нейронов, не обладающих абсолютно надежной внутренней структурой;
  - низкое быстродействие алгоритмов последовательного включения и выключения, применяемых для локализации константных неисправностей входов и выхода нейрона с абсолютно надежной внутренней структурой.

В дальнейшем планируется проведение исследований, направленных на модификацию алгоритмов для обеспечения возможности локализации параметрических отказов нейрона, а также на вычисление вероятности возникновения в нем компенсирующих неисправностей на основе результатов апробации представленных методов и алгоритмов на различных моделях ИН.

#### Библиографический список

- Потапов, И. В. Математические модели и методы оптимизации функциональной надежности искусственных нейронных



сетей : дис. канд. техн. наук : 05.13.01 / И. В. Потапов. — Томск, 2004. — 175 с.

2. Пальянов, И. А. Схемотехника и контроль элементов пороговой логики / И. А. Пальянов, В. И. Потопов — Омск : Изд. ОмГТУ, 1993. — 155 с.

3. Боганец, М. А. Методы и алгоритмы диагностики искусственных нейронных сетей с нелинейными функциями активации / М. А. Боганец // Омский научный вестник. — 2010. — № 3(93). — С. 223 — 227.

4. Мкртчян, С. О. Нейроны и нейронные сети / С. О. Мкртчян. — М. : Энергия, 1971. — 232 с.

5. Галушкин, А. И. Теория нейронных сетей : учеб. пособие для вузов. В 3 кн. Кн. 1 / А. И. Галушкин. — М. : ИПРЖР, 2000. — 416 с.

**БОГАНЕЦ Максим Александрович**, ассистент кафедры «Информатика и вычислительная техника». Адрес для переписки: e-mail: max.boganets@gmail.com

Статья поступила в редакцию 04.02.2011 г.

© М. А. Боганец

УДК 004.9 : 61(06)

**А. В. МЫЦИК**

Омская государственная  
медицинская академия

## ИСПОЛЬЗОВАНИЕ ПРОГРАММЫ IMAGEJ ДЛЯ АВТОМАТИЧЕСКОЙ МОРФОМЕТРИИ В ГИСТОЛОГИЧЕСКИХ ИССЛЕДОВАНИЯХ

**В статье идёт речь о практическом использовании программы ImageJ для автоматизации гистологических исследований и показана целесообразность внедрения данной программы в работу морфологических лабораторий.**

**Ключевые слова:** гистология, автоматическая морфометрия, ImageJ.

С проблемой анализа гистологического препарата сталкивается любой исследователь. Существует множество методик измерения размеров клеток, внутриклеточных структур, тканевых образований, которые появились практически одновременно с массовым распространением световой микроскопии. Однако они основаны на непосредственном измерении каждого объекта исследователем, поэтому являются трудоемкими, требующими большого внимания и усидчивости. Результаты такой работы в значительной степени зависят от субъективного взгляда и вследствие этого могут содержать значительное количество ошибок, различаться у разных исследователей, имеют неидеальную повторяемость.

В связи с вышеизложенным различные фирмы — производители оборудования для микроскопии (Carl Zeiss, Leica) разработали программные комплексы, позволяющие автоматизировать процесс морфометрии. Эти программы могут работать как с изображением, получаемым с цифровой камеры в режиме реального времени, так и с электронными фотографиями. Программы достаточно дороги, порой их цена сопоставима со стоимостью хорошего микроскопа, хранят и обрабатывают данные в своих собственных форматах, поэтому широкого распространения не получили.

В конце 90-х годов прошлого века, в связи с распространением аппаратуры для оцифровки изображений, переход значительного количества медицинских организаций на хранение документации, в том числе электронных фотографий микропрепаратов, рентгенограмм в электронном виде Департамент здравоохранения США, точнее его агентство National Institutes of Health (NIH), выступил инициатором разработки программы, которая могла бы проводить автоматический анализ визуальных данных и являлась общественным достоянием, то есть была бы для всех бесплатна.

Такая программа с открытой архитектурой была разработана в 1997 году Вэйном Расбандом (Wayne Rasband) в исследовательском отделении Национального института психиатрии в Бетесде, Мэриленд и получила название ImageJ [1]. Ее модернизация силами пользователей продолжается по сей день. Некоторые модификации выделились в самостоятельный продукт, например Fiji, и развиваются параллельно. Программу, плагины, макросы и полную документацию можно получить на сайте [rsbweb.nih.gov](http://rsbweb.nih.gov).

Плагины, которых в настоящее время более 500, написанные пользователями, делают возможным решение многих задач анализа и обработки изображений, позволяют проводить трехмерную визуализацию в диапазоне от внутриклеточных структур до рентгенологических изображений [2], автоматические сравнения [3], вплоть до создания автоматизированных систем изучения, обрабатывающие пакеты изображений практически без участия человека [4, 5].

ImageJ позволяет отображать, редактировать, анализировать, обрабатывать, сохранять и печатать 8-, 16- и 32-битные изображения. Программа может читать многие форматы изображений, включающие TIFF, PNG, GIF, JPEG, BMP, DICOM, FITS, а также форматы программ AxioVision (Carl Zeiss), Leica и многих других коммерческих продуктов. ImageJ поддерживает стеки — серии изображений, которые объединены в одном окне, а многопоточные трудоемкие операции могут выполняться на многопроцессорных системах в параллельном режиме. В ImageJ можно вычислять площади, статистические показатели пиксельных значений различных выделенных областей интереса на изображениях, которые выделены вручную или при помощи пороговых функций. Программа может измерять расстояния и углы, создавать гистограммы плотности и рисовать профили линий.

Поддержка сглаживания, резкости, обнаружение края, медианная фильтрация, функция порогов, наст-



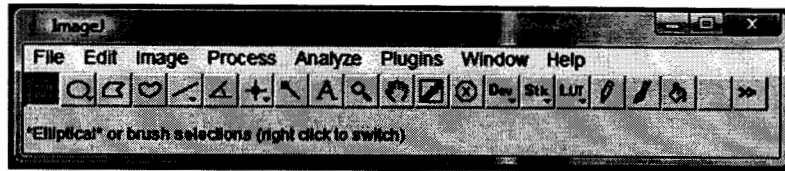


Рис. 1. Основное окно программы

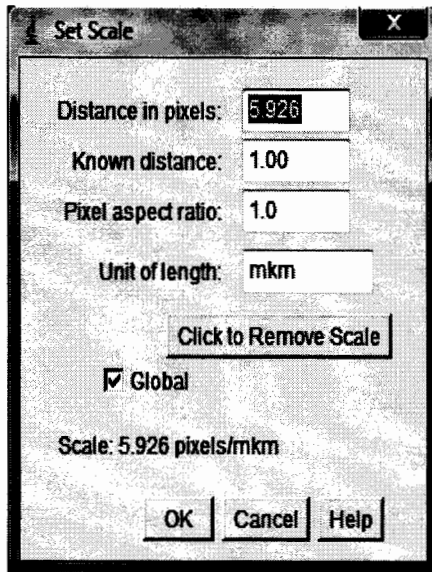


Рис. 2. Установка отношения пиксел/мкм

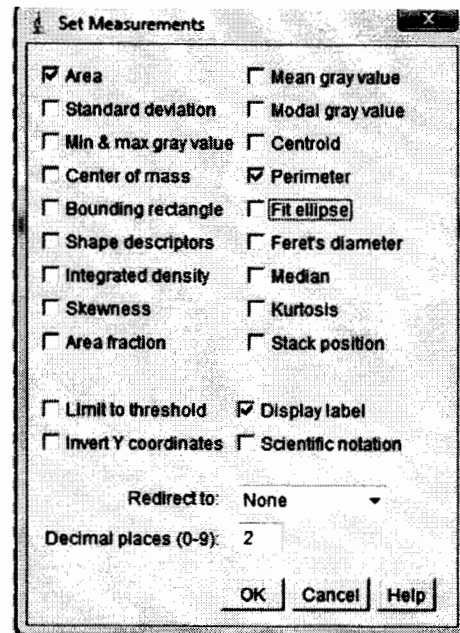


Рис. 3. Установка измеряемых показателей

ройка яркости и контрастности, кадрирование, поворот изображения помогут отредактировать и сохранить изображение для публикации.

Поскольку программа разработана на Java, она является независимой от операционной системы и может работать на следующих платформах: Microsoft Windows, Mac OS, Linux и Sharp Zaurus PDA. Кроме того, разработаны драйверы, позволяющие производить захват изображения практически со всех известных цифровых камер.

На сегодняшний день ImageJ, пожалуй, самая быстрая программа обработки изображений: она обрабатывает около 40 миллионов пикселей в секунду.

Рассмотрим работу с ImageJ на конкретном примере из проведенного нами морфометрического исследования коры головного мозга человека.

Основное окно программы (рис. 1) имеет простой, интуитивно понятный интерфейс. Открываем изображение (фотография среза коры головного мозга человека, окрашенного DAPI, 4',6-diamidino-2-phenylindole) в меню «Анализ» выбираем пункт Set Scale (рис. 2), устанавливаем соотношение пиксел/мкм для получения всех данных в микронах. При необходимости можно поместить шкалу на изображение. Затем в том же меню выбираем пункт Set Measurements (рис. 3), где доступны следующие показатели: площадь, среднее значение серого, стандартное отклонение, модальное значение серого, минимальное и максимальное значение серого, центр масс, прямоугольник (в который впишется частица), эллипс (в который впишется частица), форма дескрипторов, диаметр Ферета — максимальное расстояние между краями частицы, интегральная плотность, медиана, асимметрия, куртозис, фракционность площади, позиция стека, пороговые значения, имя файла, зеркало, научная индексация, результаты в файл, количество знаков после запятой.

Программа очень пластична, имеет множество настроек. Справку по ним, как и по самой программе, можно найти на сайте [imagejdocu.tudor.lu](http://imagejdocu.tudor.lu).

Посчитаем площадь и периметр. Накладываем на изображение маску по наиболее подходящему из 17 алгоритмов, по которой будут производиться расчеты (рис. 4) с помощью инструмента Threshold (пороги), и переходим непосредственно к анализу.

Пункт меню «Анализ частиц», Analyze Particles, (рис. 5) позволяет достаточно полно описать частицы, в нашем случае ядра клеток, которые будут учитываться при подсчете.

Сначала идет сегрегация по размеру площади, тем самым исключив подсчет, например, артефактов. Затем выбираем меру округлости объектов, можно производить подсчет всех частиц, указав широкий интервал или частиц, имеющих крайние признаки. Эта функция позволяет подсчитать на фотографии препарата, например, только пирамидные нейроны или только овальные глиальные клетки. Также можно выбрать, считать или нет частицы, оказавшиеся на краях препарата, учитывать пустоты внутри периметра и другие параметры.

Нажимаем ОК, полученный результат накладываем оверлеем на первичный снимок и проверяем корректность подсчета (рис. 6). Полученные данные вместе с именем файла заносятся в таблицу, доступную из Microsoft Excel (рис. 7). При корректном подсчете можно указать целевую папку с фотографиями препаратов, и они будут обработаны пакетно, данные занесены в таблицу, а работа, которая заняла бы недели, будет выполнена за пару минут.

Таким образом, на основании конкретного примера из нашего исследования показано, что ImageJ

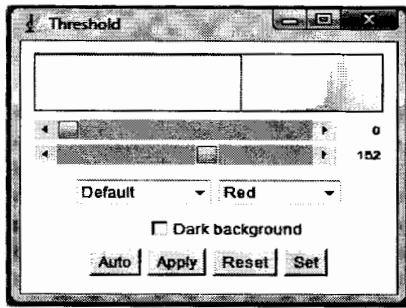


Рис. 4. Установка пороговых значений изображения

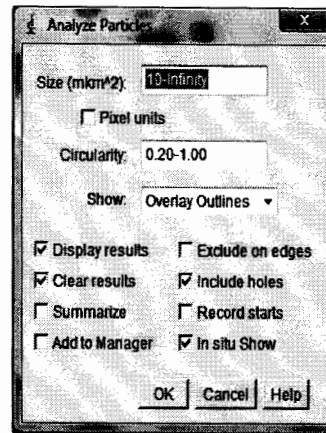


Рис. 5. Определение частиц для анализа

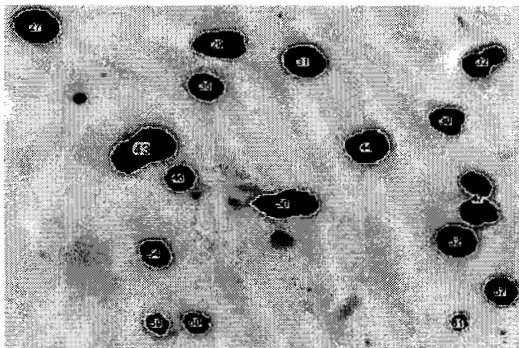


Рис. 6. Выделенные частицы на микрофотографии

File	Euler	Form	Results
1	11_3_c2-2.JPG	19.13	18.59
2	11_3_c2-2.JPG	21.84	20.15
3	11_3_c2-2.JPG	20.02	19.16
4	11_3_c2-2.JPG	107.34	71.98
5	11_3_c2-2.JPG	52.70	35.41
6	11_3_c2-2.JPG	12.95	14.51
7	11_3_c2-2.JPG	25.31	33.10
8	11_3_c2-2.JPG	45.93	26.75
9	11_3_c2-2.JPG	20.24	17.17
10	11_3_c2-2.JPG	46.12	27.35
11	11_3_c2-2.JPG	40.43	24.94
12	11_3_c2-2.JPG	15.56	16.06
13	11_3_c2-2.JPG	18.22	17.17
14	11_3_c2-2.JPG	30.35	23.57
15	11_3_c2-2.JPG	31.89	24.68
16	11_3_c2-2.JPG	15.92	16.90
17	11_3_c2-2.JPG	43.68	26.87
18	11_3_c2-2.JPG	45.95	31.84
19	11_3_c2-2.JPG	28.22	21.65

Рис. 7. Таблица с результатами измерений

является совершенным инструментом обработки массивов однотипных изображений при проведении морфометрических исследований головного мозга. Считаю целесообразным внедрение данной программы в работу морфологических лабораторий и учебный процесс на соответствующих кафедрах в качестве альтернативы рутинным методам обработки изображений.

#### Библиографический список

1. Collins, T. J. ImageJ for microscopy / T. J. Collins // BioTechniques 43 (1 Suppl). – July 2007 – P. 25–30.
2. Creation of DICOM – aware applications using ImageJ / D. Barboriak [et al.]. // J Digit Imaging 18 (2) – 2005. – P. 91–9.
3. AFM/CLSM data visualization and comparison using an open-source toolkit / B. Rajwa [et al.]. // Microsc Res Tech 64 (2). – 2004. – P. 176–84.

4. E. Gering, C. Atkinson A rapid method for counting nucleated erythrocytes on stained blood smears by digital image analysis // J Parasitol 90 (4). – P. 879–81.

5. Dougherty, G. Digital Image Processing for Medical Applications // Cambridge University Press. – 2009.

**МЫЦИК Алексей Владимирович**, ассистент кафедры гистологии, цитологии и эмбриологии.  
Адрес для переписки: e-mail: elysei@mail.ru

Статья поступила в редакцию 01.02.2011 г.  
© А. В. Мыцик

## Книжная полка

004.4/Ч-49

**Чернов, Ю. П. Программирование в системе Pascal ABC [Текст] : учеб. пособие / Ю. П. Чернов, О. П. Шафеева ; ОмГТУ. – Омск, 2010. – 100 с. – ISBN 978-5-8149-0897-1.**

Пособие включает теоретический материал с описанием элементов языка и системы программирования Pascal ABC, а также примеры разработки алгоритмов и программ на языке Паскаль для задач разного уровня сложности. Рассмотренные схемы проектируемых алгоритмов позволяют облегчить усвоение материала студентами.

004.4/М61

**Минитаева А. М. Разработка и стандартизация программных средств и информационных технологий : учеб. пособие / А. В. Минитаева ; ОмГТУ. – Омск : Изд-во ОмГТУ, 2011. – 90 с. – ISBN 978-5-8149-1063-9.**

Изложены основные понятия и общие положения о стандартизации, жизненном цикле разработки программных средств, его процессах. Подробно рассмотрены разновидности моделей жизненного цикла разработки программного продукта.

## ФОРМИРОВАНИЕ ПРЕДСТАВЛЕНИЙ ДАННЫХ ДЛЯ ПОСТРОЕНИЯ МЕДИЦИНСКИХ ДИАГНОСТИЧЕСКИХ ШКАЛ\*

В работе рассматривается проблема автоматизации построения представления данных со списочными компонентами из исходного реляционного представления данных. В основе построения лежит формальное определение промежуточной и целевой моделей данных. Полученные представления используются в анализе данных при построении медицинских оценочных шкал.

**Ключевые слова:** многомерное представление данных, анализ данных.

\* Работа выполнена по гранту РФФИ № 09-07-00059-а

### 1. Введение

Построение гиперкубического (многомерного) представления данных является основой для технологии оперативной аналитической обработки данных OLAP (online analytical processing) [1, 2] и используется в алгоритмах интеллектуального анализа данных (Data Mining) [3, 4].

В данной работе рассматривается модель данных, названная «композиционной таблицей», которая позволяет многомерное представление данных разместить на плоскости, что делает удобным проведение предварительного визуального анализа.

Рассматриваемые в статьях [2, 5] многомерные модели данных обладают достаточными свойствами для реализации различных классов приложений. Обычно условием, обеспечивающим соответствие значений меры в ячейке элементам измерений, является функциональная зависимость значений мер от атрибутов терминального уровня каждого измерения [5]. В отличие от многомерных моделей данных, рассматриваемых в публикациях, для композиционной таблицы данное условие отсутствует, что позволяет размещать списки значений мер в ячейках. Это дает возможность реализации разнообразных методов анализа данных без дополнительных затрат на сбор и преобразование данных. Другим важным отличием является то, что представление композиционной таблицы является объединением нескольких гиперкубов с одним общим измерением. Тем самым предоставляется возможность работы с различными гиперкубическими представлениями в рамках одного приложения. Модель данных композиционная таблица является обобщением гиперкуба семантическая трансформация [6] на случай нескольких значений в одной ячейке и множества независимых измерений.

### 2. Описание предлагаемого подхода

Для автоматизации построения композиционной таблицы предлагается следующая последовательность формирования ее представления:

1. Пользователь из списка атрибутов БД формирует множество атрибутов: измерения  $X, Y_1, Y_2, \dots, Y_N$  и соответствующие им меры  $Z_1, Z_2, \dots, Z_N$ . Мера  $Z_1$  соответствует измерениям  $(X, Y_1)$ , мера  $Z_2$  соответствует измерениям  $(X, Y_2)$  и т.д. Естественными являются ограничения:  $X \cap Y_i = \emptyset$ ,  $(X \cup Y_i) \cap Z_i = \emptyset$ ,  $i=1, 2, \dots, N$ . Дополнительным технологическим ограничением является запрет на использование атрибута в качестве меры, если он принадлежит какому-либо измерению и на него установлено ограничение в логическом выражении. Причины и детали такого ограничения будут ясны после рассмотрения способов формирования всех необходимых представлений данных.

2. Формирование иерархий измерений для множества атрибутов  $X, Y_1, Y_2, \dots, Y_N$ . Иерархии формируются автоматически по правилам, рассмотренным далее, и пользователю предлагается только их модифицировать.

3. По специальному правилу задаются логические ограничения на измерения  $F_0(X), F_1(Y_1), F_2(Y_2), \dots, F_N(Y_N)$ . По умолчанию каждая формула есть конъюнкция условий определенности (*IS NOT NULL*) для атрибутов измерения.

4. Формирование контекстов измерений  $C_0, C_1, \dots, C_N$  (некоторые контексты могут быть пустыми, а некоторые – псевдоконтекстами). Далее будут представлены соответствующие определения и алгоритмы формирования контекстов.

5. Формирование контекста приложения  $C_{full}$  и соответствующей реализации таблицы связанных соединений *s* со схемой *S* и логическим ограничением  $F(C) = F_0(X) \wedge (F_1(Y_1) \vee F_2(Y_2) \vee \dots \vee F_N(Y_N))$

6. Формирование реализаций измерений  $X, Y_1, Y_2, \dots, Y_N$  с сортировкой значений в соответствии с иерархией, если контекст измерения не пуст, то он используется для формирования, в противном случае реализация измерения является проекцией *s*.

7. Формирование реализации (представления) композиционной таблицы (заполнение значений мер на соответствующих местах таблицы).

Пользователь вручную выполняет шаги 1 и 3, и осуществляет выбор предложенных вариантов в шагах 2, 4 и 5. Все остальные операции выполняются автоматически.

Заметим, что в предложенной последовательности шагов формирования композиционной таблицы отсутствует необходимость каким-либо образом модифицировать исходную операционную БД, что делает возможным реализовать все принципы проектирования БД [7], в том числе самый важный — принцип независимости данных.

### 3. Формирование композиционной таблицы

Для представления данных «композиционная таблица» множества атрибутов  $X$  и  $Y_j$  ( $j=1, 2, \dots, N$ ) являются обобщенными координатами и могут рассматриваться как измерения. Иерархии атрибутов в  $X$  и  $Y_j$  ( $j=1, 2, \dots, N$ ) определяют порядок расположения значений атрибутов в заголовках строк и столбцов пользовательского представления в виде двумерной таблицы.

**Определение 3.1.** Схема иерархии — это связный ориентированный ациклический граф  $H=(A, E)$ , где  $A$  — множество атрибутов,  $E$  — множество дуг.

**Определение 3.2.** Пусть  $V, D$  — атрибуты.  $H$  — схема иерархии, тогда  $V < D$ , если в  $H$  существует путь из вершины  $V$  в  $D$ .

Определим способы задания частичного порядка на множестве атрибутов.

Для задания частичного порядка на множестве атрибутов, входящих в функциональные и многозначные зависимости, используем следующее эвристическое правило.

**Правило 3.1.** Атрибуты из множества атрибутов, принимающего меньшее количество значений, располагаются в иерархии выше, чем атрибуты из множества, принимающего большее количество значений.

Смысл этого правила поясним на примере. Если какие-либо узлы одного уровня в иерархии имеют потомков с совпадающими значениями, то эти потомки в представлении будут дублироваться вместе с собственными потомками для каждого из узлов. Следовательно, чем больше значений будут иметь исходные узлы, тем больше у них будет дублированных потомков. Перемещение таких узлов вниз по иерархии позволит избежать данной ситуации.

Для функциональной зависимости  $V \rightarrow D$ , где  $V$  и  $D$  — множества атрибутов, атрибуты из  $D$  располагаются в иерархии выше, чем атрибуты из  $V$ , так как различные значения множества атрибутов  $V$  могут определять одинаковое значение  $D$ . Таким образом, будем полагать, что для атрибутов  $V_k \in V, D_i \in D \forall k, l, V_k < D_l$ .

Для многозначной зависимости  $V \twoheadrightarrow D (E)$ , где  $V, D, E$  — множества атрибутов, атрибуты из  $V$  располагаются в иерархии выше, чем атрибуты из  $D \cup E$ , так как по определению многозначной зависимости при существовании двух кортежей, совпадающих по  $V$ , существуют еще два кортежа с тем же значением  $V$ . Таким образом, будем полагать, что для атрибутов  $V_k \in V, I_l \in D \cup E \forall k, l, I_l < V_k$ .

В работе [8] предложен алгоритм, автоматически формирующий иерархии в измерениях, удовлетворяющие перечисленным правилам.

После построения иерархий измерений формирование схемы композиционной таблицы завершено. Далее необходимо определиться со способом формирования ее представления. Для того чтобы в таблице не было лишних значений, в работе [9] предлагается учитывать свойство соединения без потерь информации (СБПИ) [7]. Для реализации этого свойства в авто-

матическом режиме в работе [10] было предложено использовать промежуточное представление данных в виде таблицы соединений. В данной работе используется частный случай этой таблицы: «таблица связанных соединений», для которой используется более «жесткое» условие формирования.

Пусть  $U = \{A_1, A_2, \dots, A_n\}$  некоторое множество атрибутов,  $R$  — исходное отношение, определенное на всем множестве  $U$  и удовлетворяющее зависимостям  $DEP$ , и  $\{R_1, R_2, \dots, R_k\}$  — множество отношений (декомпозиция  $R$ ), определенных на подмножествах атрибутов множества  $U$ .

**Определение 3.3.** Декомпозиция  $\{R_1, R_2, \dots, R_k\}$  обладает свойством соединения без потерь информации (СБПИ), если для любой реализации отношения  $R$ , удовлетворяющей множеству зависимостей  $DEP$ , выполнено:

$$R = \pi_{R_1}(R) \bowtie \pi_{R_2}(R) \bowtie \dots \bowtie \pi_{R_k}(R),$$

где  $\bowtie$  — операция естественного соединения,  $\pi_{R_j}(R)$  — проекция отношения  $R$  по атрибутам отношения  $R_j$ .

Пусть  $C_x = \{R_1, R_2, \dots, R_k\}$  — произвольное множество отношений реляционной БД.

**Определение 3.4.** Множество  $C_x$  будем называть контекстом, если оно удовлетворяет свойству СБПИ на зависимостях  $DEP$ .

**Определение 3.5.** Множество  $C_x$  будем называть псевдоконтекстом, если для него не гарантируется выполнение свойства СБПИ.

Для повышения уровня автоматизации работы пользователя и снижения требований к его квалификации формирование контекстов осуществляется по исходным множествам атрибутов  $X, Y_i, Z_i$  ( $i=1, 2, \dots, N$ ).

Для построения множества отношений, с наибольшей вероятностью удовлетворяющего свойству СБПИ используется ряд эвристических правил. В работе [11] сформулированы правила дополнения базового набора отношений другими отношениями, чтобы в совокупности все отношения удовлетворяли свойству СБПИ, а также рассмотрен алгоритм формирования контекстов и правило формирования псевдоконтекстов. В работе [12] сформулированы правила формирования базового набора отношений для множества заданных атрибутов мер и измерений. Далее предполагаем, что контексты и псевдоконтексты для измерений и контекст приложения, включающий измерения и меры, построены.

Рассмотрим формирование «таблицы связанных соединений»  $(C, I)$  для произвольного контекста:  $C_x = \{R_1, R_2, \dots, R_k\}$ . Пусть  $C$  — схема «таблицы связанных соединений», определенная на множестве атрибутов  $A_1, A_2, \dots, A_n$ ,  $I$  — вектор вхождения длины  $k$ . Определим принцип формирования кортежей  $t \in c$ , где  $c$  — реализация (множество кортежей) схемы отношения  $C$ . Рассмотрим все возможные сочетания без повторений отношений  $R_1, R_2, \dots, R_k$ , удовлетворяющие свойству СБПИ. Пусть  $P = \{R_{m(1)}, R_{m(2)}, \dots, R_{m(s)}\}$  текущее сочетание отношений и  $p'$  его реализации, ограниченная логической формулой  $F: p' = \sigma_{R_{m(1)}} \bowtie R_{m(2)} \bowtie \dots \bowtie R_{m(s)}$ .

Для каждого кортежа  $u \in p'$  формируем кортеж  $t$  по следующим правилам:  $t[A_j] = u[A_j]$ , если атрибут  $A_j$  принадлежит соединению, и  $t[A_j] = emp$  в противном случае, где  $emp$  — пустое значение. Каждому кортежу поставим в соответствие битовый вектор  $I(t) = (I_1(t), I_2(t), \dots, I_k(t))$ , где  $I_j(t) = 1$ , если реализация  $r_j$  схемы  $R_j$  участвует в текущем соединении, и  $I_j(t) = 0$  в противном случае.

Рассмотрим отношение частичного порядка над кортежами  $t \in c$ .

**Определение 3.6.** Кортеж  $t \in c$  является менее определенным или равным кортежу  $t' \in c$ , когда для любого

атрибута  $A_j$  выполнено: если  $t[A_j] \neq t'[A_j]$ , то  $t[A_j] = emp$  и  $I_j(t') \geq I_j(t)$ ,  $j=1, \dots, k$ . В этом случае будем писать:  $t < t'$  и назовем кортеж  $t$  подчиненным кортежу  $t'$ .

В представлении достаточно хранить только кортеж  $t'$ , который содержит в себе все менее определенные либо равные кортежи. Следовательно, завершающим этапом построения представления  $s$  является удаление в нем всех подчиненных кортежей.

По предложенным правилам строятся таблица связанных соединений  $s$ , соответствующая контексту приложения  $C_{full}$  и измерения. Построенные измерения служат «каркасом» композиционной таблицы  $R^*$ , а ее заполнение осуществляется из представления  $s$  по перечисленным ниже правилам.

Атрибуты измерения  $X$  остаются неизменными в  $R^*$  и являются наименованиями строк, значения атрибутов  $Y_i$  становятся именами столбцов в  $R^*$ , домены атрибутов  $Z_i$ , дополненные пустым значением, распределяются между доменами новых атрибутов, введенных для значений  $Y_i$ .  $W_i$  – дополнительное множество атрибутов, отсутствующих в  $R^*$ . Естественными являются ограничения:  $X \cap Y_i = \emptyset$ ,  $X \cap Z_i = \emptyset$ ,  $Y_i \cap Z_i = \emptyset$  ( $i = 1, 2, \dots, N$ ).  $W_i \in R \setminus (X \cup Y_1 \cup \dots \cup Y_N \cup Z_1 \cup \dots \cup Z_N)$ ,  $|Dom(Y_i)| = L_i$ ,  $|Z_i| = M_i$ , где  $Dom(Y_i)$  область значения атрибута  $Y_i$  в исходной БД.

Схема результирующего представления строится из исходных отношений по следующему правилу:

$$Sch(C) = \{X, Y_1, \dots, Y_N, Z_1, \dots, Z_N, W_1, \dots, W_N\} \Rightarrow \\ \Rightarrow Sch(CT) = \{X, \text{и } Dom(Y_i) \times \{Z_i\} \ (i=1, 2, \dots, N)\},$$

где  $Sch$  – схема описания отношения,  $Dom$  – область значений атрибута,  $Dom(Y_i) = Dom(Y_{i1}) \times Dom(Y_{i2}) \times \dots$ ,  $Y_{ij} \in Y_i$ .

В данной работе предлагается отказаться от необходимости выполнения функциональных зависимостей вида:  $X, Y_i \rightarrow Z_i$ ,  $i=1, 2, \dots, N$ , что позволит иметь в одной ячейке гиперкуба несколько значений (список) атрибутов  $Z_i$ .

**Определение 3.7.** Множество атрибутов  $KZ_{jp}$  будем называть ключом атрибута  $Z_{jp} \in Z_j$  в контексте  $P$ , если  $KZ_{jp} \subseteq P$ , зависимость  $KZ_{jp} \rightarrow Z_{jp}$  выводима в  $FD^0$ , и не существует выводимой в  $FD^0$  зависимости  $Y \rightarrow Z_{jp}$ , где  $Y \subset KZ_{jp}$  и  $FD^0$  множество функциональных зависимостей на атрибутах отношений из  $P$ .

**Определение 3.8.** Значение атрибута  $t\{Z_{jp}\}$ , где  $Z_{jp} \in Z_j$ , для текущего кортежа  $t \in C$  дублирует значение  $t\{Z_{jp}\}$ ,  $t' \in C$ , если:

- 1)  $t\{Z_{jp}\} = t'\{Z_{jp}\}$ ,
- 2)  $t\{X\} = t'\{X\}$ ,  $t\{Y_i\} = t'\{Y_i\}$ ,
- 3)  $t\{KZ_{jp}\} = t'\{KZ_{jp}\}$ .

Смысл определения 3.8 следующий: если в выбранном контексте есть отношение, в котором идентифицируется (функционально определяется) отдельное значение атрибута, то это значение является важным для приложения, и если оно совпадает с другим значением этого же атрибута, то это не будет дублированием. В противном случае в контексте приложения значения параметра интерпретируются как список возможных значений, тогда в списке не должно быть совпадающих значений. Если в контексте приложения для атрибута меры  $A_j$  отсутствует ключ, то по аксиоме рефлексивности [7] ключом будет сам атрибут. Следовательно, в ячейках для этого атрибута не может быть совпадающих значений.

#### 4. Анализ данных

Для решения задачи диагностирования пациентов кардиологического диспансера было использовано

представление «композиционной таблицы» со следующей схемой:

- атрибуты множества  $X$ : диагноз, № истории болезни, возраст;
- атрибуты множества  $Y_i$ : вид обследования, численный показатель;
- атрибуты множества  $Z_i$ : значение численного показателя;

На множество  $X$  задано следующее ограничение:  $F_0(X) = \{(diagnos = \text{Артериальная гипертензия, стадия 1}) \vee (diagnos = \text{Артериальная гипертензия, стадия 2}) \vee (diagnos = \text{Артериальная гипертензия, стадия 3})\}$ .

Для расчета диагностической шкалы [13] была использована линейная комбинация  $N$  значимых параметров, называемая в литературе [14, 15] решающей функцией.

Рассмотренная технология была использована при построении шкалы оценки тяжести артериальной гипертензии. Для этого были рассмотрены выписки из историй болезни 1443 пациентов кардиологического центра.

#### 5. Заключение

Рассмотренная технология формирования композиционной таблицы позволяет существенно автоматизировать работу пользователя за счет учета свойств схемы исходной базы данных. При этом не требуется преобразование этой схемы к виду «звезда» или «снежинка», что позволяет сохранить положительные свойства исходной операционной базы данных. Рассмотренная технология не навязывает единственно возможное распределение таблиц между мерами и измерениями, как это сделано в Microsoft Analysis Services, и не отдает все на откуп пользователю, как это сделано в ORACLE Analytic Workspace Manager, а предлагает на выбор различные комбинации отношений, из которых исполняющая среда может корректно построить гиперкуб.

Разработанная технология ориентирована, прежде всего, на использование визуального и статистического анализа данных при поиске скрытых закономерностей. Дальнейшее развитие системы предполагается провести по пути создания библиотеки модулей интеллектуального анализа данных с описанием шаблонов схем входных параметров.

#### Библиографический список

1. Vassiliadis, P. A survey of logical models for OLAP databases [Текст] / P. Vassiliadis, T. Sellis // SIGMOD Rec. – 1999. – V. 28. – № 4. – P. 64–69.
2. Pedersen, T. B. A foundation for capturing and querying complex multidimensional data [Текст] / T. B. Pedersen, C. S. Jensen, C. E. Dyreson // Inf. Syst. – 2001. – V. 26. – № 5. – P. 383–423.
3. Li, H.-G. Progressive ranking of range aggregates [Текст] / H.-G. Li, H. Yu, D. Agrawal, A.E. Abbadi // Data & Knowledge Engineering. – 2007. – V. 63. – № 1. – P. 4–25.
4. Giorgini, P. Goal-oriented requirement analysis for data warehouse design [Текст] / P. Giorgini, S. Rizzi, M. Garzetti // In Proceedings of the 8th ACM international Workshop on Data Warehousing and OLAP: DOLAP '05. – 2005. – P. 47–56.
5. Lechtenborger, J. Multidimensional normal forms for data warehouse design [Текст] / J. Lechtenborger, G. Vossen // Inf. Syst. – 2003. – V. 28. – № 5. – P. 415–434.
6. Зыкин, С. В. Формирование гиперкубического представления реляционной базы данных [Текст] / С. В. Зыкин // Программирование. – 2006. – № 6. – P. 71–80.
7. Дейт, К. Дж. Введение в системы баз данных [Текст] / К. Дж. Дейт. – М.: Издательский дом «Вильямс», 2008. – 1328с. – ISBN 978-5-8459-0788-2.
8. Редреев, П. Г. Построение иерархий в многомерных моделях данных [Текст] / П. Г. Редреев // Известия Саратовского уни-



верситета. Новая серия. Сер. Математика. Механика. Информатика. — 2009. — № 5. — С. 84–87.

9. Miller, L. Data Warehouse Modeler: A CASE Tool for Warehouse Design [Текст] / L. Miller, S. Nila // Thirty-First Annual Hawaii International Conference on System Sciences. — 1998. — № 6. — P. 42–48.

10. Зыкин, С. В. Построение отображения реляционной базы данных в списковую модель данных [Текст] / С. В. Зыкин // Управляющие системы и машины. — 2001. — № 3. — С. 42–63.

11. Зыкин, С. В. Автоматизация формирования представлений данных для их аналитической обработки [Текст] / С. В. Зыкин, А. Н. Полуянов // Вестник компьютерных и информационных технологий. — 2010. — № 4. — С. 3–9.

12. Редреев, П. Г. Построение табличных приложений со списочными компонентами [Текст] / П. Г. Редреев // Информационные технологии. — 2009. — № 5. — С. 7–12.

13. Александрович, Ю. С. Оценочные и прогностические шкалы в медицине критических состояний. Справочник. [Текст] / Ю. С. Александрович, В. И. Гордеев. — СПб.: Сотис, 2010. — 248 с. — ISBN 978-5-93979-269-1/

14. Журавлев, Ю. И. Об алгебраическом подходе к решению задач распознавания или классификации [Текст] / Ю. И. Журавлев // Проблемы кибернетики. — 1978. — Т. 33. — С. 5–68.

15. Лбов, Г. С. Метод адаптивного поиска логической решающей функции [Текст] / Г. С. Лбов, В. М. Неделько, С. В. Неделько // Сибирский журнал индустриальной математики. — 2009. — Т. XII. — № 3(39). — С. 66–74.

**ЗЫКИН Сергей Владимирович**, доктор технических наук, профессор (Россия), профессор кафедры «Прикладная математика и фундаментальная информатика» Омского государственного технического университета.

Адрес для переписки: 644050, г. Омск, пр. Мира, 11.

**РЕДРЕЕВ Павел Григорьевич**, математик, системный программист.

Адрес для переписки: e-mail: redreev@mail.ru

**ЧЕРНЫШЕВ Андрей Кириллович**, доктор медицинских наук, профессор (Россия), профессор кафедры «Детская хирургия» Омской государственной медицинской академии.

Адрес для переписки: 644050, г. Омск, пр. Мира, 11.

Статья поступила в редакцию 08.04.2011 г.

© С. В. Зыкин, П. Г. Редреев, А. К. Чернышев

УДК 004.735

**В. И. НИКОНОВ**  
**Е. В. ЩЕРБА**  
**М. В. ЩЕРБА**

Омский государственный  
технический университет

## АНАЛИЗ КОМПЛЕКСНОГО ПОДХОДА К ЗАЩИТЕ ИНФОРМАЦИИ ПРИ ЕЁ ПЕРЕДАЧЕ В РАСПРЕДЕЛЕННЫХ БЕСПРОВОДНЫХ СЕТЯХ

Стандарт IEEE 802.11s описывает новые возможности беспроводных сетей. Основанный на концепции Mesh-сетей, он позволяет интегрировать различные технологии при создании зоны информационного покрытия большой площади. Реализация Mesh-сетей масштаба города и более повысит значимость вопросов защиты передаваемых данных. В исследовании представлен альтернативный подход, позволяющий снизить вероятность успешной атаки при передаче информации в беспроводных сетях без применения алгоритмов шифрования.

**Ключевые слова:** беспроводные сети, безопасность компьютерных сетей, сетевые атаки, маршрутизируемый сервис, мультиплексирование трафика, защита информации.

### Введение

Настоящая работа продолжает исследование [1], в котором был разработан маршрутизируемый сервис передачи данных через распределенные сети. В [1] было обосновано, что использование маршрутизируемого сервиса позволяет значительно снизить вероятность класса активных сетевых атак злоумышленника без использования каких-либо инструментов шифрования.

Развитие группы стандартов IEEE 802.11 с каждым годом позволяет решать все более широкий спектр задач. Недавнее появление нового класса широкополосных беспроводных сетей Mesh позволило достичь значительного увеличения зоны информационного

покрытия. Выбор в пользу беспроводных технологий позволяет получить преимущества в срочности и мобильности, но задачи обеспечения конфиденциальности, целостности и доступности передаваемой в этих сетях информации по-прежнему остаются актуальными.

В представленном исследовании предложена методика повышения безопасности передачи информации в распределенных беспроводных сетях масштаба города и более.

Реализация данного подхода заключается во введении в существующую архитектуру сети протокола, корректирующего работу маршрутизаторов и точек доступа для маркированной информации. Приведено

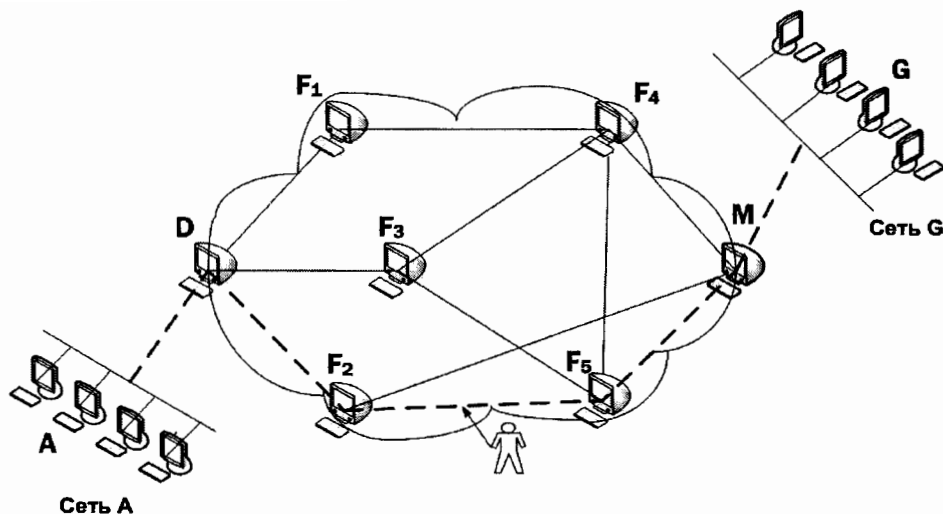


Рис. 1. Работа протоколов маршрутизации между A и G в момент  $\Delta t$ .  
Возможный вариант атаки на участке  $F_2 - F_3$

научное обоснование представленных алгоритмов маршрутизации трафика. Описаны варианты воздействия на систему, рассчитана вероятность успешной активной сетевой атаки в распределенной беспроводной сети с корректирующим протоколом. Обоснована возможность комплексного применения протокола с методами защиты конфиденциальности и обеспечения доступности передаваемой информации, построенными на основе системы мультиплексирования трафика. Полученные результаты представляют интерес для различных применений в области информационных технологий.

#### Вопросы безопасности и атаки в беспроводных сетях

Базовый стандарт 802.11 включает механизм Wired Equivalent Privacy (WEP), который встроен в протокол 802.11. Несмотря на то, что данный механизм обеспечивает защиту передаваемой информации на канальном уровне модели OSI и использует для шифрования 40-разрядный ключ, его стойкость оказалась явно недостаточной. Кроме того что WEP не обладает какими-либо механизмами аутентификации пользователей как таковой, а его ненадежность состоит, прежде всего, в криптографической слабости алгоритма шифрования RC4 [2]. Ещё одна проблема WEP заключается в использовании слишком похожих ключей для различных пакетов данных. Для решения этой проблемы была разработана спецификация 802.11i, объединяющая подмножества Wi-Fi Protected Areas (WPA) и 802.1x. Для усиления шифрования в WPA применяется Temporal Key Integrity Protocol (TKIP), благодаря которому каждый передаваемый пакет наделяется своим собственным ключом (в отличие от WEP, где ключи одинаковы в течение всей сессии). Принятие в июне 2004 г. спецификации WPA2 позволило добавить к списку механизмов защиты протокол CCMP и симметричный алгоритм шифрования AES, за счет чего WPA2 стал более защищенным. В то же время, скорость шифрования AES недостаточно высока при потоковой обработке больших объемов данных, а вопрос безопасности алгоритма по-прежнему остается открытым [3]. Кроме того, в настоящее время ведутся активные исследования в области классификации и описания сетевых атак [4].

Наиболее распространен класс так называемых активных сетевых атак, для осуществления которых

злоумышленнику необходимо напрямую совершить взаимодействие с некоторой системой, являющейся частью сети. Набор инструментов столь же широк: создание перегрузок серверов, эксплуатация недостатков протоколов, использование уязвимостей программного обеспечения. В международной литературе по вопросам информационной безопасности примеры таких атак можно встретить под названиями «sniffing», «flooding», «smurf», «spoofing», «hijacking» и др.

Атаки в беспроводных сетях, в том числе Mesh, в большинстве своем аналогичны вышеупомянутым. Mesh-маршрутизатор – устройство, выполняющее схожие функции, что и классический маршрутизатор в сети Интернет, поэтому он уязвим к таким атакам, как прослушивание и отказ в обслуживании. Также в [5] отмечается, что Mesh-маршрутизаторы, как правило, физически не защищены, так как устанавливаются на крышах зданий или закрепляются на столбах. Данный факт делает возможным подмену одного из узлов сети. Другая уязвимость связана с отличительной особенностью Mesh-сетей от остальных беспроводных сетей. В процессе передачи от одного клиента другому данные могут пройти несколько точек доступа (Mesh-маршрутизаторов). Чем больше звеньев в пути следования трафика, тем сложнее гарантировать безопасность на конкретном участке.

Таким образом, существующих на сегодняшний день методов защиты возможно достаточно для защиты на уровне абонентского трафика, но на уровне корпоративных пользователей требуется использование дополнительных механизмов, отвечающих специфике Mesh-сетей.

В работе разработан принципиально иной подход повышения устойчивости системы при целенаправленных воздействиях активного характера, позволяющий снизить вероятность успешной атаки при передаче информации в беспроводных сетях без применения алгоритмов шифрования и, как следствие, без ограничений с этим связанных.

Один из видов активных сетевых атак является класс атак, основанных на sniffing [4]. Например, нарушитель, обладая знаниями, что некоторая организация регулярно передает данные из A в G, может довольно точно определить маршрут от A до G в момент времени  $\Delta t$  и осуществить перехват на каком-нибудь из участков следования трафика (рис. 1).  $F_1, F_2, F_3, F_4, F_5$  – передатчики системы мультиплексирования трафика,



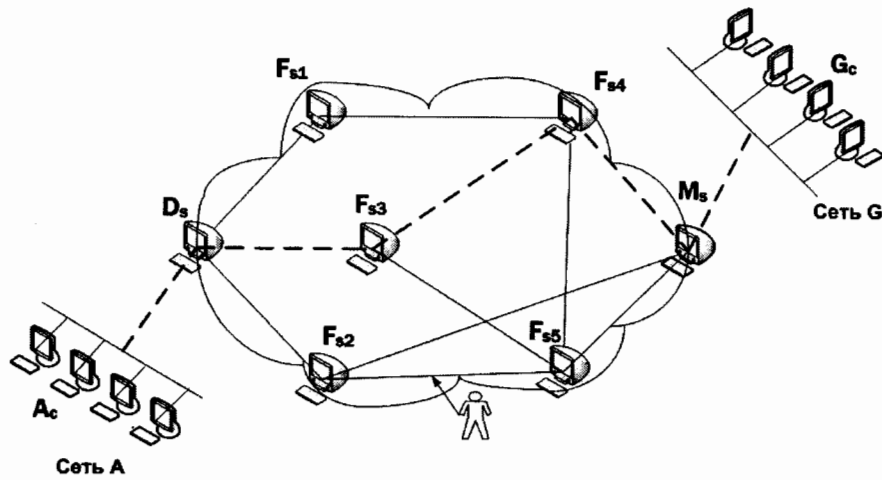


Рис. 2. Изменение маршрута трафика за счет использования маршрутизируемого сервиса  $S_M$  на передатчиках  $F_i, i \in [1, 5]$ ,

в случае ее использования в распределенной сети, либо в общем случае некоторые узловые сервера, необходимые для пространственного представления маршрута следования трафика. Производя посылку трассировочных пакетов, нарушитель в момент времени  $\Delta t$  определил маршрут следования трафика (показано пунктиром) и произвел атаку на подконтрольном маршрутизаторе, расположенном на участке  $F_2 F_5$ .

В исследовании разработано приложение «маршрутизируемый сервис» ( $S_M$ ), позволяющее повысить безопасность передачи информации в распределенных беспроводных сетях в условиях воздействия активных атак.  $S_M$  – клиент-серверное приложение, позволяющее пользователю передавать данные специфичным маршрутом.

Приведем описание компонентов  $S_M$ .

$$S_M = \{S_{MS}, S_{MC}\}.$$

$S_{MC}$  – клиентская часть  $S_M$ , которая устанавливается на компьютерах пользователей и предоставляет диалог для инициализации процесса передачи информации с помощью маршрутизируемого сервиса.

$S_{MS}$  – серверная часть  $S_M$ , которая устанавливается на доверенном сервере и выполняет динамическую маршрутизацию информации, поступающей на этот сервер.

$$S_{MS} = \{F_S, F_S^{gocm}, M, f\}.$$

$F_S = \{F_{S1}, F_{S2}, \dots, F_{SF}\}$  – множество доверенных серверов сети. Под доверенным сервером понимается многофункциональный сервер распределенной беспроводной сети, к которому нарушитель не имеет доступа.

$F = |F_S|$  – количество доверенных серверов сети. В рамках терминологии системы мультиплексирования трафика в роли доверенного сервера может выступать и передатчик при выполнении вышеприведенных условий.

$F_S^{gocm} = \{F_{S1}^{gocm}, F_{S2}^{gocm}, \dots, F_{SF}^{gocm}\}$  – множество, описывающее количество доступных доверенных серверов в начальный момент времени  $t_0$ , а затем через интервалы времени, равные  $\tau$ .  $F_{Si}^{gocm}$  – количество доступных доверенных серверов для  $F_{Si}, i \in [1, F]$ .

$M = \{M_1, M_2, \dots, M_F\}$  – множество матриц маршрутизации. Матрица маршрутизации  $M_i$  формируется на сервере  $F_{Si}$  в начальный момент времени  $t_0$ , а затем

перестраивается через интервалы  $\tau, i \in [1, F]$ . Каждая матрица  $M_i \in F_{Si}$  содержит элементы  $m_{kj}$ , характеризующие доступность доверенных серверов относительно друг друга из  $F_{Si}, k \in [1, F], j \in [1, F]$ .

$F_S^{gocm}$  вычисляется с помощью элементов матрицы  $M_i$  следующим образом:

$$F_{Si}^{gocm} = \sum_{j=1}^F m_{ij} \quad (1)$$

$f$  – параметр, определяющий количество используемых доверенных серверов на всем маршруте от отправителя до получателя в течение одного сеанса (размер «кластера сеанса передачи»).

На доверенных серверах из множества  $F_S$  устанавливается серверная часть сервиса –  $S_{MS}$ , выполняющая автоматическую «интеллектуальную» маршрутизацию трафика.

Показано, что использование  $S_M$  позволило избежать прохождения трафиком подконтрольного нарушителю участка (рис. 2). Данное решение  $S_M$  (итоговый маршрут) является вероятностным с вероятностью принятия  $p_j, 0 < p_j \leq 1, j \in [1, k]$ . Где  $k$  – количество различных маршрутов от  $D_S$  до  $M_S$  на графе с вершинами  $D_S, F_{S1}, F_{S2}, F_{S3}, F_{S4}, F_{S5}, M_S$  и ребрами, определяемыми текущей топологией сети. Расчет значений  $p_j$  будет рассмотрен далее.

В процессе передачи с помощью  $S_M$  данные проходят через некоторое число доверенных серверов, равное  $f$ . Выбор каждого следующего сервера происходит динамически. Учитывая приведенное выше определение таблиц маршрутизации для  $S_{MS}$ , выбор каждого следующего сервера описывается гипергеометрическим распределением.

Для гипергеометрического распределения вероятности принятия случайной величиной  $y$  значения  $y_0$  имеет вид:

$$p(y = y_0 | n, d, c) = \frac{\binom{c}{y_0} \binom{n-c}{d-y_0}}{\binom{n}{d}}. \quad (2)$$

Для маршрутизируемого сервиса вышеприведенные параметры имеют следующие значения:

$n = F$  – количество доверенных серверов;  
 $d_i = F_{Si}^{gocm}(\tau)$  – количество доступных доверенных серверов для  $F_{Si}$  в момент времени  $\tau$ ;

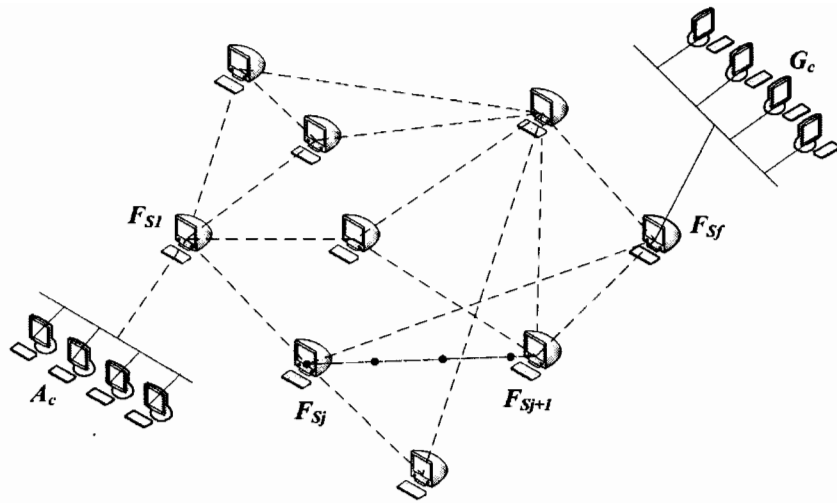


Рис. 3. Возможное представление беспроводной распределенной сети с контролируемым нарушителем участком  $F_{Sj}F_{Sj+1}$

$c = 1$  – количество выбираемых доверенных серверов на каждом этапе передаче;

$y_0 = 1$  – количество доступных доверенных серверов в выборке.

Таким образом, выбор каждого следующего сервера описывается гипергеометрическое распределение  $HG(1, F_{Si}^{gocm}, F)$ .

Итоговый маршрут трафика от отправителя до получателя при использовании  $S_M$  и  $f$  доверенных серверов из  $F$ , находящихся в сети, будет выбран с вероятностью:

$$p_j = \prod_{i=0}^{f-1} \frac{\binom{F-i-1}{F_{Si}^{docm} - i - 1}}{\binom{F-i}{F_{Si}^{docm} - i}}, j \in [1, k]; \quad (3)$$

$F_{Si}^{gocm}$  – число доступных серверов для  $F_{Si}$  при выборке  $F_{S(i+1)}$  доверенного сервера на  $i + 1$  шаге, определяемое формулой (1).

Формула (3) определяет вероятность построения системой  $S_M$  одного из возможных маршрутов, использующего только доступные доверенные сервера. Если рассматривать систему доверенных серверов как статическую систему, то из (3) следует неизменность значения  $p_j$ . Но в реальности система таковой не является, в разные моменты времени каждый из доверенных серверов может быть как доступен, так и недоступен. Данный факт учитывается в формуле (3) с помощью параметров  $F_{Si}^{gocm}$ .

В формуле (3) используются выборки без возвращения. Напомним, что

$$\binom{n}{k} = C_n^k = \frac{n!}{k!(n-k)!} \quad (4)$$

Значение формулы (4) также называется биномиальным коэффициентом. Преобразуем формулу (3), раскрыв все биномиальные коэффициенты:

$$p_j = \prod_{i=0}^{f-1} \frac{\binom{F-i-1}{F_{Si}^{docm} - i - 1}}{\binom{F-i}{F_{Si}^{docm} - i}} = \prod_{i=0}^{f-1} \frac{C_{F-i-1}^{F_{Si}^{docm} - i - 1}}{C_{F-i}^{F_{Si}^{docm} - i}} = \prod_{i=0}^{f-1} \frac{F_{Si}^{docm} - i}{F - i}, j \in [1, k], \quad (5)$$

В формуле (5) показана вероятность построения сервисом  $S_M$  маршрута трафика от отправителя до получателя при использовании в процессе передачи  $f$ -доверенных серверов.

Значительно больший интерес с точки зрения оценки уязвимости  $S_M$  представляет расчет вероятности построения  $j$ -го маршрута (из  $k$ -возможных) в том случае, когда контролируется участок, например, между  $i$ -ым и  $(i+1)$ -ым доверенными серверами, входящими в  $j$ -ый маршрут.

Представим, что нарушитель получил несанкционированный доступ к единственному физическому каналу из всего множества каналов задействованных в системе и не имеет доступа к остальным каналам. То есть он имеет возможность анализа, блокирования и модификации всей информации, передаваемой по этому каналу (рис. 2).

В этом случае представляется возможным вычислить оценку успешной атаки первого класса  $R_{A1}$ , когда нарушитель контролирует участок между доверенными серверами  $F_{Sj}$  и  $F_{Sj+1}$  (рис. 3). При неизвестном пространственном расположении  $F_S$  считаем атаку успешной, если при работе сервиса  $S_M$  сервера  $F_{Sj}$  и  $F_{Sj+1}$  были выбраны на  $i$  и  $i + 1$  этапе передачи,  $j \in [1, F], i \in [1, f]$ .

$$R_{A1} = \begin{cases} \frac{2}{F_{S0}^{docm} \cdot (F_{S0}^{docm} - 1)}, f = 2 \\ \frac{2}{F_{S0}^{docm} \cdot (F_{S0}^{docm} - 1)} + \sum_{i=1}^{f-2} \left[ \frac{2}{F_{S0}^{docm} - i} \cdot \frac{1}{F_{S0}^{docm} - i - 1} \right] \cdot \left[ \prod_{j=1}^i \frac{F_{S0}^{docm} - j - 1}{F_{S0}^{docm} - j + 1} \right], f > 2. \end{cases} \quad (6)$$

Результаты экспериментальных исследований подтвердили применимость разработанной методики. Данная методика прошла внедрение в системы передачи информации ОАО «Омскводоканал» и ГОУ ВПО «Омский государственный технический университет». Показана возможность встраивания в  $S_M$  алгоритмов мультиплексирования [6]. Таким образом, можно объединить два подхода к обеспечению безопасности передаваемой информации: с одной стороны, снизить вероятность доступа злоумышленника к используемым каналам связи, а с другой – применить логическое преобразование информации, позволяющее обеспечить дополнительную защиту конфиденциальности или доступности передаваемой информации, в соответствии с выбранной политикой

безопасности. Общая схема логического преобразования реализуется посредством системы, выполняющей разделение данных по нескольким разнесенным каналам передачи таким образом, что, с физической точки зрения, перехват всех частей затруднителен и сложность восстановления исходной последовательности без какой-либо ее отдельной части максимальна или минимальна. В указанной схеме системы разделения данных можно выделить три основных элемента: мультиплексор, демультиплексор и передатчики, в роли которых при этом могут выступать доверенные сервера  $S_M$  [7, 8]. Отличительной особенностью предлагаемой схемы является возможность обеспечения максимальной доступности передаваемой информации посредством сегментации исходных данных на уровне содержания, обеспечивающей наилучшее восстановление исходных данных законным получателем, в случае блокирования или модификации одной или нескольких передаваемых частей нарушителем [9].

#### Заключение

В результате выполнения исследовательских работ, был реализован «маршрутизируемый сервис» передачи данных через распределенные сети. Выработаны основные компоненты, необходимые для функционирования системы. Даны оценки вероятностям сетевых атак на передаваемую информацию в случае применения «маршрутизируемого сервиса». Обоснована возможность комплексного применения сервиса с методами защиты конфиденциальности и обеспечения доступности передаваемой информации, построенными на основе системы мультиплексирования трафика. Произведена апробация сервиса на глобальной сети предприятия.

Использование «маршрутизируемого сервиса»  $S_M$  для передачи данных через распределенные сети позволяет значительно снизить вероятность класса активных сетевых атак злоумышленника без применения криптографических алгоритмов.

#### Библиографический список

1. Никонов, В. И. Маршрутизируемый сервис передачи данных через распределенные сети / В. И. Никонов // Материалы Всерос. конкурса-конференции студентов и аспирантов по информационной безопасности SIBINFO-2009 / Томск, ТУСУР, 21 –

22 апреля 2009. Научная сессия ТУСУР 2009. – Томск, 2009. – Т. 1. – С. 92–95.

2. Tews E., Weinmann R.-P., Pyshkin A. Breaking 104 bit WEP in less than 60 seconds [Электронный ресурс]. – Режим доступа : <http://eprint.iacr.org/2007/120.pdf>, свободный (дата обращения : 01.03.2011).

3. Bangerter E., Gullasch D., Krenn S. Cache Games – Bringing Access-Based Cache Attacks on AES to Practice [Электронный ресурс]. – Режим доступа : <http://eprint.iacr.org/2010/594.pdf>, свободный (дата обращения : 01.03.2011).

4. Kak, A. Port Scanning, Vulnerability Scanning, and Packet / A. Kak // Computer and Network Security. – 2008. – № 23. – P. 29–38.

5. Ben Salem, N. Securing Wireless Mesh Networks / N. Ben Salem, J.-P. Hubaux // IEEE Wireless Communications. – 2006. – № 13/2. – P. 50–55.

6. Никонов, В. И. Маршрутизируемый сервис передачи данных через распределенные сети. / В. И. Никонов // Технологии Microsoft в информатике и программировании : материалы конференции-конкурса / Новосибирск, 1–2 марта 2008. – Новосибирск, 2008. – С. 83–84.

7. Ефимов, В. И. Атака на систему разнесенного TCP/IP трафика на основе анализа корреляции потоков / В. И. Ефимов, Е. В. Щерба // Информационные технологии моделирования и управления. – 2005. – № 6(24) – С. 859–863.

8. Никонов, В. И. Атаки на маршрутизируемый трафик / И. В. Никонов // Информационные технологии моделирования и управления. – 2009. – № 7(59) – С. 962–968.

9. Щерба, Е. В. Метод защиты цифровой видеoinформации при её передаче в распределенных компьютерных сетях / Е. В. Щерба // Прикладная дискретная математика. – 2009. – Приложение № 1. – С. 60–62.

**НИКОНОВ Вячеслав Игоревич**, ассистент кафедры «Средства связи».

Адрес для переписки: e-mail: [vi.nikonov@gmail.com](mailto:vi.nikonov@gmail.com)  
**ЩЕРБА Евгений Викторович**, кандидат технических наук, старший преподаватель кафедры «Комплексная защита информации».

Адрес для переписки: e-mail: [evscherba@gmail.com](mailto:evscherba@gmail.com)  
**ЩЕРБА Мария Витальевна**, аспирантка кафедры «Комплексная защита информации».

Адрес для переписки: e-mail: [mariz3@mail.ru](mailto:mariz3@mail.ru)

Статья поступила в редакцию 08.04.2011 г.

© В. И. Никонов, Е. В. Щерба, М. В. Щерба

## Книжная полка

004.9/3-15

**Задорожный, В. Н. Аналитико-имитационные исследования систем и сетей массового обслуживания [Текст] : монография / В. Н. Задорожный ; ОмГТУ. – Омск, 2010. – 279 с. : рис., табл. – Библиогр. : с. 226-234. – ISBN 978-5-8149-0917-6.**

Монография содержит изложение методологии, теории, методов и алгоритмов аналитико-имитационного моделирования систем и сетей массового обслуживания, основанное на оригинальных исследованиях автора и снабженное большим числом подробно рассмотренных примеров. Будет полезна как математикам, так и специалистам-прикладникам, использующим технологию имитационного моделирования в различных областях практического применения теории массового обслуживания (проектирование вычислительных систем и сетей, оптимизация структуры и режимов работы обслуживающих предприятий, транспорт, медицина, военное дело и т. д.). Отдельные части монографии могут использоваться в учебном процессе университетов и других вузов на факультетах и кафедрах прикладной математики, информатики и вычислительной техники, автоматизации управления и системного анализа в курсах «Теория массового обслуживания», «Моделирование систем», «Теория принятия решений» и др. преподавателями, студентами и аспирантами инженерных и экономических специальностей.

## ПРИМЕНЕНИЕ МЕТОДА КОМБИНАТОРНЫХ СОЧЕТАНИЙ ПРИ АВТОМАТИЗАЦИИ ПРОЕКТИРОВАНИЯ ПОДРОСТКОВОГО ГАРДЕРОБА

В статье рассмотрены основные аспекты метода комбинаторных сочетаний, которые могут быть использованы при автоматизации проектирования подросткового гардероба; сформулирована задача на проектирование гардероба; описывается содержание и структура базы данных для программного комплекса проектирования.

**Ключевые слова:** подростковый гардероб, автоматизация проектирования, комбинаторные сочетания, базовый элемент изделия, база данных, структурирование данных.

### Введение

На сегодняшний день разрабатывается множество различных проектов для систем автоматизации производства, в частности, систем автоматизированного проектирования подростковой одежды. Автоматизация значительно увеличивает потенциал швейного производства, она позволяет наращивать объемы выпуска продукции, оптимизирует производственный процесс на всех стадиях. Сохраняется тенденция усовершенствования в направлении компьютерной интеграции от проектирования и производства продукции к продажам и коммерческим операциям [1].

Тема исследования в области проектирования подросткового гардероба с использованием методов комбинаторики актуальна, поскольку дает возможность добиться максимального производственного результата, позволяя при этом минимизировать материальные затраты на изготовление.

### 1. Постановка и решение комбинаторной задачи

Задача по созданию подросткового гардероба из геометрических элементов носит комбинаторный характер. Применение метода комбинаторики [2, 3] для создания конструкций подростковой одежды является целесообразным в силу обеспечения некоторого множества изделий определенным количеством унифицированных элементов, которые используются не в одном изделии, а в нескольких изделиях в пределах одного проектируемого гардероба.

Одним из основных принципов комбинаторики является принцип умножения. Суть его заключается в следующем: если существует возможность сделать выбор какого-либо элемента  $n$ -способами, и затем другого элемента, комбинируемого с ним,  $n_2$ -способами, третьего  $n_3$ -способами и т. д., то общее количество всевозможных комбинаций  $N$  равно:

$$N = n_1 \cdot n_2 \cdot n_3 \cdot n_k, \quad (1)$$

где  $n$  — количество способов выбора  $k$ -го геометрического элемента;  $k$  — количество деталей в комплекте.

Подсчет количества комбинаций необходим для того, чтобы автоматизированная система, пользуясь

этим подсчетом, предоставляла пользователю возможность выявить оптимальное соотношение количества элементов и изделий в гардеробе.

Для удобства подсчета количества комбинаций геометрических элементов в гардеробе имеет смысл принимать переднюю и заднюю половинки деталей одежды за разные изделия.

Если предположить, что в брюках максимальное количество элементов для выбора  $n_1 = 4$  для передней половинки и  $n'_1 = 4$  для задней половинки; в юбке  $n_2 = 12$  для передней половинки и  $n'_2 = 12$  для задней половинки; в блузе  $n_3 = 16$  для полочки и  $n'_3 = 16$  для спинки; в платье  $n_4 = 16$  для полочки и  $n'_4 = 16$  для спинки; в рукаве  $n_5 = 6$ , то по формуле (1) общее количество комбинаций  $N$  равно:

$$N = 4 \cdot 4 \cdot 12 \cdot 12 \cdot 16 \cdot 16 \cdot 16 \cdot 16 \cdot 6 = 905969664 \quad (2)$$

Из расчета (2) следует, что общее количество комбинаций геометрических элементов в гардеробе велико. Моделировать такой гардероб вручную — трудоемкая и графически трудно представимая работа. Поэтому необходимо выработать алгоритм перебора элементов. Для этого формируется набор базовых элементов для всех базовых (обязательных) изделий в гардеробе, которые могут быть поделены на комбинаторные геометрические элементы, составляющие базовые.

Базовыми изделиями в гардеробе являются брюки, юбка, платье, блуза, рукав; для каждого из них существует определенный набор базовых элементов. Например, передняя половинка брюк имеет 4 базовых элемента определенных размеров, из которых 3 элемента могут быть поделены на меньшие по размерам (рис. 1). Базовые элементы имеют форму геометрических фигур разных размеров, которые сопрягаются своими сторонами по определенным конструктивным линиям изделий.

Один базовый элемент, который не может быть поделенным на комбинаторные элементы, является унифицированной деталью, которая необходима для изготовления брюк, даже при использовании комбинаторного метода проектирования и может быть выбрана для изготовления других брюк в гардеробе.

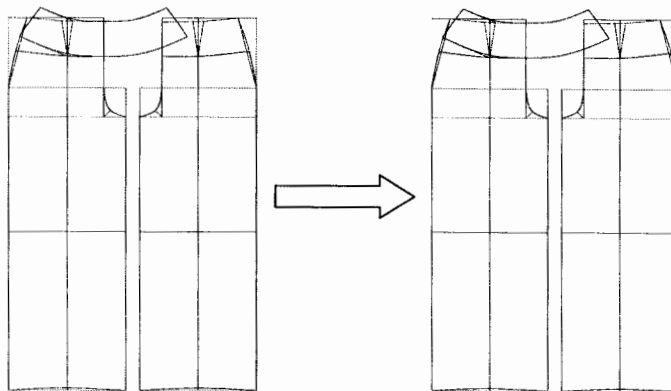


Рис. 1. Пример разделения базовых элементов брюк на комбинаторные геометрические элементы

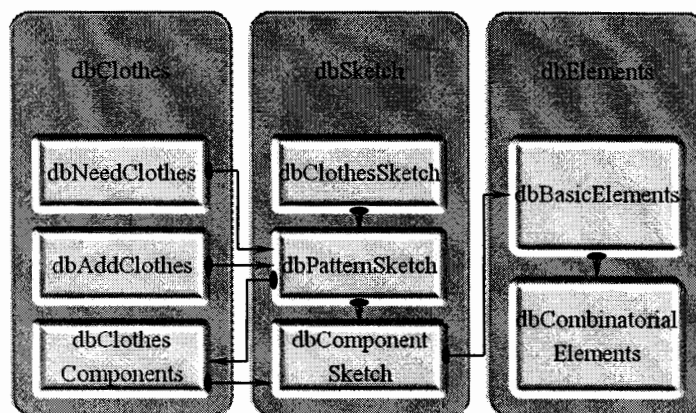


Рис. 2. Структура взаимосвязи баз данных для автоматизированного проектирования подростковой одежды

## 2. Постановка и решение оптимизационной задачи

Кроме комбинаторной задача является также оптимизационной. Задача дискретной оптимизации состоит в следующем: следует подобрать такой набор элементов нужных размеров, чтобы количество изделий в гардеробе было максимальным при минимальном количестве базовых элементов.

На одной из ступеней перехода работы конструктора от ручного проектирования одежды к автоматизированному происходит формализация необходимой информации. Предварительно создаются базы данных для программного комплекса, с помощью которого и происходит автоматизированное конструирование одежды.

Для выявления необходимой информации и формы ее представления в базе данных анализируются основные этапы проектирования и учитываются требования к изготовлению подросткового гардероба, на основании которых выявляются данные для последующего структурирования. Формализованная информация включает несколько баз данных, которые представлены в форме эскизов, таблиц и ID для изделий, их деталей, базовых и комбинаторных геометрических элементов. База данных представляет собой структуру взаимосвязи исследуемой информации (рис. 2).

Эффективная работа программы проектирования подросткового гардероба зависит от корректного представления информации в БД, а также от создания универсального для всех гардеробов алгоритма проектирования.

Пусть есть набор базовых элементов  $\{b_1, \dots, b_m\}$  разных размеров и матрицы:

$$A_{n_1 \times m}, B_{n_2 \times m}, C_{n_3 \times m}, D_{n_4 \times m} \quad (3)$$

Элементы матриц (3) – целые неотрицательные числа, где  $a_{ij}, b_{ij}, c_{ij}, d_{ij}$  равны количеству элементов  $b_i$  в  $j$ -том изделии, где  $b_i$  –  $i$ -тый базовый элемент определенного размера. Матрицы  $A, B, C, D$  являются матрицами для моделей брюк, юбок, платьев, блуз соответственно.

Рассмотрим на примере матрицу  $A$ . Каждая строка матрицы определяет набор деталей для данной модели изделия. Элементы матрицы:

$m$  – количество базовых элементов  $b_i$  разных размеров;

$n_i$  – количество разных моделей изделий. Переменные  $n_i$  и  $m$  являются динамическими:  $n_i$  может изменяться за счет внесения в базу данных новых моделей, а  $m$  – за счет пополнения базы данных геометрических элементов.

Идея проектирования гардероба состоит в разработке алгоритма, в котором формирование комплекта осуществляется последовательным перебором по определенному количеству элементов в комплекте. На каждом этапе перебора происходит последовательный выбор и сравнение строк различных матриц, имеющих наибольшее число равных элементов (совпадений). Перебор элементов может начаться с любой строки матрицы, т. е. с выбора любого изделия проектируемого гардероба.

**Результат**

Согласно установленному алгоритму перебора элементов, который сводится к определённой последовательности действий проектировщика или пользователя, может быть реализована проектирующая автоматизированная система, на выходе которой должен быть представлен комплект комбинаторных геометрических элементов, имеющих конкретные форму и размеры, составляющих изделия гардероба.

**Заключение**

Комбинаторный характер решения задачи проектирования является целесообразным для подросткового гардероба, поскольку позволяет:

— за счет «унифицирования» комбинаторных элементов изделий увеличить количество изделий в одном приобретаемом гардеробе. Это значительно экономит материальные затраты проектирующих предприятий и, следовательно, снижает рыночную стоимость комплекта изделий;

— повысить рентабельность изготовления одежды для омских швейных предприятий, поскольку себестоимость такого подросткового гардероба невысока, а значит? и доступна большинству среднеобеспеченных семей г. Омска.

**Библиографический список**

1. Renee Weiss Chase CAD for fashion design / Renee Weiss Chase; College of Design Arts, Drexel University. — NJ. — 1997. 174 p. — ISBN 0-13-3732428.

2. Коробова, А. Б. Применение методов комбинаторики в формировании рационального подросткового гардероба в автоматизированных системах [Текст] / А. Б. Коробова, О. Н. Демченко // Актуальные проблемы подготовки кадров высшей квалификации : сб. материалов / ОГИС. — Омск, 2010. — С. 15–21.

3. Демченко, О. Н. Методы комбинаторики и трансформации, используемые при изготовлении подростковой одежды [Текст] / О. Н. Демченко // Теоретические знания — в практические дела : сб. статей / РосЗИТЛП. — Омск, 2010. — Ч. 1. — С. 6–8. — ISBN 978-5-903153-16-9.

**ДЕМЧЕНКО Ольга Николаевна**, аспирантка кафедры конструирования швейных изделий.

**КОРОБОВА Антонина Брониславовна**, кандидат технических наук, профессор (Россия), декан художественно-технологического факультета.

**РАССКАЗОВА Марина Николаевна**, кандидат физико-математических наук, доцент кафедры прикладной информатики и математики.

Адрес для переписки : hotta\_muse@mail.ru

Статья поступила в редакцию 28.02.2011 г.

© О. Н. Демченко, А. Б. Коробова, М. Н. Рассказова

УДК 004.421-24-443.5

**М. А. ЖБАННИКОВА**  
**А. Б. КОРОБОВА**  
**А. Г. БУРЦЕВ**

Омский государственный  
институт сервиса

## **ОБЩИЕ КОНЦЕПЦИИ И МОДУЛИ ПРОГРАММНОГО ПРОДУКТА ДЛЯ СНЯТИЯ РАЗМЕРНЫХ ПРИЗНАКОВ ОСОБЕННОСТЕЙ ФИГУРЫ ДЕВУШЕК-ПОДРОСТКОВ В АВТОМАТИЗИРОВАННОМ РЕЖИМЕ**

В статье рассматриваются перспективы использования, общие концепции, модули и принцип работы программного продукта для получения размерных характеристик особенностей фигуры девушек-подростков в автоматизированном режиме и построение конструкции брюк.

**Ключевые слова:** размерные характеристики, построение конструкции, автоматизированное проектирование, язык программирования C, операционная система, библиотеки элементов интерфейса GTK.

**Введение**

В настоящее время значительно вырос интерес производителей к современным технологиям в области легкой промышленности. Главным помощником в решении этих задач являются системы автоматизированного проектирования швейных изделий. Большое

количество новых современных бесконтактных методов измерения, поверхности тела человека являются весьма перспективными.

Существует множество систем, с помощью которых определяются параметры, качественно и количественно характеризующие сложную поверхность.



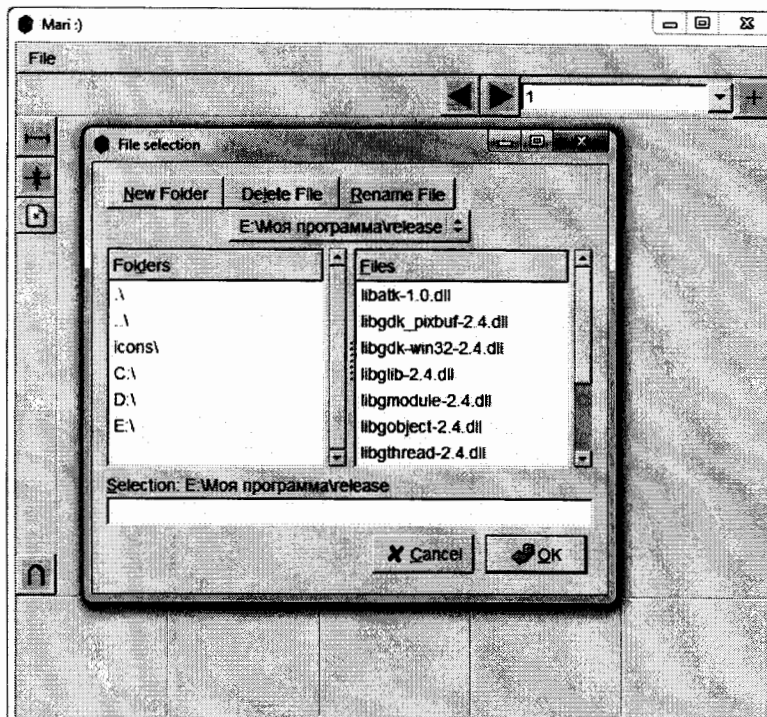


Рис.1. Выбор количества обрабатываемых фотографий

Возможность получения измерений фигуры может быть реализована с помощью современных технических средств: 3D сканеров, сканеров и видеосканеров, цифровых фото- и видеокамер. Несмотря на огромное разнообразие новейших технических средств, использующихся для получения информации о форме и размерах поверхности тела человека, наиболее перспективным остается использование цифровых фотоаппаратов.

Бесспорно, системы боди-сканирования стоят на шаг впереди цифровых фотоаппаратов, т.к. при помощи сканирования можно получить более точное компьютерное изображение человеческого тела, поэтому этот метод трехмерного снятия размерных признаков подходит для построения лекал одежды на людей с любым отклонением от типовой фигуры. Однако, по мнению авторов, в силу высокой стоимости системы боди-сканирования появление данной системы в России в ближайшие несколько лет, а возможно, и несколько десятков лет, не имеют перспектив к развитию и совершенствованию.

Сейчас существуют программы для автоматизации процесса снятия размерных признаков с фигуры и совмещение его с автоматизированным проектированием [1]. Однако, появившись сравнительно недавно, они имеют ряд недостатков, т.к. основной принцип их работы заключается либо в виде подгонки изображения фигуры человека под имеющийся в программе шаблон манекена, либо в определении размерных признаков по контрольным антропометрическим точкам, которые интуитивно определяются оператором [2].

Таким образом, цели совершенствования и повышения эффективности автоматизированного измерения поверхности фигуры человека могут быть сведены к повышению точности, сокращению времени измерения, уменьшению времени анализа обработки данных, повышению информативности и надежности.

**Перспективы использования, общие концепции и модули создаваемого программного продукта**

В современном мире каждый производитель одежды нацелен на снижение себестоимости производимой им одежды. Предложенный автором программный продукт может быть использован на двух платформах Linux и Windows. Данный программный продукт выполнен на языке программирования Си с использованием библиотеки элементов интерфейса GTK.

Стандартизированный процедурный язык программирования Си отличается минимализмом. Си был создан для использования в операционной системе UNIX (Linux является UNIX-совместимой, однако основывается на собственном исходном коде). На данный момент большинство операционных систем написано на языке программирования Си. Поэтому он является самым популярным языком для создания системного программного обеспечения. Его также часто используют для создания прикладных программ. Главные достоинства языка Си – это его эффективность, простота программирования и отладки программ. Несмотря на то, что Си не разрабатывался для новичков, он активно используется для обучения программированию.

Для языка Си характерны лаконичность, современный набор конструкций управления потоком выполнения, структур данных и обширный набор операций. Программы на Си разрабатываются сравнительно легко благодаря относительно низкому уровню языка и скромному набору элементов. Несмотря на это, возможности языка Си колоссальны, на нем может быть написана любая программа, которая написана на любом другом языке. Поэтому данный язык доступен на самых различных платформах (возможно, круг этих платформ шире, чем у любого другого существующего языка). К тому же, несмотря на свою низкоуровневую природу, язык позволяет создавать переносимые программы и поддерживает в этом программиста. Программы, соответствующие стандарту языка, могут компилироваться на самых различных компьютерах [3, 4]. Все выше перечисленные достоинства снизят себестоимость и повысят производительность.



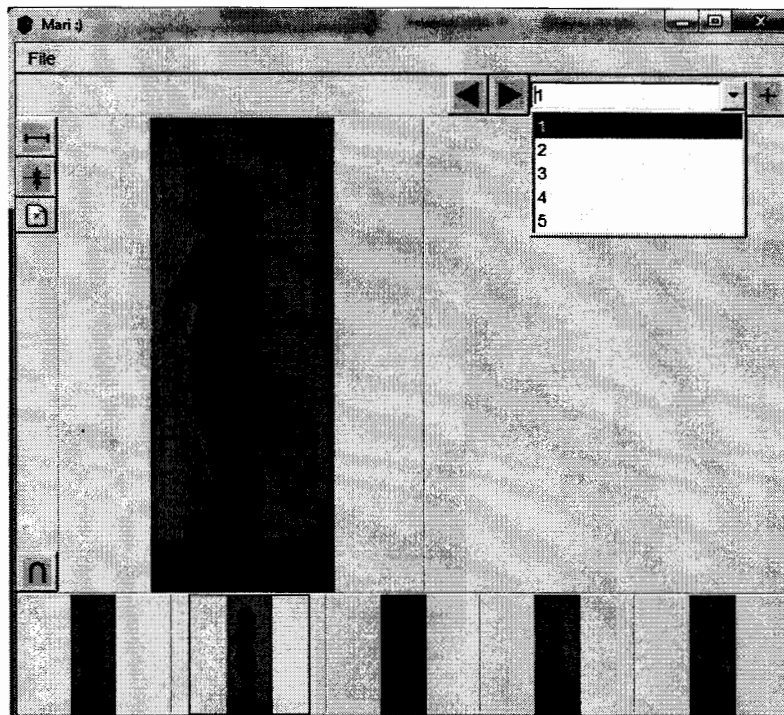


Рис 2. Переход по фотографиям и проведение измерения по набору загруженных фотографий

Гибкая и динамичная система разработки определяет исключительную экономическую эффективность Linux, невозможная для проектов с закрытым кодом [5]. Конечно, такая высокая эффективность разработки не могла не заинтересовать крупные фирмы, которые стали открывать свои проекты. Такие фирмы как IBM и NOVEL участвуют в разработке операционной системы Linux [6, 7].

С другой стороны, открытый код значительно снижает себестоимость разработки закрытых систем для Linux и позволяет снизить цену решения для пользователя [8].

#### Принцип работы разработанного программного продукта

Использование программного продукта возможно в двух режимах:

- создание проекта
- измерение

#### Режим создания проекта

В данном режиме конструктор подготавливает материалы для проведения измерений:

- 1) задается название и развернутое описание проекта;
- 2) задаются параметры проекта:
  - количество обрабатываемых фотографий (рис. 1),
  - расположение фотографий относительно друг друга;
- 3) измерения кривых в проекте:
  - измерения кривых в проекте разбиваются на этапы, для каждого этапа задается список измеряемых кривых;
  - в рамках одного этапа возможно измерение нескольких кривых, для каждой из которых пример измерения представлен на иллюстрации;
  - каждое измерение конструктора привязано к используемой на данном этапе фотографии;
  - для каждой кривой конструктором задается название и условное обозначение;
- 4) дополнительные параметры кривых:

— конструктор может дополнить правила вычисления длины кривой (например, для некоторых частей ломанной возможно задать коэффициент длины для учета кривизны кривой или расположения не параллельно плоскости фотографии).

Подготовленные материалы могут быть сохранены в виде файла проекта, которые в дальнейшем могут использоваться для изменения параметров в «Режиме создания проекта» или для проведения измерения в «Режиме измерения»

#### Режим измерения

В данном режиме работы конструктор использует готовый проект, подготовленный в режиме создания проекта, и проводит измерения по набору загруженных фотографий (рис. 2).

При открытии проекта пользователю сообщается требуемый набор исходных материалов (фотографий) и их расположение относительно друг друга.

Пользователь загружает выбранные фотографии и проводит измерения в установленном порядке (допускается проведение измерений в произвольном порядке с использованием перечня всех измерений), каждое измерение конструктора привязано к используемой фотографии.

Результаты измерений сохраняются в файл в виде таблицы с названиями кривых, условными обозначениями и результатом в сантиметрах. Результаты могут быть сохранены на любом этапе проведения измерения, при этом сохраняются результаты для измеренных кривых. После перезапуска программы возможна загрузка результатов с последующим продолжением измерений с момента сохранения.

В каждом режиме для проведения точных замеров для фотографий применяются следующие операции:

1. Задание масштаба — на фотографии задается один или несколько отрезков с реальным размером в сантиметрах. По заданным отрезкам вычисляется масштаб фотографии по горизонтали и вертикали. Использование нескольких отрезков позволяет задать масштаб с минимальной погрешностью.

2. Выравнивание фотографий по вертикали — предназначено для задания единого уровня по вертикали на всех фотографиях. Выравнивание производится указанием одинаковой точки на двух и более фотографиях. Допускается задание нескольких уровней для уменьшения погрешности выравнивания.

3. Масштабирование и перемещение изображения фотографии в окне просмотра программы производится конструктором, для отображения необходимой области фотографии в увеличенном масштабе для снятия более точных размерных характеристик.

#### Измерение длины кривой

На фотографии длина кривой измеряется аппроксимацией ее ломаной, в дальнейшем планируется расширение функциональности до аппроксимации кривыми Безье. Ломаная, задается как набор точек, устанавливаемых мышкой на фотографии. Для задания точки на фотографии возможно использование функций привязки (см. ниже). Длина ломаной вычисляется с использованием установленного масштаба.

#### Функция привязки

При задании точки на фотографии возможно использование функции привязки. Функция привязки автоматически находит точки на фотографии вблизи курсора мыши. Например, точка, вершина угла, пересечение двух кривых или прямых линий и т.д.

Указанные объекты должны быть контрастными. И могут быть созданы искусственно при фотографировании для облегчения проведения измерений.

Например:

- булавкой с контрастной головкой;
- небольшим клеящимся стикером;
- нанесением кривых на одежде фотографируемого;
- привязка к уже имеющимся точкам и прямым (кривым).

#### Измерение кривых 3D, создание 3D каркаса

Для проведения измерений в 3D необходимо:

1) знание расположения фотографий относительно друг друга, задается на этапе создания проекта, имеется возможность корректировки на этапе измерений;

2) задание каждой точки с использованием двух и более фотографий. Если точка задана только на одной фотографии, то на других она отображается, как прямая соответствующая проекции прямой проходящей через заданные точки и перпендикулярной плоскости фотографии на которой эта точка задана;

3) каркас создается заданием нескольких точек на поверхности измеряемого объекта с автоматической триангуляцией полученной сетки.

На этапе создания 3D каркаса будет возможна дополнительная функция развертка каркаса фигуры. С помощью этой функции конструктор сможет получить точную развертку фигуры относительно линии бедер с учетом всех возможных изгибов нижней опорной поверхности фигуры девушки-подростка. В программу заложена конструкция брюк на типовую фигуру, размер и рост на данном этапе проектирования будет вычисляться автоматически в соответствии с полученными данными измерения фигуры в одном из двух режимов. При работе с нежелательными особенностями фигуры девушки-подростка конструктору будет предложен выбор двух смежных размеров приближенных к размерам фигуры. Наложение конструкции на развертку нижней опорной поверхности фигуры производит конструктор, ему будет предложено выбрать наложение первого, второго или обоих размеров, которые на развертке будут обозначены другими линиями.

#### Получение конструкции на индивидуальную фигуру

Для получения конструкции на индивидуальную фигуру конструктор производит преобразование уже имеющейся конструкции в соответствии с предложенными коэффициентами из классификации особенностей нижней опорной поверхности. После корректировки по коэффициентам конструктор получает чертеж конструкции брюк.

#### Заключение

Находящаяся в постоянном изменении сфера швейной промышленности на сегодняшний день требует научно обоснованных подходов к проектированию и производству изделий, базирующихся на последних достижениях в области информационных технологий. Современные тенденции развития швейной промышленности определяют необходимость постоянного использования средств вычислительной техники в процессе проектирования изделий.

Проект позволяет провести снятие размерных признаков с фигуры дистанционно и получить чертеж конструкции брюк.

Промышленные системы автоматизированного проектирования (САПР) одежды предоставляют специалистам большое количество программных средств для выполнения проектно-конструкторских задач, имеющих установленные алгоритмы решений. В настоящее время задача создания системы автоматического проектирования (САПР) для создателей одежды является достаточно актуальной, что видно из анализа развития и места на рынке программного обеспечения аналогичных разработок. Высокая точность конструирования разверток деталей одежды не может быть достигнута с помощью расчетно-графических методов, которые опираются только на антропологические исследования.

Таким образом, создание методик, алгоритмов и прикладных программ для выполнения этапов снятия размерных характеристик, конструкторской разработки градации и преобразование конструкций поясной одежды на различные типы телосложения фигур является актуальным направлением автоматизации проектирования одежды. Новизна данной работы заключается в возможности бесконтактного снятия размерных характеристик [4], и создании конструкции на различные типы телосложения подростковых фигур.

В настоящее время для любого производителя одежды компьютерное проектирование выгоднее ручного. Поэтому использование предложенного программного продукта снизит затраты времени на измерение фигуры, облегчит труд конструктора, ускорит процесс проектирования, повысит наглядность и эффективность работы.

#### Библиографический список

1. Кузнецова, Н. Д. Совершенствование системы конструирования одежды с использованием современных технических средств / Н. Д. Кузнецова. — М.: Наука, 1989. — 46 с.
2. Мичикова, Н. В. Методы бесконтактного измерения фигуры человека / Н. В. Мичикова, И. В. Лашина, В. Ю. Юрий // Региональные аспекты развития легкой промышленности в России: перспективы и конкурентоспособность. III Международный фестиваль «Формула моды»: научно-практическая конференция: сб. ст. / под общ. ред. ректора ОГИС проф. Н. У. Казачуна. — Омск: ОГИС, 2006. — 191 с.
3. Керниган, С. Брайан Язык программирования / С. Брайан Керниган, Деннис Ритчи. — Изд.: Вильямс, 2009. — 304 с. — ISBN 978-5-8459-0891-9, 5-8459-0891-4, 0-13-110362-8.

4. Керниган, С. Брайан. Язык программирования / С. Брайан Керниган, Деннис Ритчи; пер с англ. — 3-е изд., испр. — СПб.: Невский диалект, 2001. — 232 с.
5. Linux, статьи, обзоры, учебные пособия [Электронный ресурс]. — Режим доступа : <http://www.ibm.com/developerworks/ru/linux/>. (дата обращения : 22.12.2010).
6. Новые горизонты Linux [Электронный ресурс]. — Режим доступа : <http://www.ibm.com/ru/linuxcenter/articles/01.html>. (дата обращения : 22.12.2010).
7. Linux [Электронный ресурс]. — Режим доступа : <http://www.novell.com/linux/> (дата обращения : 28.12.2010).
8. Linux [Электронный ресурс]. — Режим доступа : [www.gnu.org/gnu/why-gnu-linux.html](http://www.gnu.org/gnu/why-gnu-linux.html). (дата обращения : 24.12.2010).

**ЖБАННИКОВА Мария Алексеевна**, аспирантка кафедры конструирования швейных изделий.  
**КОРОБОВА Антонина Брониславовна**, кандидат технических наук, профессор (Россия), декан художественно-технологического факультета.  
**БУРЦЕВ Александр Геннадьевич**, специалист по защите информации по специальности «Комплексное обеспечение информационной безопасности автоматизированных систем».  
 Адрес для переписки: e-mail: [mazyanya@gmail.com](mailto:mazyanya@gmail.com)

Статья поступила в редакцию 28.02.2011 г.

© М. А. Жбанникова, А. Б. Коробова, А. Г. Бурцев

## Информация

### Всероссийский конкурс «Ползуновские гранты»-2011

Министерство образования и науки Российской Федерации, Государственный Фонд содействия развитию малых форм предприятий в научно-технической сфере, ГОУ ВПО «Алтайский государственный технический университет им. И.И. Ползунова» сообщают о проведении Всероссийского конкурса докладов по совместной программе Министерства образования и науки Российской Федерации и Государственного фонда содействия развитию малых форм предприятий в научно-технической сфере «**Студенты, аспиранты и молодые ученые – малому наукоемкому бизнесу – Ползуновские гранты**» для привлечения авторов к процессу создания, разработки и реализации наукоемкого продукта или технологий непосредственно в субъекте малого предпринимательства на стадии обучения и работы в вузе, с проведением **Всероссийской научно-технической конференции** по его итогам.

Сроки проведения конкурса:

— прием заявок на участие: **до 15 сентября 2011 г.**

— проведение Всероссийской научно-технической конференции: сентябрь – октябрь 2011 г. (точная дата будет объявлена дополнительно)

Тематические направления конкурса «Ползуновские гранты»:

- биотехнология. Легкая, лесная и пищевая промышленность. Сельское хозяйство;
- добыча и переработка полезных ископаемых;
- информатика, вычислительная техника;
- машиностроение. Металлургия. Материаловедение;
- медицина и здравоохранение;
- охрана окружающей среды. Экология человека;
- приборы. Методы контроля. Управление качеством продуктов и услуг;
- строительство. Архитектура;
- транспортные, авиационные и космические системы;
- туристско-экскурсионное обслуживание;
- химическая технология. Химическая промышленность;
- электроника, автоматика, радиотехника, связь;
- энергетика, электротехника.

К участию приглашаются студенты, аспиранты и молодые ученые и преподаватели в возрасте до 28 лет в авторском коллективе от 1 до 5 человек. Участие в конкурсе предполагает их работу в сфере наукоемкого бизнеса субъектов малого инновационного предпринимательства.

Планируемый объем гранта – не менее 50 тыс. руб.

Адрес оргкомитета конкурса: 656038, г. Барнаул, пр. Ленина, 46, каб. 211.

Тел. (3852) 29-09-30 — Селезнева Лариса Евгеньевна.

e-mail: [polzunov-grant@list.ru](mailto:polzunov-grant@list.ru)

Подробная информация о конкурсе на сайте Алтайского государственного технического университета.

Источник : [http://www.rsci.ru/grants/grant\\_news/284/229814.php](http://www.rsci.ru/grants/grant_news/284/229814.php) (дата обращения : 15.06.11).