

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ УКРАИНЫ

ХАРЬКОВСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ
РАДИОЭЛЕКТРОНИКИ

ISSN 0555-2656

БИОНИКА
ИНТЕЛЛЕКТА
ИНФОРМАЦИЯ, ЯЗЫК, ИНТЕЛЛЕКТ

№ 2 (71)

2009

НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ

Основан в 1967 г.

Свидетельство о государственной регистрации КВ № 12072-943 ПР от 07.12.2006

Журнал включен в список специальных изданий ВАК Украины
по техническим наукам
(приложение к постановлению ВАК Украины № 1-05/7 от 04.07.2006)

СОДЕРЖАНИЕ

<i>Бондаренко М.Ф., Шабанов-Кушнарченко С.Ю., Шабанов-Кушнарченко Ю.П.</i> Практические приложения компараторной идентификации линейных конечномерных объектов.....	5
<i>Бондаренко М.Ф., Шабанов-Кушнарченко С.Ю., Шабанов-Кушнарченко Ю.П.</i> Разработка системы экономного кодирования цвета.....	13
<i>Бондаренко М.Ф., Шабанов-Кушнарченко С.Ю., Шабанов-Кушнарченко Ю.П.</i> Методы идентификации механизма субъективных социально-экономических оценок.....	24
<i>Вечірська І.Д.</i> Дослідження розмірності предметного простору в задачах моделювання об'єктів у вигляді реляційних мереж	31
<i>Хайрова Н.Ф.</i> Использование математического аппарата предикатных категорий для моделирования семантики сверхфразовых единств	36
<i>Остапова И.В.</i> Метод и технология представления этимологического словаря украинского языка в цифровой среде	40
<i>Дикарева С.С., Бержанский В.Н., Полулях С.Н., Потапова Е.В.</i> Инструментальная лексикографическая система в области физики магнитных явлений	48
<i>Заболотня Т.М.</i> Дослідження ефективності інверсного контекстно-асоціативного методу автоматизованої орфокодекції	54
<i>Шарий Т.В.</i> Об одном методе автоматической сегментации речевых сигналов	61
<i>Єрохін А.Л., Кольченко О.В.</i> Інтелектуальна система підтримки взаємодії з комп'ютером для людей з руховими порушеннями	66
<i>Емцев Е.Д., Лесная Н.С., Келеберда И.Н.</i> Исследование и построение игровой модели обучения на основе технологий социальных сетей.....	70
<i>Шатовская Т.Б., Репка В.Б., Каменева И.В., Марченко М.М.</i> Репозиторий интеллектуального анализа данных.....	75
<i>Купин О.Н., Лесная Н.С., Келеберда И.Н.</i> Использование метода учебных проектов на базе технологии социальных сетей	79
<i>Олейник А.А., Субботин С.А.</i> Гибридный мультиагентный подход для построения нейро-фаззи моделей в задачах диагностики.....	85
<i>Вишняк М.Ю., Довгань С.С.</i> Модель структурированного представления знаний о процедурах.....	93
<i>Клименко А.В.</i> Моделирование диффузии с фазовым переходом на сферической поверхности.....	99
<i>Олейник А.И.</i> Математическое моделирование диффузии нейротрансмиттера при фиксированных углах открытия везикулы	106
<i>Кротких С.С., Кириченко Л.О.</i> Определение структурной близости участков ЭЭГ сигнала для распознавания умственных задач, выполняемых человеком.....	111
<i>Субботин С.А.</i> Структурно-параметрическая идентификация нейро-нечетких моделей для диагностики	118
<i>Четвериков Г.Г., Королева Я.Ю., Бережная М.А.</i> Синтез проверяющих тестов для однородных схем.....	123
Об авторах	128
Правила оформления рукописей для авторов научно-технического журнала «Бионика интеллекта»	130



УДК 519.7



ПРАКТИЧЕСКИЕ ПРИЛОЖЕНИЯ КОМПАРАТОРНОЙ ИДЕНТИФИКАЦИИ ЛИНЕЙНЫХ КОНЕЧНОМЕРНЫХ ОБЪЕКТОВ

М.Ф. Бондаренко¹, С.Ю. Шабанов-Кушнаренко², Ю.П. Шабанов-Кушнаренко³

^{1, 2, 3} ХНУРЭ, г. Харьков, Украина

Хотя теория компараторной идентификации представляет собой вполне самостоятельную область знания, однако, она имеет много точек соприкосновения с теорией прямой идентификации. В связи с этим здесь рассматриваются те достижения классической теории идентификации, на которые пришлось опираться при разработке вопросов компараторной идентификации.

КОМПАРАТОРНАЯ ИДЕНТИФИКАЦИЯ, ЦВЕТОВОЕ ЗРЕНИЕ, ГИЛЬБЕРТОВО ПРОСТРАНСТВО

Введение

После того, как методами, описанными в [1, 2], установлено, что объект компараторной идентификации линеен и конечномерен, опытным путем должны быть определены параметры, характеризующие этот объект. В этой статье описываются два метода решения этой задачи.

Первый метод является компараторным вариантом метода импульсного возмущения, используемого при прямой идентификации линейного объекта [3].

Второй метод является компараторным вариантом метода ступенчатого возмущения, используемого при прямой идентификации линейного объекта [3].

В [4] был разработан метод структурной компараторной идентификации системы, реакции которой описываются метризирующим предикатом Φ . В этом разделе статьи рассматриваются два метода параметрической идентификации системы Φ , которые сводятся к построению сетки, характеризующей гомеоморфизм φ [4].

1. Разработка методов параметрической компараторной идентификации линейных конечномерных объектов

Рассмотрим первый метод. Параметрами линейного объекта, подлежащими определению, служат вещественные числа $c_{1j}, c_{2j}, \dots, c_{mj}$, являющиеся компонентами векторов k_j . Всего имеется n таких векторов: k_1, k_2, \dots, k_n [1]. Метод основан на использовании закона n -мерности [1]. На первом входе идентифицируемой системы подается сигнал в виде вектора $x_i = (0, 0, \dots, 0, 1, 0, \dots, 0)$, i -тым компонентом которого служит единица, остальные компоненты — нулевые. На второй вход системы подается смесь

$$\sum_{j=1}^n F_j(x_i) e_j \quad (1)$$

сигналов e_1, e_2, \dots, e_n . В роли e_1, e_2, \dots, e_n можно взять те же сигналы, которые были использованы при проверке закона n -мерности в процессе предшес-

твующей структурной компараторной идентификации объекта.

Пропорции $F_1(x_j), F_2(x_j), \dots, F_n(x_j)$ сигналов e_1, e_2, \dots, e_n в смеси (1) регулируются таким образом, чтобы идентифицируемая система E отреагировала на сигналы x_i и (1) единичным ответом, то есть чтобы выполнялось равенство

$$E(x_i, \sum_{j=1}^n F_j(x_i) e_j) = 1. \quad (2)$$

Согласно закону n -мерности набор коэффициентов $(F_1(x_j), F_2(x_j), \dots, F_n(x_j))$ всегда будет единственным. Из равенств (9)–(11) [4] непосредственно следует, что найденные в описанном опыте числа $F_j(x_j)$ совпадают с искомыми параметрами c_{ij} линейного объекта, то есть что

$$c_{ij} = F_j(x_j). \quad (3)$$

Подавая на первый вход идентифицируемой системы E по очереди сигналы x_1, x_2, \dots, x_m и определяя для каждого из них опытным путем коэффициенты F_1, F_2, \dots, F_n , можно найти всю матрицу

$$\begin{pmatrix} \chi_{11} \chi_{21} \dots \chi_{m1} \\ \chi_{12} \chi_{22} \dots \chi_{m2} \\ \dots \dots \dots \dots \\ \chi_{1n} \chi_{2n} \dots \chi_{mn} \end{pmatrix} \quad (4)$$

параметров идентифицируемого линейного конечномерного объекта.

Сигнал x_i ($i = \overline{1, m}$) можно образовать не только из единичного компонента, стоящего на i -том месте в координатном представлении вектора x_i при прочих нулевых его компонентах, но и из компонента a_i , имеющего произвольное ненулевое значение, так чтобы $x_i = (0, 0, \dots, 0, a_i, 0, \dots, 0)$. В этом случае параметры идентифицируемого объекта могут быть вычислены по формуле

$$c_{ij} = \frac{F_j(x_i)}{\alpha_i}, \quad (5)$$

вытекающей из соотношений (9)–(11) [4].

Сигналы e_1, e_2, \dots, e_n могут быть выбраны по следующей методике. Сигнал e_1 можно взять произ-

обеспечивает равенство расстояний $\varphi(a'_{11}, a'_{13})$, $\varphi(a'_{12}, a'_{13})$ и $\varphi(a'_{11}, a'_{12})$. Точек, удовлетворяющих этому условию, две. Выбор, показанный на рис. 1, б, соответствует сохранению ориентации вершин треугольника $a'_{11}a'_{12}a'_{13}$ при отображении φ^{-1} . Другой возможный выбор (точка a_{23} взамен a_{13} на рис. 1, б) означал бы применение зеркального отражения по отношению к первому варианту. Полагаем $\varphi^{-1}(a'_{13}) = a_{13}$. Дальнейшее построение узлов сетки на плоскости N однозначно. Например прообразом точки a'_{22} на рис. 1, а, определенной условиями $\varphi(a'_{22}, a'_{11}) = \varphi(a'_{22}, a'_{13}) = \varphi(a'_{11}, a'_{13})$, $a'_{22} \neq a'_{12}$, является точка a_{22} на рис. 1, б, определенная условиями $\Phi(a_{22}, a_{11}, a_{22}, a_{13}) = \Phi(a_{22}, a_{11}, a_{11}, a_{13}) = 1$, $a'_{22} \neq a'_{12}$. Аналогично строятся точки a_{21} и a_{23} . Затем можно перейти к построению третьего слоя точек $a_{31}, a_{32}, \dots, a_{36}$ и так далее.

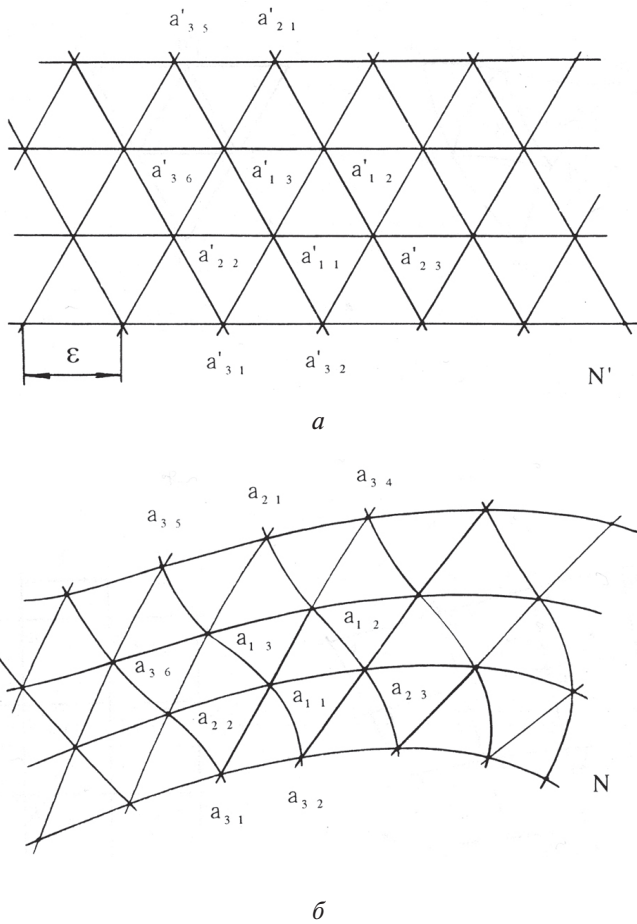


Рис. 1

После построения отображения φ^{-1} в узлах сетки может возникнуть вопрос о значении φ^{-1} в какой-либо точке x' , не являющейся узловой (рис. 2, а). Для ответа на этот вопрос можно пойти по двум путям. Один из них позволяет найти точку $\varphi^{-1}(x')$ с любой заданной точностью, но требует дополнительных экспериментов. Например пусть точка x' принадлежит треугольнику $a'_{11}a'_{12}a'_{13}$. Точками b'_1, b'_2, b'_3 разобьем этот треугольник на четыре рав-

носторонних треугольника со сторонами в два раза меньшими, чем исходный. В эксперименте находим точки $b_1 = a_{13} \circ a_{12}$, $b_2 = a_{11} \circ a_{13}$, $b_3 = a_{11} \circ a_{12}$. Они показаны на рис. 2, б. Поскольку

$$b'_1 = \frac{a'_{13} + a'_{12}}{2}, \quad b'_2 = \frac{a'_{11} + a'_{13}}{2}, \quad b'_3 = \frac{a'_{11} + a'_{12}}{2},$$

$\varphi^{-1}(b'_1) = b_1$, $\varphi^{-1}(b'_2) = b_2$, $\varphi^{-1}(b'_3) = b_3$. Точка x принадлежит одному из четырех треугольников $a_{11}b_2b_3$, $a_{13}b_1b_2$, $a_{12}b_1b_3$, $b_1b_2b_3$ (на рис. 2, б – треугольнику $b_1b_2b_3$). Если нас не удовлетворяет степень приближения точки x к ближайшей вершине своего треугольника (в нашем примере – это точка b_1), то процесс дробления треугольника следует продолжать.

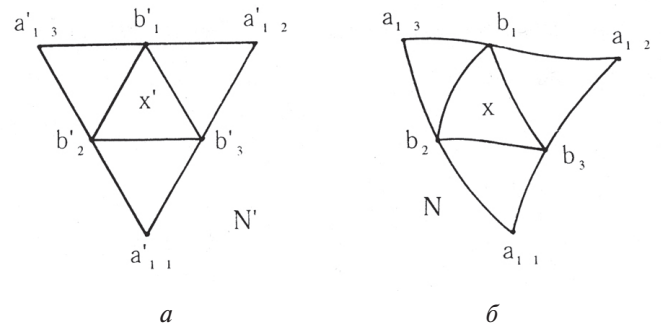


Рис. 2

Если же возникает вопрос о приближенном нахождении значения $\varphi^{-1}(x')$ без проведения дополнительных экспериментов, то можно применить какую-либо интерполяцию. Простейший способ состоит в следующем. Пусть $x' = g_1 a'_{11} + g_2 a'_{12} + g_3 a'_{13}$, $g_i \geq 0$, $g_1 + g_2 + g_3 = 1$ – барицентрическое [5] представление точки x' . Тогда в качестве приближения для значения $\varphi^{-1}(x') = x$ можно взять точку $x = g_1 a_{11} + g_2 a_{12} + g_3 a_{13}$.

Отметим, что предложенный способ приближенного построения отображения φ неприменим при $n=2$. Дело в том, что пространство R^n при $n=2$ не может быть вымощено правильными тетраэдрами. Укажем другой способ, лишенный этого недостатка, но являющийся более громоздким в случае $n=2$. Обобщение на случай более высокой размерности пространства N' не составляет труда. Заддим произвольное $\epsilon > 0$. Построим на плоскости сетку из правильных квадратов со стороной ϵ (см. рис. 3, а). Выберем какой-либо из этих квадратов, например, $a'_{01}a'_{01}a'_{11}a'_{01}$. Далее зафиксируем на плоскости N' произвольную точку a_{00} (рис. 3, б). Положим $\varphi^{-1}(a'_{00}) = a_{00}$. Затем зафиксируем произвольную точку $a_{10} \neq a_{00}$ и положим $\varphi^{-1}(a'_{10}) = a_{10}$. Пусть a_{-10} – точка, определенная условием $a_{-10} \circ a_{10} = a_{00}$. Положим $\varphi^{-1}(a'_{-10}) = a_{-10}$. Аналогично положим $\varphi^{-1}(a'_{20}) = a_{20}$, где точка a_{20} определена условием $a_{20} \circ a_{00} = a_{10}$.

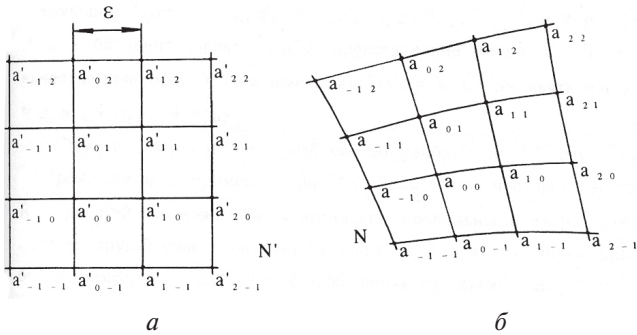


Рис. 3

С помощью этой процедуры может быть построена любая из точек $\varphi^{-1}(a'_{k_0})$, $k = 0, \pm 1, \pm 2, \dots$. Построим теперь точку a_{01} , удовлетворяющую условиям

$$\Phi(a_{01}, a_{00}, a_{00}, a_{10}) = \Phi(a_{-10}, a_{01}, a_{10}, a_{01}). \quad (11)$$

3. Построение метризирующего отображения для цветового пространства

Разработанный в предыдущем разделе метод использования сетки с квадратными ячейками для компараторной идентификации метризирующего отображения теперь применим для изучения метрики цветового пространства. Состояние исследований в этой области было проанализировано в [3]. Чтобы наметить путь для решения указанной задачи, воспользуемся аналогией, имеющей место между механизмом восприятия цвета и механизмом зрительного восприятия пространства.

Руководствуясь схемой зрительного восприятия физического пространства, представленной в [6], приходим к следующей модели преобразования объективного поля зрения N в субъективное N' . В качестве исходных постулатов используются соотношения

$$\gamma = \beta, \quad (12)$$

$$\rho = k\alpha, \quad (13)$$

где k – коэффициент пропорциональности. Равенство (12) означает, что любая прямая, проходящая через начало координат e на плоскости N , преобразуется снова в прямую на плоскости N' , причем углы их наклона γ и β всегда совпадают. Соотношение (13) означает, что на плоскости N' расстояние точки x' от начала координат e' всегда пропорционально углу α , под которым наблюдатель видит точку X физического пространства M .

Постулаты (12) и (13) достаточно точно согласуются с фактами зрения, которые были изображены Гельмгольцем [7] в виде так называемой «шахматной доски», изображенной на рис. 4. При рассмотрении рисунка одним глазом с расстояния 20 см в семикратном увеличении он воспринимается не с искажениями, а с правильными квадратами как шахматная доска.

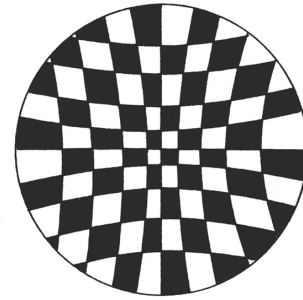


Рис. 4

Гельмгольц объясняет это явление тем, что наше поле зрения является для нас вовсе не плоскостью, но что мы видим предметы как бы прикрепленными к вогнутой, приблизительно шарообразной поверхности. Такое объяснение, очевидно, согласуется с постулатами (12) и (13).

Из постулатов (12) и (13) выведены формулы

$$\xi' = \frac{k \cdot \operatorname{arctg} \frac{\sqrt{\xi^2 + \eta^2}}{R}}{\sqrt{\xi^2 + \eta^2}} \cdot \xi, \quad (14)$$

$$\eta' = \frac{k \cdot \operatorname{arctg} \frac{\sqrt{\xi^2 + \eta^2}}{R}}{\sqrt{\xi^2 + \eta^2}} \cdot \eta, \quad (15)$$

описывающие преобразование φ объективного поля зрения N в субъективное N' .

Формулы

$$\xi = \frac{R \cdot \operatorname{tg} \frac{\sqrt{\xi'^2 + \eta'^2}}{k}}{\sqrt{\xi'^2 + \eta'^2}} \cdot \xi', \quad (16)$$

$$\eta = \frac{R \cdot \operatorname{tg} \frac{\sqrt{\xi'^2 + \eta'^2}}{k}}{\sqrt{\xi'^2 + \eta'^2}} \cdot \eta' \quad (17)$$

описывают обратное преобразование φ^{-1} субъективного поля зрения N' в объективное N .

Существует глубокая аналогия между механизмом цветового зрения и механизмом зрительного восприятия пространства. Давно подмечено, что природа часто использует одни и те же математические зависимости при реализации механизмов, предназначенных для совершенно различных целей. Цветовое пространство трехмерно, также трехмерно и физическое пространство, воспринимаемое глазом человека. Лучам, исходящим из оптического центра глаза, соответствуют лучи в цветовом пространстве, исходящие из точки черного цвета. Все точки одного и того же луча физического пространства проектируются в единственную точку поля зрения. Точно так же все точки каждого луча цветового пространства проектируются в одну точку цветового графика. Отдельной точке поля зрения соответствует точка цветового графика, за-

дающая вполне определенную цветность светового излучения. Кампиметрической плоскости, то есть плоскости N физического пространства, перпендикулярной зрительной оси, соответствует в цветовом пространстве плоскость равной яркости. Расстоянию R кампиметрической плоскости от оптического центра глаза соответствует в цветовом пространстве яркость светового излучения. Объективному полю зрения соответствует объективное цветовое пространство, субъективному полю зрения – субъективное цветовое пространство.

Руководствуясь вышеприведенной аналогией, мы попытаемся распространить действие зависимостей (12)-(17), описывающих преобразование объективного поля зрения в субъективное, на механизм цветового зрения и таким образом достичь математического описания метризирующего отображения объективного цветового пространства в субъективное. Если аналогия подтвердится, то для достижения полного математического описания отображения, метризирующего цветовое пространство, нам предстоит решить следующие задачи: 1) найти в цветовом пространстве луч, соответствующий зрительной оси, а на цветовом графике – его отпечаток, то есть точку, соответствующую для поля зрения точке фиксации; 2) найти на цветовом графике положение координатных осей, соответствующих осям x_1 и x_2 на кампиметрической плоскости; 3) найти масштаб для шахматной доски Гельмгольца, определяемый величиной параметра k , связывающего зависимость (12) величины α и ρ .

При решении перечисленных задач мы исходили из равноконтрастного цветового графика (рис. 5), построенного Н.Г. Болдыревым и К.Б. Мартыновым [8] на основании опытов Д. Мак-Адама [9] по определению пороговых эллипсов в различных точках цветового графика (рис. 6).

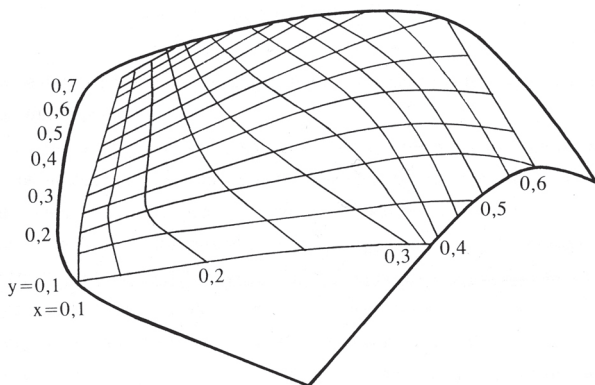


Рис. 5

Переход от рис. 6 к рис. 5 произведен с помощью такой деформации цветового графика, при которой все эллипсы превращаются в окружности одинакового диаметра. График, изображенный на рис. 5, играет роль аналога субъективного поля зрения. Его мы назовем субъективным цветовым

графиком. Цветовой график, который бы играл роль аналога объективного поля зрения, назовем объективным цветовым графиком.

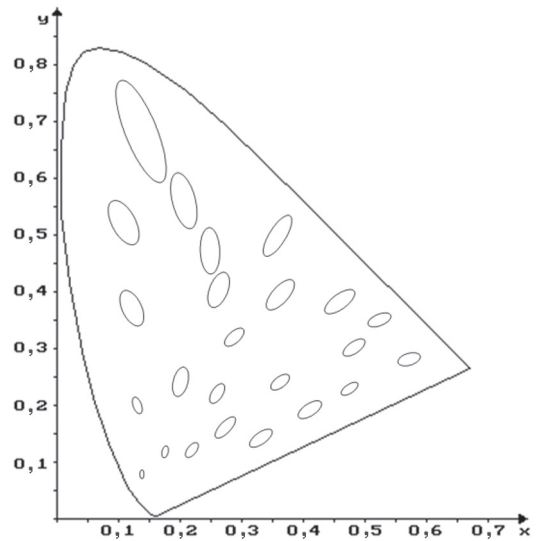


Рис. 6

Нанесем на субъективном цветовом графике прямоугольную сетку, размер ее ячеек и ориентацию сетки выбираем произвольно. Пример такой сетки представлен на рис. 7.

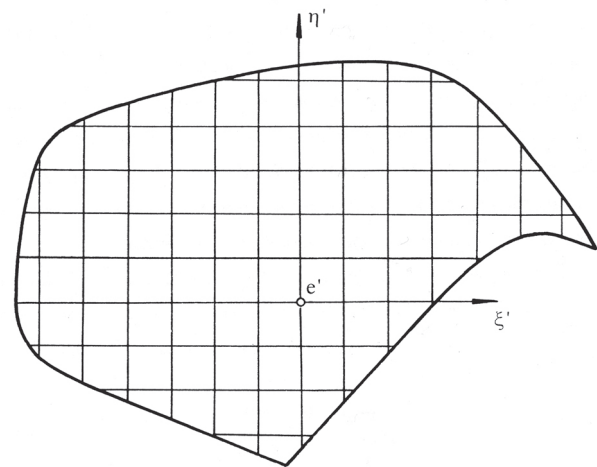


Рис. 7

Как видно из рис. 4, все прямые линии сетки при проекции последней на объективный цветовой график должны превратиться в кривые, кроме двух линий, остающихся прямыми и ортогональными друг другу. Проектируя сетку на цветовой график рис. 6, обнаруживаем, что действительно все прямые линии преобразуются в кривые, кроме двух линий, которые остаются практически прямыми при любой ориентации сетки. Эти прямые пересекаются в точке $x = 0,270$; $y = 0,130$. Оказывается, что точка пересечения прямых почти не зависит от ориентации исходной сетки. Таким образом, мы нашли на цветовом графике начало координатной системы, которое соответствует точке фиксации в поле зрения.

Тем не менее, прямые на графике рис. 6 пересекаются друг с другом при различных ориентациях сетки не всегда под прямым углом. Это означает, что плоскость цветового графика, изображенного на рис. 6, нельзя принять за аналог кампиметрической плоскости, а значит, эту диаграмму нельзя рассматривать как объективный цветовой график. Мы попытались найти в цветовом пространстве такое положение плоскости цветового графика, чтобы при произвольной ориентации сетки на субъективном цветовом графике ее оси всегда превращались после перехода к объективному цветовому графику в пару ортогональных прямых. Такая плоскость существует, и она практически совпадает с плоскостью равной яркости, а объективный цветовой график близок к диаграмме, указанной Д. Джаддом [10]. Этот график представлен на рис. 8.

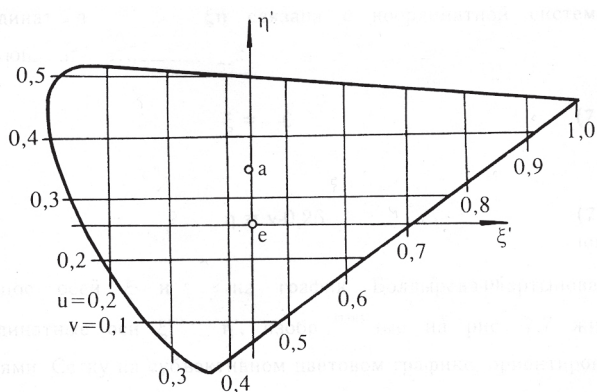


Рис. 8

Преобразование стандартного цветового графика (рис. 6) в цветовой график Джадда описывается уравнениями

$$u = \frac{0,4661x + 0,1593y}{y - 0,1574x + 0,2424}, \quad (18)$$

$$v = \frac{0,6581y}{y - 0,1574x + 0,2424}. \quad (19)$$

Направление осей ξ и η на объективном цветовом графике можно выбрать произвольным, но таким, чтобы оси были взаимно перпендикулярными и проходили через найденное выше начало координатной системы. Удобно взять оси, совпадающие по направлению с осями u и v . Соответствующая координатная сетка показана на рис. 9, оси ξ и η на ней проведены жирными линиями. Координатная система $\xi\eta$ связана с координатной системой uv следующими соотношениями:

$$\xi = u - 0,44, \quad (20)$$

$$\eta = v - 0,26. \quad (21)$$

Перенос осей ξ и η на график Болдырева-Мартынова дает координатные оси ξ' и η' , изображенные на рис. 7 жирными линиями.

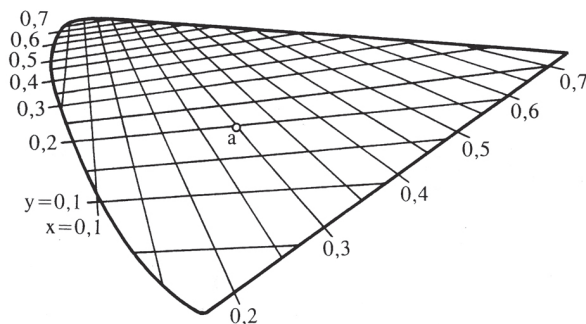


Рис. 9

Сетку на субъективном цветовом графике, ориентированную в соответствии с найденными координатными осями (см. рис. 7), проектируем на диаграмму Джадда. В результате получаем диаграмму, изображенную на рис. 10.

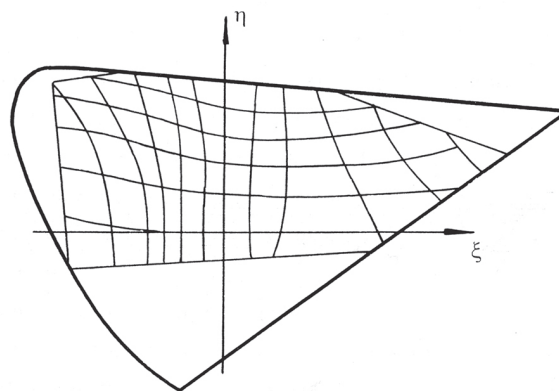


Рис. 10

На диаграмму Джадда также наносим деформированную шахматную доску Гельмгольца (рис. 11), масштаб которой подбирается с таким расчетом, чтобы она по характеру кривых наиболее приблизилась к кривым рис. 10. Кривые обеих диаграмм по своему направлению довольно хорошо согласуются друг с другом, за исключением четырех вертикальных линий в верхнем правом квадранте графика.

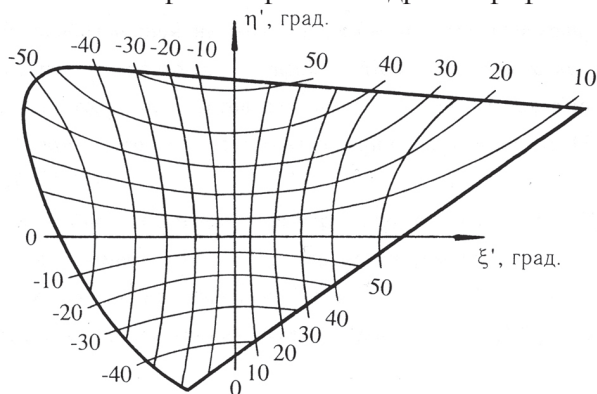


Рис. 11

Однако эти четыре кривые также противоречат и исходным экспериментальным данным Мак-Адама (см. ориентацию эллипсов в правой верхней части рис. 6), которые, однако, хорошо согласуются с диаграммой рис. 11.

Переход от координатной системы $\xi\eta$ к координатной системе $\xi'\eta'$ математически описывается формулами (14) и (15) при $k=1$ и

$$R = 0,170 \text{ 1/град.} \quad (22)$$

Таким образом, общие формулы (12)-(15) для случая цветового зрения человека запишутся в виде:

$$\xi' = \frac{5,9 \cdot \arctg \sqrt{\xi^2 + \eta^2}}{\sqrt{\xi^2 + \eta^2}} \cdot \xi, \quad (23)$$

$$\eta' = \frac{5,9 \cdot \arctg \sqrt{\xi^2 + \eta^2}}{\sqrt{\xi^2 + \eta^2}} \cdot \eta, \quad (24)$$

$$\xi = \frac{0,170 \cdot \tg \sqrt{\xi'^2 + \eta'^2}}{k} \cdot \xi', \quad (25)$$

$$\eta = \frac{0,170 \cdot \tg \sqrt{\xi'^2 + \eta'^2}}{k} \cdot \eta'. \quad (26)$$

Выравнивая координатную сетку $\xi\eta$ цветового графика, изображенного на рис. 11, получаем скорректированную диаграмму субъективного цветового графика (рис. 12).

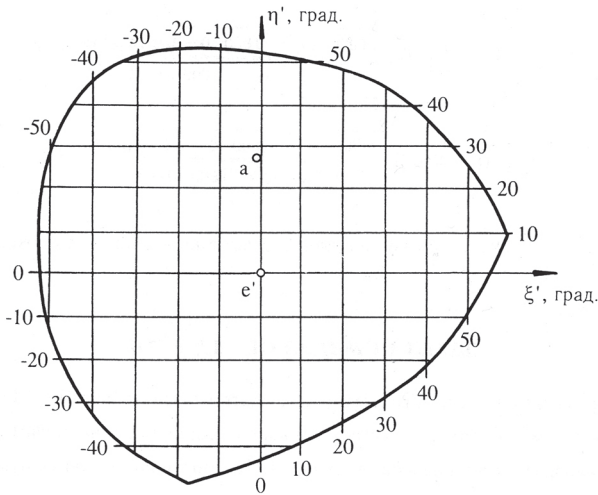


Рис. 12

Как видим, границы цветового графика приобрели форму, близкую к окружности, а начало координатной системы $\xi\eta$ расположилось почти в ее центре. В этом факте проявляется еще одна аналогия между цветовым и пространственным зрительным восприятием, поскольку поле зрения человека также имеет форму, близкую к круговой, точка фиксации располагается приблизительно в его центре. Заранее эта аналогия не имелась в виду и явилась для автора приятной неожиданностью, она проявилась только в конце длинной цепи рассуждений и вычислений.

На рис. 13 на субъективный цветовой график нанесена координатная сетка xu . Пользуясь этой диаграммой совместно с диаграммой, изображенной на рис. 12, можно легко переходить от точек стандартного цветового графика МКО к соответствующим точкам равноконтрастного цветового графика. Тот же переход можно совершать, кроме того, и путем вычислений по формулам (18)-(26).

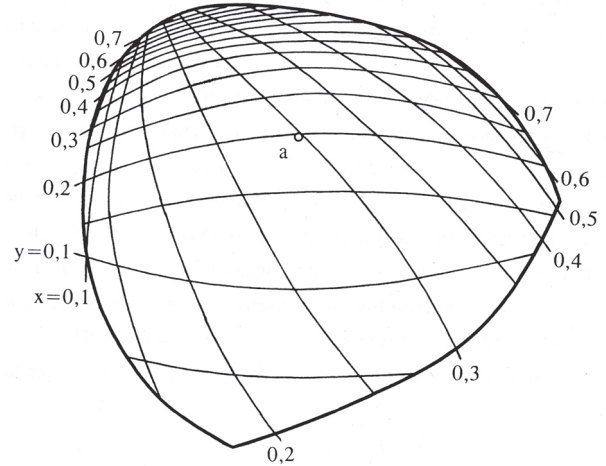


Рис. 13

В заключение приведем пример такого перехода. Пусть некоторый цвет имеет цветность, характеризуемую координатами $x=0,3$; $y=0,2$ (см. точку a на рис. 8 и на рис. 13). Вычисляем координаты u и v для той же цветности по формулам (18) и (19):

$$u = \frac{0,4661 \cdot 0,3 + 0,1593 \cdot 0,2}{0,2 - 0,1574 \cdot 0,3 + 0,2424} = 0,437,$$

$$v = \frac{0,6581 \cdot 0,2}{0,2 - 0,1574 \cdot 0,3 + 0,2424} = 0,349.$$

Положение точки a в системе uv отмечено на рис. 9. По формулам (20) и (22) вычисляем координаты точки a в системе $\xi\eta$:

$$\xi = 0,437 - 0,440 = -0,003,$$

$$\eta = 0,349 - 0,260 = 0,089.$$

По формулам (24) и (25) вычисляем координаты точки a в системе $\xi\eta$:

$$\xi' = \frac{\arctg 5,9 \sqrt{(-0,003)^2 + (0,089)^2}}{\sqrt{(-0,003)^2 + (0,089)^2}} (-0,003) = -1^\circ,$$

$$\eta' = \frac{\arctg 5,9 \sqrt{(-0,003)^2 + (0,089)^2}}{\sqrt{(-0,003)^2 + (0,089)^2}} 0,089 = 27^\circ.$$

Положение точки a в системе $\xi\eta$ показано на рис. 11 и рис. 12.

Выводы

Разработаны методы параметрической компараторной идентификации линейных конечномерных объектов — импульсного возмущения,

ступенчатого возмущения и обобщенного ступенчатого возмущения. Разработаны методы построения сетки для метризирующего отображения, в которых используются треугольные и четырехугольные ячейки. Методы компараторной идентификации отображения, метризирующего пространство выходных сигналов линейного конечномерного объекта, применены для построения равноконтрастного цветового графика. Получено уточнение равноконтрастного цветового графика, используемого в настоящее время в качестве международного стандарта в области колориметрии и цветного телевидения. Получены расчетные зависимости для перехода от стандартного цветового графика к уточненному и обратно.

Список литературы: 1. Бондаренко, М.Ф. Об общей теории компараторной идентификации [Текст] / М.Ф. Бондаренко, С.Ю. Шабанов-Кушнаренко, Ю.П. Шабанов-Кушнаренко // Бионика интеллекта: науч.-техн. журнал. – 2008. № 2(68). – С. 13-22. 2. Бондаренко, М.Ф. Идентификация объектов, описываемых числовыми системами [Текст] / М.Ф. Бондаренко, С.Ю. Шабанов-Кушнаренко, Ю.П. Шабанов-Кушнаренко // Бионика интеллекта: науч.-техн. журнал. – 2008. – № 2(68). – С. 23-31. 3. Бондаренко, М.Ф. Компараторная идентификация цветового зрения человека [Текст] / М.Ф. Бондаренко, С.Ю. Шабанов-Кушнаренко, Ю.П. Шабанов-Кушнаренко // Бионика интеллекта: науч.-техн. журнал. – 2008. – № 2(68). – С. 3-12. 4. Бондаренко, М.Ф. Идентификация объектов, описываемых векторами и линейными операциями над ними [Текст] / М.Ф. Бондаренко, С.Ю. Шабанов-Кушнаренко, Ю.П. Шабанов-Кушнаренко // Бионика интеллекта: науч.-техн. журнал. – 2009. – № 1(70). – С. 3-10. 5. Ефимов, Н.В. Высшая геометрия [Текст] / Н.В. Ефимов. – М.: Физматгиз, 1961. – 580 с. 6. Бондаренко, М.Ф. Разработка теории структурной метризации конечномерного арифметического пространства [Текст] /

М.Ф. Бондаренко, С.Ю. Шабанов-Кушнаренко, Ю.П. Шабанов-Кушнаренко // Бионика интеллекта: науч.-техн. журнал. – 2009. – № 1(70). – С. 17-27. 7. Гебер, Р. Курс физиологии человека [Текст] / Р. Гебер. – М.-Л.: Биомедгиз, 1936. – 678 с. 8. Болдырев, Н.Г. Равноконтрастный цветовой график [Текст] / Н.Г. Болдырев, К.Б. Мартынов // Проблемы физиологической оптики. – 1953. – Т. VIII. – С. 54-63. 9. Mac-Adam, D.L. Representations of colour tolerance on the chromaticity diagram [Текст] / D.L. Mac-Adam // JOSA. – 1939. – Vol. 4. – P. 46. 10. Judd, D. A Maxwell triangle yielding uniform chromaticity diagram [Текст] / D. Judd // JOSA. – 1935. – Vol. 25. – P. 24.

Поступила в редколлегию 03.09.2009

УДК 519.7

Практичні додатки компараторної ідентифікації лінійних скінченномірних об'єктів / М.Ф. Бондаренко, С.Ю. Шабанов-Кушнаренко, Ю.П. Шабанов-Кушнаренко // Біоніка інтелекту: наук.-техн. журнал. – 2009. – № 2 (71). – С. 5–12.

У статті розвиваються методи параметричної компараторної ідентифікації лінійних скінченномірних об'єктів. Отримані методи компараторної ідентифікації метризуючого відображення застосовано для побудови аналітичної моделі рівноконтрастного кольорового графіка.

Лл. 13. Бібліогр.: 10 найм.

UDC 519.7

The linear final-dimensional objects comparator identifications applications / M.F. Bondarenko, S.Yu. Shabanov-Kushnarenko, Yu.P. Shabanov-Kushnarenko // Bionika intellekta: Sci. Mag. – 2009. – № 2(71). – P. 5-12.

In article is develops the linear final-dimensional objects parametrical comparator identifications methods. The obtained comparator identifications methods of metric imagery are applied to construction of the colour schedule equal-contrast analytical model.

Fig.: 13. Ref.: 10 items.

УДК 519.7

**РАЗРАБОТКА СИСТЕМЫ ЭКОНОМНОГО КОДИРОВАНИЯ ЦВЕТА**М.Ф. Бондаренко¹, С.Ю. Шабанов-Кушнаренко², Ю.П. Шабанов-Кушнаренко³^{1, 2, 3} ХНУРЭ, г. Харьков, Украина

В статье сформулирована и решена задача о построении системы экономного кодирования цвета. На базе метрической теории конечномерного пространства разработана математическая модель равноконтрастного цветового пространства. Найдены параметры взаимно однозначного и взаимно непрерывного преобразования колориметрического пространства цветов в равноконтрастное цветовое пространство.

КОЛОРИМЕТРИЧЕСКОЕ ПРОСТРАНСТВО, ЦВЕТОВОЙ ПОРОГ, ЦВЕТОВОЕ ЗРЕНИЕ, КОЛОРИМЕТР, РАВНОКОНТРАСТНЫЙ АТЛАС ЦВЕТОВ

Введение

Классическая линейная теория цветового зрения [1] указывает способ кодирования цветов, который в настоящее время широко используется на практике [2]. Этим способом каждый цвет формально представляют в виде вектора $a = (\alpha_1, \alpha_2, \alpha_3)$ трехмерного арифметического пространства [3], называемого колориметрическим. Координаты цвета $\alpha_1, \alpha_2, \alpha_3$ линейно зависят от спектра породившего его светового излучения. Указанный способ кодирования цвета, несмотря на свою простоту и естественность, имеет существенный недостаток, заключающийся в его неэкономности. Это обнаруживается в том, что психологическое расстояние между цветами, измеряемое числом порогов между ними, не всегда соответствует евклидову расстоянию между концами векторов этих цветов.

Практически это приводит к тому, что одни цвета приходится кодировать с избыточным числом знаков, а другие – с недостаточным [4]. Избыточное число знаков в коде цвета вызывает перерасход объема памяти ЭВМ при запоминании цветных изображений, недостаточное же число ведет к неточной передаче цвета при воспроизведении изображения [5]. Задача об экономном кодировании цвета будет решена, если удастся указать такое взаимно однозначное и взаимно непрерывное преобразование колориметрического пространства цветов в равноконтрастное цветовое пространство, в котором расстояние между его векторами измеряется числом разделяющих их цветовых порогов. До настоящего времени задача об экономном кодировании цвета удовлетворительным образом еще не решена [6, 7].

1. Создание колориметра для измерения цветовых порогов на базе ПК

Проблема экономного кодирования цвета упирается в недостаточную изученность цветовых порогов и отсутствие адекватной математической модели равноконтрастного цветового пространства. Ниже описывается предпринятая в работе попытка решения данной задачи на базе изложенной ранее метрической теории конечномерного простран-

ства. Недостаточная изученность цветовых порогов обусловлена чрезвычайной трудоемкостью выполнения необходимых экспериментов. Для определения цветовых порогов до настоящего времени используются исключительно оптические колориметры, которые имеют весьма сложную конструкцию и большое число регулировок. На выполнение подобных экспериментов исследователь тратит годы упорного и тяжелого труда, получая в результате весьма фрагментарные результаты. Так, например, в опытах Мак-Адама [8, 9], выполненных в 1937 – 1943 гг., которые считаются до сих пор непревзойденными, обследовано всего 25 пороговых зон. Вместе с тем в цветовом пространстве имеется много миллионов пороговых зон. Опыты Мак-Адама выполнялись при фиксированном фоне, хотя очевидно, что в реальных условиях фон, на котором человек видит цвета, меняется. Это вызвано, кроме усложнения эксперимента, еще и тем, что в существующих оптических колориметрах необходимая для выполнения пороговых опытов регулировка фона не предусмотрена.

Представляется, что выход из создавшегося затруднительного положения может быть найден благодаря использованию принципиально новой техники для проведения экспериментов. А именно, современные ПК дают прекрасную возможность создания высокоточного и высокопроизводительного колориметра для изучения цветовых порогов. Изучение состояния вопроса показывает, что такая задача до сих пор никем не решалась и даже не ставилась. Основная идея создания порогового колориметра на базе ПК заключается в следующем. Оптические колориметры всегда создавались для изучения зависимости цвета от светового излучения. Для решения этой задачи использование оптики неизбежно, поскольку световое излучение невозможно исключить из такого эксперимента. Когда же изучаются цветовые пороги, то анализируется лишь связь между цветами, поэтому световые излучения, их породившие, можно и не вводить. Именно поэтому опыты по изучению цветовых порогов можно проводить без оптической техники. Следовательно, формирование цве-

тов на экране ПК является достаточным средством для осуществления пороговых экспериментов. Использование оптических колориметров для изучения цветовых порогов в прежние времена было вынужденным, поскольку отсутствовали другие средства для формирования цвета. В настоящее же время в этом нет необходимости, поскольку цвета можно формировать на экране ПК.

Имеется несколько обстоятельств, препятствующих непосредственному использованию ПК в качестве колориметра для измерения цветовых порогов. Во-первых, цвета, формируемые на экране ПК, маркируются не в той системе, которая общепринята в колориметрии. А пока не будет осуществлена привязка цветов, формируемых ПК, к колориметрической системе, исследователь, производящий на ЭВМ опыты с цветами, не сможет полученные результаты применить для построения равноконтрастного цветового пространства. Во-вторых, количество цветов, которые можно воспроизвести на экране ПК, недостаточно для достижения требуемой точности измерения цветовых порогов. В-третьих, в настоящее время отсутствуют сервисные программы, с помощью которых экспериментатор мог бы беспрепятственно производить на ПК все необходимые эксперименты и связанные с ними расчеты. Именно решению этих задач и посвящена нижеследующая часть этой статьи.

Вопрос об определении колориметрических координат цвета машинных цветов решается на базе использования пространственного закона Талбота [10]. Известно, что мозаика, составленная из достаточно мелких цветных точек, субъективно сливается, и в результате наблюдатель видит непрерывное и однородное цветное изображение. На этом принципе основано действие цветного телевизора. При слиянии цветных точек глаз производит точное линейное усреднение цвета, измеренного в линейной колориметрической системе [11]. В этом и состоит сущность закона Талбота. Разработана программа, которая рассчитывает распределение точек на экране для произвольного заранее заданного цвета. Линейный размер пикселя составляет обычно от 0,2 до 0,3 мм. По каждому из основных цветов (красный, зеленый и синий) предусмотрено 256 уровней интенсивности (от 0 до 255).

Принцип действия программы следующий. Отсутствие цвета на экране дисплея — это черный цвет с колориметрическими координатами (0, 0, 0). Основным красным цветом максимальной интенсивности (маркированный числом 255) принимаем за вектор (1, 0, 0) колориметрического пространства. Аналогичный зеленый цвет принимаем за вектор (0, 1, 0), синий — за вектор (0, 0, 1). Мозаику строим из однотипных блоков, каждый из которых имеет размер $15 \times 15 = 225$ точек. Такого числа точек в мо-

заике достаточно для регулирования интенсивности основных цветов с приемлемой точностью, составляющей $100/225 = 0,45\%$. При этой точности ошибка измерения, как показывает опыт, никогда не выходит за пределы цветового порога.

Внутри блока используются точки двух цветов: черного и одного из основных цветов (красного, зеленого или синего) максимальной интенсивности (255-я градация). Если требуется получить основной цвет (например, красный) с колориметрической интенсивностью α , то $\alpha \times 225$ точек в блоке берутся красными (255-я градация), остальные $(1 - \alpha) \times 225$ точек — черными. Программа построена так, что распределяет красные точки равномерно по полю блока. Первое поле сравнения имеет размер около 40 мм (в зависимости от размера пикселя), оно состоит из 10×10 блоков и размещается в центре экрана. Второе поле сравнения окружает первое и образовано остальной площадью экрана. На нем формируется основной цвет произвольной интенсивности.

Задача определения колориметрических координат данной компьютерной интенсивности основного цвета состоит в том, чтобы подобрать такое значение колориметрической интенсивности α (назовем ее “мозаичной”) того же основного цвета, при которой цвет центрального мозаичного поля визуально сравняется с цветом окружающего его поля. Опыты, выполненные описанным методом, показали, что мозаичная интенсивность α каждого основного цвета в пределах точности эксперимента оказывается строго пропорциональной квадрату компьютерной интенсивности x того же цвета. Связь между ними описывается зависимостью

$$\alpha = x^2 / 225^2. \quad (1)$$

Здесь x — машинная координата основного цвета (от 0 до 255), α — мозаичная координата того же основного цвета (от 0 до 1). Пересчет машинных координат цвета в мозаичные и обратно производится программно согласно формуле (1).

Второе препятствие на пути создания компьютерного колориметра, как уже говорилось, заключается в том, что число градаций цвета, которое можно сформировать на экране ПК, неизмеримо меньше того, которое необходимо для достаточно точного измерения цветовых порогов. Так, по каждому основному цвету (красный, зеленый, синий) предусмотрено всего лишь 255 градаций, в то время как требуются многие тысячи.

Для эффективного измерения цветовых порогов необходимо, чтобы дозирование цвета велось, по крайней мере, с точностью до 5% пороговой величины. Требуемое увеличение числа градаций цвета, формируемого на экране компьютера, достигается методом цветовой мозаики, описанным

выше. Тест-поле формируется из блоков, образующих растр 10×10 цветных точек. По заданным колориметрическим координатам цвета, который требуется точно воспроизвести, программно находятся ближайшие к нему 8 грубо дозируемых (255 градаций) компьютерных цветов. В колориметрическом пространстве эти 8 цветов образуют вершины шестигранника, охватывающего точку заданного цвета. Программа, кроме того, находит нужное число точек раstra 10×10 для каждого из 8 грубо дозируемых цветов и распределяет эти точки равномерно по площади блока. Таким способом можно получить до 25500 градаций интенсивности по каждому из основных цветов, что обеспечивает 2-3%-ю точность измерения порогов.

2. Разработка методики исследования цветowych порогов

Создание компьютерного колориметра, описанного в предыдущем параграфе, позволило осуществить необходимые опыты по определению цветowych порогов. Прежде чем изложить методику описываемых ниже экспериментов, рассмотрим достижения в этой области, содержащиеся в опубликованных литературных источниках. Ценными исследованиями, в которых цветowych пороги были подвергнуты систематическому изучению, являются работы Райта [12, 13]. Эксперименты проводились на полях сравнения с угловым размером 2° при черном фоне, причем сравниваемые цвета предварительно выравнивались к одной и той же яркости. Опыты были выполнены на оптическом колориметре. Их результаты представлены на рис. 1. Пороги в виде отрезков линий нанесены на цветовой график МКО 1931 г. По указанным направлениям для цветностей, представленных серединами отрезков линий было измерено всего 35 цветowych порогов. Как видно из графика, пороги резко различаются друг от друга по величине.

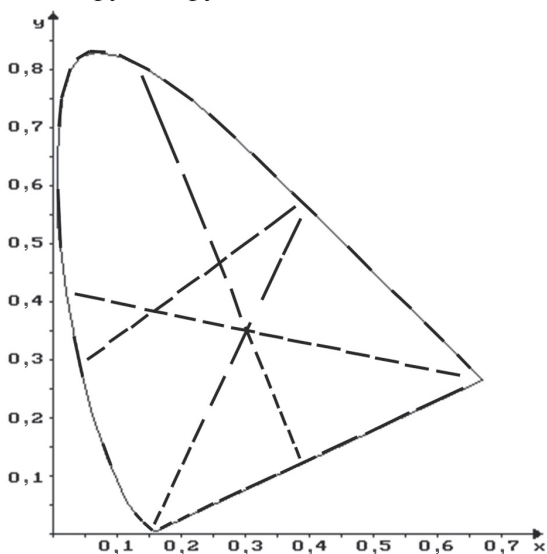


Рис. 1

Отношение максимального порога к минимальному более 9.

Обширные исследования цветowych порогов на одном наблюдателе осуществил Мак-Адам [8, 9, 14]. Они выполнялись на оптическом колориметре для 25 цветов одной и той же яркости ($48 \text{ кд} \cdot \text{м}^{-2}$) на сером фоне половинной яркости, имеющем цветность стандартного источника С. Для каждого цвета измерения выполнялись по 16-ти направлениям в плоскости равной яркости, причем каждое измерение повторялось по 50 раз. В результате усреднения замеров и последующей интерполяции получены пороговые эллипсы, нанесенные на график МКО 1931 года (рис. 2).

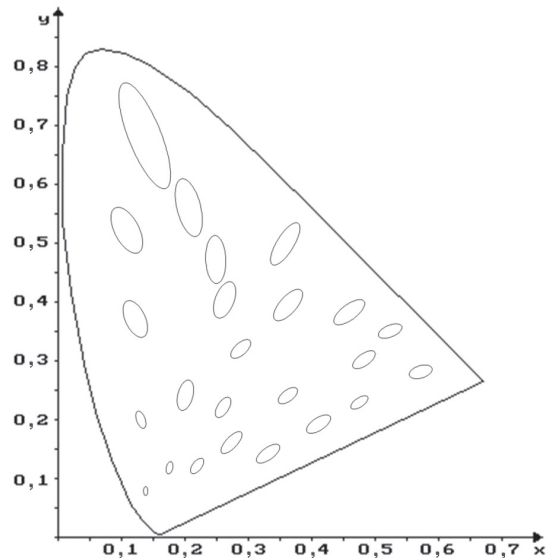


Рис. 2

Отношение максимального порога к минимальному составляет величину около 25. Точность интерполяции ансамбля пороговых точек эллипсом иллюстрируется рис. 3, на котором изображен в увеличенном масштабе верхний эллипс, взятый из рис. 2. Здесь же нанесены пороговые точки, для которых производилась интерполяция. Разброс цветowych порогов, отнесенный к соответствующему радиусу эллипса, достигает 50%. По тщательности и стабильности измерений, а также по объему экспериментального материала опыты Мак-Адама до настоящего времени остаются непревзойденными.



Рис. 3

Более поздние исследования Брауна и Мак-Адама [15] были осуществлены на одном наблюдателе для 38 цветов. Они менее пригодны для целей построения равноконтрастного цветowego пространства,

поскольку выполнены для цветов разной яркости, и поэтому их труднее объединить в единую систему. В опытах Брауна [16], выполненных на 12 наблюдателях, яркость цвета также не стабилизировалась. Пороговые эллипсы, построенные по результатам этих двух работ, располагаются на цветовом графике менее системно, чем на рис. 2. Последними упомянем опыты Вышецки и Филдера [17, 18], которые в методическом отношении похожи на опыты Мак-Адама. Отличие состояло в том, что яркость сравниваемых цветов и фона была в два раза меньше, чем у Мак-Адама. Новым явилось то, что измерения порогов выполнялись не на плоскости, а в трехмерном пространстве. В результате была получена серия пороговых эллипсоидов. Точность этих экспериментов ниже, чем в опытах Мак-Адама.

Оценивая приведенные литературные данные, отметим, что в них совершенно отсутствуют опыты, у которых цветовые пороги определялись бы на фоне того же цвета. Как уже говорилось, это связано с непригодностью оптических колориметров к такого рода измерениям. Вместе с тем, очевидно, что подобные эксперименты дали бы наилучшую основу для построения теории равноконтрастного пространства, поскольку в данном случае для каждого цвета фон выбирался бы однозначно, причем простейшим способом. Нарушение этого принципа на практике привело к разнобою при выборе фона и сделало различные опыты несравнимыми между собой. Стабилизация яркости цветов, для которых определяются пороги, представляется удачным решением. То же можно сказать и об ограничении пороговых испытаний плоскостью равной яркости. В теоретическом плане – это простейшие ограничения, которые препятствуют распаду экспериментального материала.

В свете сказанного, описываемые ниже эксперименты по определению цветовых порогов выполнялись на компьютерном колориметре при следующих ограничениях: 1) цвет фона равен цвету тест-поля; 2) все опыты проводились при одной и той же яркости; 3) цветовые пороги измерялись только в плоскости равной яркости. Если для экспериментов, выполненных при таких условиях, удастся построить адекватную теорию, то она будет иметь наибольшую практическую значимость, поскольку в условиях реального процесса зрения для каждой точки изображения фоном обычно являются цвета, близкие к цвету данной точки как по цветности, так и по яркости.

Обозначим символами R', G', B' координаты цвета в компьютерной системе маркировки, а символами R'', G'', B'' – координаты цвета в «мозаичной» системе. Как было сказано выше, компьютерные и мозаичные координаты связаны между собой квадратичной зависимостью. При этом значению 255 компьютерной маркировки соответс-

твует единичное значение мозаичной маркировки. Отсюда вытекают следующие соотношения:

$$R'' = \frac{R'^2}{255^2}; \quad G'' = \frac{G'^2}{255^2}; \quad B'' = \frac{B'^2}{255^2}. \quad (2)$$

Равенства (2) позволяют осуществлять переход от компьютерных координат цвета к мозаичным и обратно.

Красный, зеленый и синий цвета, маркированные компьютерной координатой 255, согласно американскому стандарту (система NTSC), образуют белый цвет с координатами цветности $x_w = 0,310$; $y_w = 0,316$; [6]. Соотношения яркости этих цветов, согласно стандарту, имеют следующие значения: $0,299:0,587:0,114$ [19]. Яркость синего цвета, маркированного числом 255, принимаем за 1. Тогда яркость красного и зеленого цветов, обозначенных числом 255, составит соответственно $0,299/0,114 = 2,62$ и $0,587/0,114 = 5,16$. Пусть R, G, B – координаты цвета в системе NTSC. Тогда, согласно сказанному выше,

$$R = 2,62R''; \quad G = 5,16G''; \quad B = B''. \quad (3)$$

Равенства (3) позволяют осуществлять переход от мозаичных координат цвета к координатам того же цвета в системе NTSC и обратно.

Пересчет по формулам (2) и (3) дает следующие значения компьютерных координат для красного и зеленого цветов единичной яркости (в системе NTSC):

$$r' = \frac{255}{\sqrt{\frac{0,299}{0,114}}} = 157,45; \quad g' = \frac{255}{\sqrt{\frac{0,587}{0,114}}} = 112,38.$$

Специальные опыты с подравниванием яркости красного и зеленого основных цветов к основному синему интенсивности 255, выполненные с помощью цветовой мозаики, свидетельствуют, что фактические значения машинных координат красного и зеленого цветов соответствуют расчетным в пределах точности измерений.

Свяжем координаты цвета R, G, B принятой здесь системы NTSC с координатами цвета X, Y, Z международной системы МКО. Координаты цветности основных цветов в системе NTSC определяются, согласно стандарту, следующими значениями: $x_R = 0,67$; $y_R = 0,33$; $x_G = 0,21$; $y_G = 0,71$; $x_B = 0,14$; $y_B = 0,08$. Вычисляем координаты ортов R, G, B в системе XYZ . Согласно сказанному ранее, все три орта имеют единичную яркость, поэтому $Y_R = Y_G = Y_B = 1$. Координаты цветности x, y, z связаны с координатами цвета X, Y, Z зависимостями [19]:

$$x = \frac{X}{X+Y+Z}, \quad y = \frac{Y}{X+Y+Z}, \quad z = \frac{Z}{X+Y+Z},$$

поэтому:

$$X_R = Y_R \frac{x_R}{y_R} = 1 \frac{0,67}{0,33} = 2,0303 ,$$

$$Z_R = Y_R \frac{1-x_R-y_R}{y_R} = 1 \frac{1-0,67-0,33}{0,33} = 0 ,$$

$$X_G = Y_G \frac{x_G}{y_G} = 1 \frac{0,21}{0,71} = 0,2958 ,$$

$$Z_G = Y_G \frac{1-x_G-y_G}{y_G} = 1 \frac{1-0,21-0,71}{0,71} = 0,1127 ,$$

$$X_B = Y_B \frac{x_B}{y_B} = 1 \frac{0,14}{0,08} = 1,75 ,$$

$$Z_B = Y_B \frac{1-x_B-y_B}{y_B} = 1 \frac{1-0,14-0,08}{0,08} = 9,75 .$$

Согласно найденным значениям пересчет координат R, G, B цвета из колориметрической системы NTSC в координаты цвета X, Y, Z системы МКО описывается системой:

$$\begin{cases} X = 2,0303R + 0,2958G + 1,75B \\ Y = R + G + B \\ Z = 0,1127G + 9,75B \end{cases} \quad (4)$$

Обратный переход определяется системой равенств

$$\begin{cases} R = 0,5709372X - 0,1591740Y - 0,0861204Z \\ G = -0,5776139X + 1,1727296Y - 0,0166056Z \\ B = 0,0066766X - 0,0135555Y + 0,1027560Z \end{cases} \quad (5)$$

3. Экспериментальные исследования цветовых порогов

Использование цвета фона, который совпадает с цветами тест-полей, находящимися на пороге различения, позволило сократить число цветов, участвующих в эксперименте, с трех до двух. Во всех экспериментах, выполненных предшествующими исследователями и описанных в предыдущем параграфе, два маленьких тест-поля окружались третьим большим полем, играющим для них роль фона. В описываемых опытах использовались всего два поля: первое – небольшое в центре и второе – обширное поле, окружающее первое. Оба поля играют роль тест-полей, для цветов которых отыскивается пороговое отличие. Второе поле, кроме того, выполняет роль фона. В момент измерения порогового различия цветов тест-полей, цвет внешнего поля отличается всего лишь на один порог от цвета внутреннего поля и совпадает с цветом фона. Первое тест-поле располагается в центре экрана, оно имеет форму квадрата со стороной 20 мм. Фон формируется внутри квадрата со стороной 160 мм. За пределами этого квадрата – темнота (опыты выполнялись в затемненном помещении при отсутствии внешнего освещения). Испытуемый ведет наблюдение двумя глазами, фиксируя взгляд на малом тест-поле и рассматривая его с расстояния

70 см. Экран дисплея перпендикулярен оси зора наблюдателя, который смотрит горизонтально вперед. Угловой размер внутреннего тест-поля – около 2° , внешнего – около 16° .

Цвета, для которых отыскивались пороги, были выбраны следующим образом. Всего использовалось 15 цветов. Их расположение внутри треугольника телевизионных цветов показано на рис. 4. По сравнению с расположением цветов, принятым Мак-Адамом, распределение цветов на плоскости цветового графика более равномерное. Точки цветов, лежащие на периферии треугольника, максимально приближены к его границам.

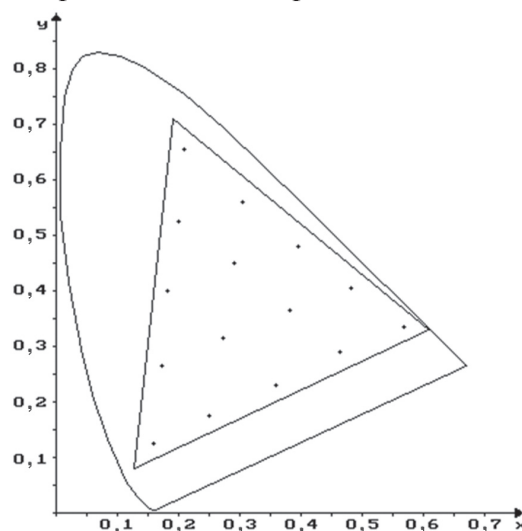


Рис. 4

Они отстоят от границы на величину, лишь немногим превышающую пороговую, что обеспечивает возможность определения порога по любому направлению от данного цвета. Этим достигается максимально возможный охват изучаемой области цветов экспериментом. Яркость всех цветов – единичная (в принятом здесь варианте колориметрической системы NTSC).

Координаты цветности x и y точек (в системе МКО) указаны в табл. 1.

Таблица 1

$x [1] = 0,175$	$y [1] = 0,125$
$x [2] = 0,190$	$y [2] = 0,265$
$x [3] = 0,200$	$y [3] = 0,400$
$x [4] = 0,220$	$y [4] = 0,525$
$x [5] = 0,230$	$y [5] = 0,655$
$x [6] = 0,335$	$y [6] = 0,560$
$x [7] = 0,320$	$y [7] = 0,450$
$x [8] = 0,300$	$y [8] = 0,315$
$x [9] = 0,275$	$y [9] = 0,175$
$x [10] = 0,395$	$y [10] = 0,230$
$x [11] = 0,420$	$y [11] = 0,365$
$x [12] = 0,435$	$y [12] = 0,480$
$x [13] = 0,530$	$y [13] = 0,405$
$x [14] = 0,510$	$y [14] = 0,290$
$x [15] = 0,625$	$y [15] = 0,335$

Пороги для каждого цвета измерялись по 16-ти направлениям. Способ выбора нумерации направлений определяется схемой, изображенной на рис. 5.

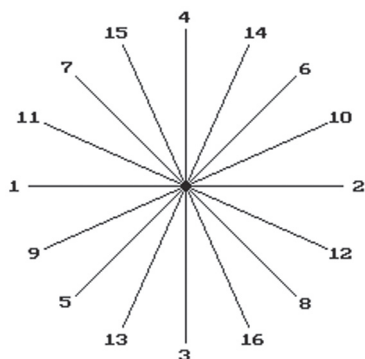


Рис. 5

В каждом эксперименте цветность внешнего тест-поля определялась координатами x и y одной из точек, взятых из табл. 1. Координаты цветности x' и y' внутреннего тест-поля определялись по формулам:

$$x' = 0,0001km, \quad y' = 0,0001kn. \quad (6)$$

Здесь числа m и n указывают направление, по которому цветность внутреннего тест-поля удаляется от цветности внешнего. Они указаны в табл. 2.

Таблица 2

№ направления	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
m	2	-2	0	0	2	2	-2	-2	2	2	-2	-2	1	1	-1	-1
n	0	0	2	-2	2	-2	-2	1	-1	1	1	-1	2	-2	2	-2

Число 0,0001 указывает размер одного продвижения по горизонтали или вертикали. Число m указывает, сколько всех продвижений делается по горизонтали за один шаг движения в заданном направлении. Число n указывает то же самое, но по вертикали. Число k указывает, сколько шагов сделано всего в заданном направлении.

Цвет тест-полей формируем с помощью мозаики, составленной из блоков размером 10×10 точек. Цвета отдельных точек блока выбираем следующим способом (одинаковым для всех блоков мозаики). По координатам цветности x, y заданного цвета тест-поля отыскиваем координаты этого же цвета X, Z в системе МКО по формулам [19]:

$$X = Y \frac{x}{y}; \quad Z = Y \frac{1-x-y}{y}. \quad (7)$$

Значение координаты Y данного цвета принимаем равным единице. Этим стабилизируется яркость цвета. Далее с помощью формул (5) по известным X, Y, Z вычисляем координаты RGB этого же цвета в системе NTSC. Затем по формулам (2) находим мозаичные координаты R'', G'', B'' цвета и, наконец, по формулам (4) определяем машинные координаты цвета тест-поля R', G', B' . Округляя их в меньшую и большую стороны до ближайших

целых чисел, получаем шесть машинных уровней красного, зеленого и синего основных цветов $R_{\min}, R_{\max}, G_{\min}, G_{\max}, B_{\min}, B_{\max}$.

Выбор цветов для точек блока начинаем с его левого верхнего угла, а затем движемся до конца блока слева направо, после этого переходим вниз на следующий ряд точек и движемся справа налево и так далее. Такой челночный способ обхода точек предотвращает образование муара и обеспечивает достаточно равномерное распределение цвета, видимого наблюдателем, по всей площади тест-поля с расстояния 0,7 м. Первую координату цвета текущей точки берем R_{\min} , если среднее значение этой же координаты, подсчитанной для всех предшествующих точек, больше R' . В противном случае берем R_{\max} . Аналогично выбираем вторую и третью координаты цвета текущей точки. Для первой точки берем цвет с компьютерными координатами $R_{\max}, G_{\max}, B_{\max}$. Ясно, что при таком способе выбора средняя i -я ($i = 1, 2, 3$) компьютерная координата цвета будет отличаться от той же по счету компьютерной координаты заданного цвета не более чем на 1% одного деления компьютерной шкалы, что обеспечивает приемлемую точность дозирования цвета тест-поля. Строго говоря, вычисление среднего значения следовало бы производить не для компьютерных, а для мозаичных координат, однако различие результатов оказывается ничтожно малым. Описанный способ формирования тест-полей реализован программно.

Во время проведения опыта цвет обоих тест-полей берется одинаковым. После этого числу k последовательно приписываются значения 1, 2, По формулам (6) подсчитываются координаты цветности x', y' внутреннего тест-поля, а по ним корректируется структура мозаики внутреннего поля. При этом подсчитываются приращения координат цвета, что на экране отображается практически мгновенно. Наблюдатель субъективно воспринимает работу программы как плавное изменение цвета по заданному направлению (одному из шестнадцати). Темп изменения цвета может быть по желанию экспериментатора произвольно ускорен или замедлен.

Экспериментатор увеличивает число k до тех пор, пока не обнаружит отличие цвета внутреннего поля от внешнего. После этого он начинает двигаться в противоположном направлении до исчезновения порога. Затем направление движения снова меняется. Такие колебательные движения производятся многократно, при этом темп изменения цвета постепенно снижается, а границы искомого цвета находятся более точно, в результате чего они сближаются. Регистрация координат полученных границ производится после того, как процесс стабилизируется и границы перестанут сближаться. В качестве результата эксперимента

берется цвет тест-поля, равный среднему значению цвета верхней и нижней границ. Описанный опыт повторялся 50 раз и подсчитывалось среднее значение для полученных замеров. Такие опыты производились по 16-ти направлениям для каждой точки. Все это делалось для каждой из 15-ти точек. Всего было произведено 12 тыс. измерений.

По средним значениям координат цвета внутреннего тест-поля по всем 16-ти направлениям для каждого из 15-ти цветов внешнего поля на цветовом графике МКО построены точки пороговых эллипсов (рис. 6). Сравнивая полученные результаты с пороговыми эллипсами Мак-Адама (рис. 2), замечаем, что размеры и ориентация эллипсов меняется в поле цветового графика более регулярно и монотонно, чем у Мак-Адама. Размещение эллипсов на плоскости цветового графика более равномерное.

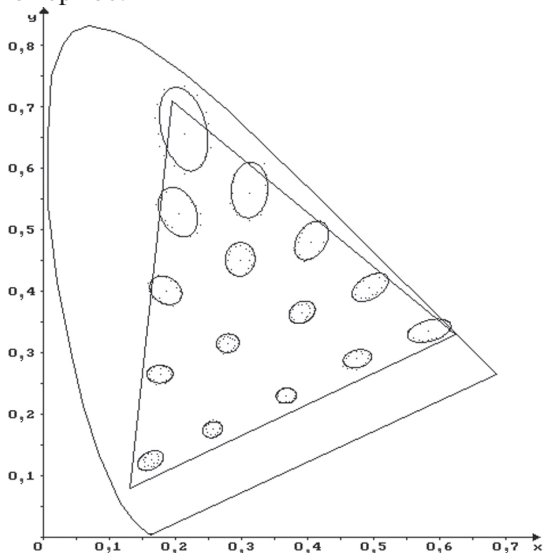


Рис. 6

В области серого цвета на графике Мак-Адама имеет место локальное уменьшение размеров эллипсов. Такое нарушение монотонности изменения параметров эллипсов, несомненно, вызвано влиянием искусственно выбранного серого фона. Эллипс наименьшего размера расположен на границе графика Мак-Адама в левом нижнем углу (синий цвет). На этом же графике эллипс наименьшего размера находится внутри графика в области серо-голубого цвета. Все сказанное позволяет сделать вывод, что результаты наших опытов более предпочтительны в роли экспериментальной основы для проверки теоретических построений.

4. Построение равноконтрастного цветового графика

После того как в предыдущем параграфе были получены эллипсы для цветовых порогов, построенные при условии, что цвет фона совпадает с цветом тест-поля, мы имеем возможность вернуться к задаче построения равноконтрастно-

го цветового графика. Решение подобной задачи было осуществлено в [20] на основе экспериментов Мак-Адама. Как уже говорилось, пороговые эллипсы Мак-Адама строились для случая, когда цвет фона не совпадает с цветом тест-поля. Поэтому следует ожидать, что построенный на новом экспериментальном материале равноконтрастный цветовой график получится иным. При решении задачи по-прежнему будем руководствоваться аналогией между механизмом цветового зрения и механизмом зрительного восприятия пространства. Вместе с тем следует иметь в виду, что аналогия, сколь бы глубокой она ни была, не обеспечивает полного тождества этих двух механизмов.

Одно из важных различий состоит в том, что физическое пространство обладает метрикой, тогда как для колориметрического пространства метрика вообще не введена. Оно задано лишь с точностью до любых аффинных преобразований. Поэтому все координатные системы, которые можно ввести в колориметрическом пространстве, равноправны, в том числе и косоугольные. Для физического же пространства подходят только прямоугольные координатные системы, поскольку только с ними согласуется его метрика. В связи с этим выбор направления координатных осей и масштабов по ним для физического пространства не вызвал никаких затруднений. Для колориметрического же пространства этот выбор представляет особую проблему. В случае использования косоугольных координат переход от одной координатной системы к другой приводит, вообще говоря, к перекосу пространства и к неодинаковым его деформациям по различным направлениям, нарушающим равенство расстояний. Поэтому эффект равноконтрастности может быть достигнут только при единственной аффинной деформации пространства (с точностью до смещения пространства, его поворотов и изменения масштаба, которые сохраняют равенство расстояний).

Исходной информацией для построения равноконтрастного цветового графика являются найденные пороговые эллипсы, изображенные на рис. 6. Пороговые эллипсы нанесены на диаграмме цветности МКО с координатами x и y , которые выражаются через координаты цвета X, Y, Z зависимостями (7). Диаграмма цветности получена следующим образом [19]. Цветовое пространство рассекается плоскостью, пересекающей оси X, Y, Z в точках $(1,0,0)$, $(0,1,0)$ и $(0,0,1)$. Получаемый на ней отпечаток локуса цветов затем проектируется на плоскость XY .

Возьмем в цветовом пространстве произвольную координатную систему UVW с началом в точке черного цвета и рассежем полученный в ней локус цветов какой-нибудь плоскостью. Из теории проективных преобразований [21] известно, что

отпечаток локуса на такой плоскости можно получить из диаграммы цветности с помощью проективного преобразования

$$u = \frac{ax}{bx + cy + d}, \quad v = \frac{y}{bx + cy + d}. \quad (8)$$

Здесь u, v – некоторые прямоугольные координаты на плоскости сечения; a, b, c, d – параметры проективного преобразования. Варьируя числовые значения параметров a, b, c, d , всегда можно получить из диаграммы цветности такой отпечаток локуса, который совпадет с его отпечатком на произвольном сечении цветового пространства при любой координатной системе. Говоря о совпадении отпечатков, мы имеем в виду возможность совмещения полученного отпечатка с заданным путем его перемещений и поворотов на плоскости сечения и изменения его масштаба.

Зададимся теперь вопросом, каким должен быть отпечаток сечения локуса плоскостью равной яркости, чтобы эту плоскость можно было принять в качестве аналога кампиметрической плоскости. Очевидно, для этого достаточно, чтобы на ней один из эллипсов превратился в круг, причем его диаметр был меньше, чем наименьший размер любого из оставшихся эллипсов. Местоположение этого круга совпадает с центром шахматной доски Гельмгольца. Для выполнения проективных преобразований диаграммы цветности по формулам (8) была разработана специальная программа. Изменяя значения параметров проективного преобразования и наблюдая на экране дисплея деформации эллипсов на диаграмме цветности, мы обнаружили, что в круг с наименьшим диаметром может обращаться только эллипс, находящийся в зоне серого цвета. Этот факт подтвердил аналогию между ахроматическим лучом в цветовом пространстве и зрительной осью в физическом пространстве. Полученное сечение локуса представлено на рис. 7.

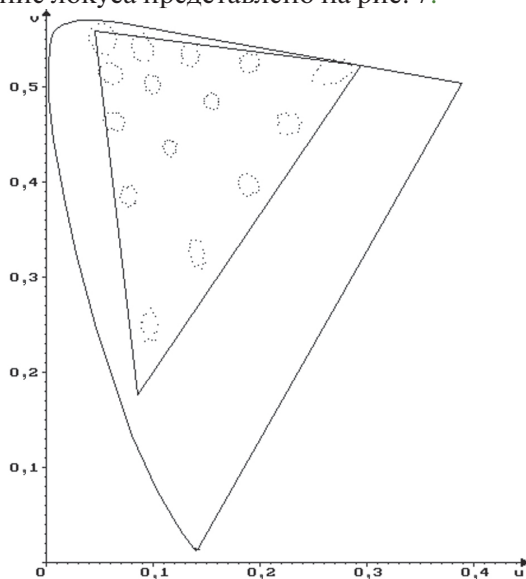


Рис. 7

Ему соответствуют следующие значения параметров проективного преобразования: $a = 3,2$; $b = 2,4$; $c = 3,4$; $d = 2,5$.

Осталось подобрать численное значение параметра R доски Гельмгольца с таким расчетом, чтобы деформация диаграммы цветности по формулам (9) превратила эллипсы в окружности [20].

$$\xi = \frac{k \cdot \operatorname{arctg} \frac{\sqrt{\xi^2 + \eta^2}}{R}}{\sqrt{\xi^2 + \eta^2}} \cdot \xi'; \quad \eta = \frac{k \cdot \operatorname{arctg} \frac{\sqrt{\xi^2 + \eta^2}}{R}}{\sqrt{\xi^2 + \eta^2}} \cdot \eta'. \quad (9)$$

С известным приближением это удастся осуществить при $R = 0,16$. Преобразование производилось с помощью специально созданной программы. Полученный график представлен на рис. 8. Это и есть искомый равноконтрастный цветовой график. Отклонение экспериментальных точек от расчетных положений (окружности одинакового радиуса) не превышает 25%.

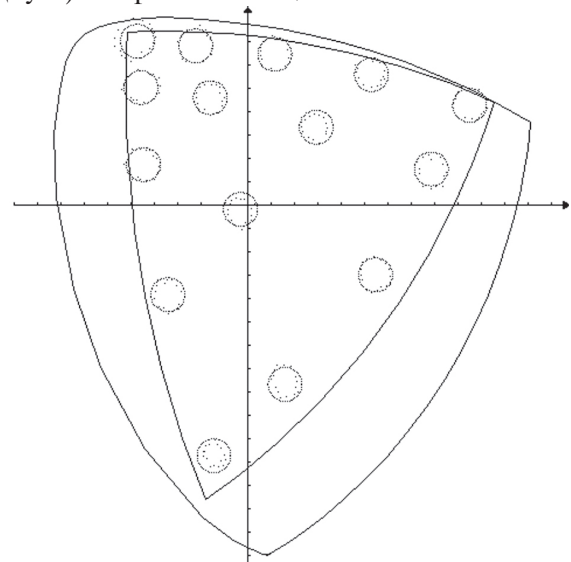


Рис. 8

Отношение максимального размера эллипса к минимальному не превосходит значения 1,5. Если мы ограничимся лишь проективным преобразованием, то отношение максимального размера эллипса к минимальному увеличивается до трех. На исходной же диаграмме цветности МКО это отношение возрастает до 8. Полученный результат свидетельствует о том, что аналогия между механизмами формирования диаграммы цветности и поля зрения человека, которой мы руководствовались, полностью себя оправдала. Теперь уже с полным основанием можно утверждать, что трехмерное цветовое пространство соответствует трехмерному физическому пространству, окружающему человека. Точке черного цвета цветового пространства соответствует оптический центр глаза. Лучам равной цветности в цветовом пространстве соответствуют лучи физического пространства, исходящие из оптического центра глаза. Ахроматическому

лучу соответствует зрительная ось. Равноконтрастной диаграмме цветности соответствует карта поля зрения. Линиям равного цветового тона на равноконтрастной диаграмме цветности соответствуют радиальные линии на карте поля зрения. Линиям равной насыщенности соответствуют кольцевые линии на карте поля зрения. Точке белого цвета на диаграмме цветности соответствует центр поля зрения.

5. Разработка способа экономного кодирования цвета

Изложенные выше исследования позволяют предложить способ экономного кодирования цвета. Он основан на пересчете координат цветности из системы xu , рекомендованной МКО, в разработанную равноконтрастную систему. Прямой пересчет производится по формулам:

$$u = \frac{2,5x}{-2,4x + 10,88y + 3,2}, \quad v = \frac{8y}{-2,4x + 10,88y + 3,2}, \quad (10)$$

$$\xi' = u - 0,135, \quad \eta' = v - 0,427, \quad (11)$$

$$\xi = \frac{k \cdot \arctg \frac{\sqrt{\xi'^2 + \eta'^2}}{R}}{\sqrt{\xi'^2 + \eta'^2}} \cdot \xi', \quad \eta = \frac{k \cdot \arctg \frac{\sqrt{\xi'^2 + \eta'^2}}{R}}{\sqrt{\xi'^2 + \eta'^2}} \cdot \eta'. \quad (12)$$

Обратный – по формулам:

$$\xi' = \frac{0,16 \operatorname{tg}(\frac{\sqrt{\xi^2 + \eta^2}}{1,5})}{\sqrt{\xi^2 + \eta^2}} \cdot \xi, \quad \eta' = \frac{0,16 \operatorname{tg}(\frac{\sqrt{\xi^2 + \eta^2}}{1,5})}{\sqrt{\xi^2 + \eta^2}} \cdot \eta, \quad (13)$$

$$u = \xi' + 0,135, \quad v = \eta' + 0,427, \quad (14)$$

$$x = \frac{3,2u}{2,4u + 3,4v + 2,5}; \quad y = \frac{v}{2,4u + 3,4v + 2,5}. \quad (15)$$

Ниже приводится пример пересчета. Пусть $x = 0,2$; $y = 0,3$. Тогда

$$\begin{aligned} u &= 0,0836, & v &= 0,4011, \\ \xi' &= -0,0474, & \eta' &= -0,0279, \\ \xi &= -24,54, & \eta &= -14,45. \end{aligned}$$

Пусть $\xi = 0,01$, $\eta = 0,01$. Тогда

$$\begin{aligned} \xi' &= 0,0000, & \eta' &= 0,0000, \\ u &= 0,1310, & v &= 0,4290, \\ x &= 0,310, & y &= 0,316. \end{aligned}$$

Единица младшего разряда получаемых таким способом кодов соответствует одному цветовому порогу. Таким образом, использование предлагаемых кодов, с одной стороны, обеспечивает точное воспроизведение цвета, а с другой – предельную экономию памяти при запоминании цветовых изображений.

Зависимости, предназначенные для экономного кодирования цвета, стандартизованы Междуна-

родной комиссией по освещению (МКО). Первый стандарт был принят в 1960 году [6], он имел вид зависимостей

$$u = \frac{4x}{-2x + 12y + 3}, \quad v = \frac{6y}{-2x + 12y + 3},$$

предложенных Мак-Адамом. Специальные исследования показали, что отношение максимального размера пороговых эллипсов к минимальному достигает пяти.

Второй и пока последний стандарт – система CIELUV, рекомендована МКО в 1976 г. по предложению Вышецки [6]. В плоскости равной яркости используется растянутый в полтора раза по вертикальной оси (модифицированный) график Мак-Адама 1960 года, дополнительно введены следующие эмпирические зависимости для различного уровня яркости:

$$L^* = 116 \sqrt[3]{\frac{Y}{Y_w} - 16},$$

$$u^* = 13 L^* (u' - u'_w), \quad v^* = 13 L^* (v' - v'_w),$$

где

$$u' = \frac{4x}{-2x + 12y + 3}; \quad v' = \frac{9y}{-2x + 12y + 3};$$

Y, Y_w – координаты исследуемого и белого цветов в системе МКО 1931 года; u', v', u'_w, v'_w – координаты цветности исследуемого и белого цветов в модифицированной системе Мак-Адама. В этой системе ошибка увеличивается до отношения (9), но зато цветовые коды можно пересчитать для всех точек цветового пространства, а не только для цветов равной яркости.

Предлагаемые в работе зависимости имеют следующие отличия от стандартизованных зависимостей МКО. В основе схемы лежат два простых и легко проверяемых закона: $\gamma = \beta, \rho = k\alpha$ [20]. Проведенные экспериментальные исследования показали, что эти законы выполняются для цветового зрения человека с высокой точностью. Они основаны не только на эмпирическом материале, но построены согласно простой теоретической схеме, открытой Гельмгольцем в механизме зрительного восприятия пространства. Численные значения параметров рекомендуемых преобразований получены для случая, когда фон равен наблюдаемому цвету, что является лучшим приближением к условиям реального видения предметов, чем в случае постоянной фиксации фона для всех цветов (а именно это заложено в основу стандартов МКО). Отношение между максимальным и минимальным размерами пороговых эллипсов не превышает 1,5, что в 4 раза ниже, чем в стандарте 1960 года. Все сказанное дает основание рекомендовать формулы (10)-(12) в качестве уточнения стандарта МКО.

Для удобства практического пользования формулами (10)-(12) разработан компьютерный цве-

товой атлас, в котором сформированы все различаемые глазом телевизионные цвета. Все цветовое пространство расслаивается плоскостями равной яркости. Положение плоскости равной яркости фиксируется по желанию с точностью до одного порога на уровне любой яркости. Для этой яркости формируется равноконтрастный график цветности. Одновременно можно вывести на экран ПК некоторую часть цветового атласа (в зависимости от масштаба). По желанию пользователя различия между соседними полями можно задать как допороговыми, так и сверхпороговыми с любым заданным шагом. Кроме того, можно вывести на экран код любого цвета и его координаты цвета XYZ , и координаты цветности xu . Погрешность задания цвета в координатах цветности xu не превышает 0,0005.

Использование цвета фона, равного цвету тест-поля, привело к тому, что эффекты Бецольда-Брюкке и Эбнея [19] отсутствуют, а яркостные и цветовые пороги в пределах принятого диапазона изменения интенсивности цвета практически точно подчиняются закону Вебера-Фехнера. Для проверки этого факта формировались цилиндрические сечения цветового тела, соосные его ахроматической оси и оценивалась степень постоянства насыщенности в каждом сечении. Кроме того, производились в разных направлениях плоские сечения цветового тела, проходящие через ахроматическую ось, и оценивалась степень постоянства цветов в каждом таком сечении. Кроме того, для различных цветностей проверялись по различным направлениям пропорциональность цветовых порогов уровню яркости цвета.

Атлас цветов, реализованный на ПК, обладает рядом преимуществ по сравнению с существующими атласами (атлас Манселла [19], атлас DIN [6] и другие). Цвета его полей строго фиксированы и не зависят от характера освещения нарисованных. Число цветов атласа практически неограничено, можно сформировать любой желаемый цвет в пределах принятого диапазона цветов. Атлас реализует равноконтрастное цветовое пространство. Для любого цвета указываются его координаты в любой системе (колориметрической XYZ , RGB , машинной и других).

Формулы (10)-(12), определяющие равноконтрастное цветовое пространство, колориметр на базе ПК и атлас цветов рекомендуются для практического применения.

Выводы

На базе ПК создан программируемый колориметр – система, позволяющая проводить колориметрические эксперименты. Ее возможности в некоторых отношениях шире, чем у оптического колориметра: во-первых, работа ведется не с излучениями, а с

цветами, во-вторых – можно ставить эксперименты с переменным цветом фона, что позволяет изучать индукцию цветового зрения. Возможны также эксперименты с динамическими изображениями для изучения адаптационных явлений.

Экспериментально исследованы пороги цветоразличения с учетом влияния явления цветовой индукции в пределах цветового треугольника системы NTSC.

На основе результатов эксперимента построена модель и найдены параметры преобразования колориметрического пространства цветов в равноконтрастное цветовое пространство.

По найденной аналитической модели равноконтрастного цветового пространства создан программируемый равноконтрастный цветовой атлас с изменяемым шагом и яркостью, независимый от внешнего освещения.

Предложен метод экономного кодирования цветового сигнала, содержащего полную (для человеческого зрения) и неискаженную информацию о цвете.

Список литературы: 1. Нюберг, Н.Д. Измерение цвета и цветовые стандарты [Текст] / Н.Д. Нюберг – М.: Стандартизация и рационализация, 1933. – 104 с. 2. Зингер, Б. Достижения в технике передачи и восприятия изображений [Текст] / Сост. Б. Зингер, Р. Кола, Д.Ф. Барбе, С.Б. Кампана / Пер. с англ. Ю.Н. Богачкова и др.; Под ред. Ю.Н. Богачкова. – М.: Мир, 1980. – 312 с. 3. Бондаренко, М.Ф. Компараторная идентификация цветового зрения человека [Текст] / М.Ф. Бондаренко, С.Ю. Шабанов-Кушнаренко, Ю.П. Шабанов-Кушнаренко // Бионика интеллекта: науч.-техн. журнал. – 2008. – № 2(68). – С. 3-12. 4. Смирнов, А.Д. Математические модели теории передачи изображений [Текст] / А.Д. Смирнов – М.: Связь, 1979. – 96 с. 5. Прэтт, У.К. Методы передачи изображений: Сокращение избыточности [Текст] / У.К. Прэтт, Д.Д. Саркисон, Г.Д. Ханс и др.; под ред. У.К. Прэтта / Пер. с англ. под ред. Л.С. Виленчика. – М.: Радио и связь, 1983. – 264 с. 6. Мешков, В.В. Основы светотехники [Текст] / В.В. Мешков, А.Б. Матвеев; Ч.2. – М.: Энергоатомиздат, 1989. – 429 с. 7. Цуккерман, И.И. Цифровое кодирование телевизионных изображений [Текст] / И.И. Цуккерман, Б.М. Кац, Д.С. Лебедев и др.; под ред. И.И. Цуккермана. – М.: Радио и связь, 1981. – 239 с. 8. Mac-Adam, D.L. Projective transformations of I.C.I. color specifications [Текст] / D.L. Mac-Adam // JOSA, 1937, Vol. 27. – P. 294. 9. Mac-Adam, D.L. Specification of small chromaticity differences [Текст] / D.L. Mac-Adam // JOSA, 1943, Vol. 33. – P. 18. 10. Теплов, Б.М. О законах пространственного и временного смешения цветов [Текст] / Б.М. Теплов, С.П. Яковлева // Зрительные ощущения и восприятия. – М.: Соцэкгиз, 1935. – Т.2. – С. 79-96. 11. Talbot, H. Experiments of light [Текст] / H. Talbot // Phil. Mag., 1834, №5. – P. 65-91. 12. Wright, W. The graphical representation of small colour differences [Текст] / W. Wright // JOSA, 1943, Vol. 33. – P. 232-245. 13. Wright, W. The sensitivity of the eye to small colour differences [Текст] / W. Wright // Proc. Phys. Soc, London, 1941, Vol. 53. – P.93-106.

14. Mac-Adam, D.L. The graphical representation of small color differences [Текст] / D.L. Mac-Adam // JOSA, 1943, Vol.33. – P. 632. 15. Brown, W. R. L. Visual sensitivities to combined chromaticity and luminance differences [Текст] / Brown W. R. L., Mac Adam D. L. // JOSA, 1949, Vol. 39, № 10. – P. 808-834. 16. Brown, W. R. L. Color discrimination of twelve observers [Текст] / W. R. L. Brown // JOSA, 1957, Vol. 49. – P. 134-143. 17. Wyszecki, G. Color-differences matches [Текст] / G.Wyszecki, G. Fielder // JOSA, 1971, Vol. 61. – P.1501. 18. Wyszecki, G. New color-matching ellipses [Текст] / G.Wyszecki, G. Fielder // JOSA, 1971, Vol. 61. – P. 1135. 19. Мешков, В.В. Основы светотехники [Текст] / В.В. Мешков – Ч. 2. – М.-Л.: Госэнергоиздат, 1961. – 416 с. 20. Бондаренко, М.Ф. Практические приложения компараторной идентификации линейных конечномерных объектов [Текст] / М.Ф. Бондаренко, С.Ю. Шабанов-Кушнарченко, Ю.П. Шабанов-Кушнарченко // Бионика интеллекта: науч.-техн. журнал. – 2009. – № 2(70). – С. 00-00. 21. Ефимов, Н.В. Высшая геометрия [Текст] / Н.В. Ефимов. – М.: Физматгиз, 1961. – 580 с.

Поступила в редколлегию 10.09.2009

УДК 519.7

Розробка системи ошадливого кодування кольору / М.Ф. Бондаренко, С.Ю. Шабанов-Кушнарченко, Ю.П. Шабанов-Кушнарченко // Біоніка інтелекту: наук.-техн. журнал. – 2009. – № 2 (71). – С. 13-23.

У статті описано постановку й результати колориметричних експериментів з дослідження порогів кольоророзрізнення. Побудовано аналітичну модель перетворення колориметричного простору кольорів у рівноконтрастний кольоровий простір.

Л. 8. Бібліогр.: 21 найм.

UDC 519.7

The colour economical coding system engineering / M.F. Bondarenko, S.Yu. Shabanov-Kushnarenko, Yu.P. Shabanov-Kushnarenko // Bionika intellekta: Sci. Mag. – 2009. – № 2 (71). – P. 13-23.

In article colorimetric experiments statement and results on research of colorimetric thresholds are described. The analytical model of colorimetric spaces transformation of colours in equal-contrast colour space is constructed.

Fig. 8. Ref.: 21 items.

УДК 519.7



МЕТОДЫ ИДЕНТИФИКАЦИИ МЕХАНИЗМА СУБЪЕКТИВНЫХ СОЦИАЛЬНО-ЭКОНОМИЧЕСКИХ ОЦЕНОК

М.Ф. Бондаренко¹, С.Ю. Шабанов-Кушнаренко², Ю.П. Шабанов-Кушнаренко³

^{1, 2, 3}ХНУРЭ, г. Харьков, Украина

Теория компараторной идентификации может быть применена не только для моделирования цветового зрения человека, но также и для целей математического описания других функций человеческого интеллекта (слух, осязание, узнавание, понимание), технических объектов. Здесь рассматривается компараторная идентификация субъективных оценок социально-экономических процессов.

КОМПАРАТОРНАЯ ПАРАМЕТРИЧЕСКАЯ ИДЕНТИФИКАЦИЯ, МОДЕЛИ ОЦЕНОК, ПРЕДПОЧТЕНИЯ, ЛИНЕЙНАЯ МОДЕЛЬ, ПРЕДИКАТ ПОРЯДКА

Введение

Наличие модели, достаточно точно предсказывающей оценку человеком места работы при ее поиске и выборе, важно для управления движением кадров и миграцией населения, для снижения текучести кадров на предприятиях, сокращения потерь среднего времени между увольнением и поступлением на работу, оптимизации распределения трудовых ресурсов по отраслям, планирования развития предприятия, жилищного строительства и тому подобного.

Место работы формально характеризуется вектором $x = (x_1, x_2, \dots, x_m)$. Здесь x_1, x_2, \dots, x_m – вещественные числа, характеризующие различные стороны места работы, такие как размер зарплаты, удаленность от места жительства, срок предоставления жилплощади и ее размер, уровень обеспеченности дошкольными учреждениями и тому подобное. Оценка места работы характеризуется вещественным числом u , называемым степенью привлекательности места работы, которое линейно зависит от вектора x :

$$u = F(x) = x_1 k_1 + x_2 k_2 + \dots + x_m k_m. \quad (1)$$

Обозначим $x_1 k_1 = u_1, x_2 k_2 = u_2, \dots, x_m k_m = u_m$, тогда

$$u = u_1 + u_2 + \dots + u_m. \quad (2)$$

Числа $u_1 + u_2 + \dots + u_m$ называются компонентами степени привлекательности места работы, они характеризуют степень привлекательности каждой стороны места работы.

Весовые коэффициенты k_1, k_2, \dots, k_m численно характеризуют удельную привлекательность каждой стороны места работы для данного человека. Набор чисел (k_1, k_2, \dots, k_m) полностью определяет реакцию конкретного лица, выбирающего место работы. Согласно описываемой здесь модели человек всегда предпочитает то место работы, которое обладает для него наивысшей степенью привлекательности. Целью структурной идентификации процесса оценки человеком места работы является решение вопроса о применимости описанной выше линейной модели к лицам, размышляющим

над выбором места работы. Задачей параметрической идентификации служит выведение из фактов поведения лица, выбирающего место работы, численных значений коэффициентов k_1, k_2, \dots, k_m .

Наличие социально-экономической интерпретации открывает дорогу для практического применения теории компараторной идентификации линейных конечномерных объектов к изучению процесса оценки человеком места работы. Вопрос о применимости линейной модели решается методами, изложенными в [1, 2]. Весовые коэффициенты k_1, k_2, \dots, k_m , характеризующие поведение лица, выбирающего место работы, могут быть найдены методами, описанными в [3].

1. Компараторная идентификация процесса оценки человеком места работы

Применение компараторного метода предполагает наличие у человека способности устанавливать равенство и неравенство степени привлекательности любых двух мест работы. В действительности человек способен даже на большее: он может, кроме того, устанавливать, какое из предъявленных ему мест работы лучше, а какое – хуже. Два места работы будем считать равноценными, если человек затрудняется отдать предпочтение одному из них. Человек, устанавливающий равноценность и неравноценность двух мест работы x и y , реализует своим поведением предикат

$$E(x, y) = D(F(x), F(y)), \quad (3)$$

равный единице, когда места работы равноценны, и равный нулю – в противном случае, то есть в случае, когда одно из мест работы предпочтительнее другого. Приведенные выше соображения вводят, в дополнение к рассмотренным ранее, еще одну интерпретацию метода компараторной идентификации, которую мы назовем социально-экономической. Ранее же нами были введены три интерпретации, а именно – цветовая, техническая и пространственная. В отличие от других интерпретаций, при социально-экономической интерпретации выходной сигнал линейного объекта одномерен, следовательно, $n = 1$.

Специфика социально-экономических объектов требует большого объема статистических измерений для обеспечения достаточной точности моделирования. В этом случае была использована модель не в виде предиката равенства (3), требующего большого количества измерений, а в виде предиката порядка

$$P(F(x), F(y)),$$

где

$$P(u, v) = \begin{cases} 1, & \text{если } u \geq v, \\ 0, & \text{если } u < v. \end{cases}$$

Поскольку выходные сигналы объекта u , v можно идентифицировать как вещественные числа [4], то порядок на множестве входных сигналов существует и применение в качестве модели предиката порядка математически обосновано. Как показала практика, эта модель для социально-экономических объектов удобнее и значительно сокращает объем необходимых измерений.

2. Функциональный метод шкалирования входных сигналов объекта идентификации

Обратим внимание на одну проблему, выходящую за рамки задачи компараторной идентификации, без решения которой невозможно приступить к математическому описанию объекта. Речь идет о правильном выборе шкалы описания входных сигналов объекта. При компараторной идентификации цветового зрения человека эта проблема не возникала, поскольку Ньютон, описывая световое излучение энергетическим спектром, очень удачно выбрал шкалу. Именно при такой шкале цвет излучения оказывается линейной функцией излучения. Совсем иное создалось бы положение, если бы мы попытались использовать для описания светового излучения амплитудный спектр соответствующего электромагнитного колебания. Вторая шкала связана с первой нелинейно, поскольку мощность излучения при каждой длине волны пропорциональна квадрату амплитуды соответствующей гармоники электромагнитного колебания. Поэтому при описании светового излучения вторым способом его цвет уже не будет линейно зависеть от цвета, и метод компараторной идентификации линейного конечномерного объекта в данном случае окажется неприменимым.

Специфика задачи компараторной идентификации процесса оценки человеком места работы заключается в том, что компоненты x_1, x_2, \dots, x_m вектора места работы x по своей природе разнородны, каждый из них требует описания в своей собственной шкале. Нет никаких оснований надеяться, что с самого начала удастся выбрать шкалы правильно. Предположим, что все шкалы выбраны неправильно. Обозначим вектор места работы, получаемый при этих шкалах, символом

x' , а его компоненты – символами x'_1, x'_2, \dots, x'_m , так что $x' = (x'_1, x'_2, \dots, x'_m)$. Зависимость степени привлекательности u места работы от параметров x'_1, x'_2, \dots, x'_m запишется в виде:

$$u = F'(x) = g_1(x'_1)k_1, g_2(x'_2)k_2, \dots, g_m(x'_m)k_m. \quad (4)$$

Здесь $g_i(x'_i) = x_i \overline{(i=1, m)}$ – некоторые функции, которые связывают сигналы x'_i и x_i , представленные соответственно в неправильной и правильной шкале.

Обозначим через $G(x') = x$ зависимость, связывающую векторы x' и x одного и того же места работы, представленного соответственно в неправильных и правильных шкалах. Она задается равенством

$$G(x'_1, x'_2, \dots, x'_m) = (g_1(x'_1)k_1, g_2(x'_2)k_2, \dots, g_m(x'_m)k_m). \quad (5)$$

Человек, устанавливающий равноценность и неравноценность двух мест работы x' и y' , описанных в неправильных шкалах, реализует своим поведением предикат

$$E'(x', y') = E(G(x'), G(y')). \quad (6)$$

Предикат $E(x, y)$ характеризует то же поведение человека, но для входных сигналов $x = G(x')$ и $y = G(y')$, описанных в правильных шкалах.

Рассмотрим условие

$$E'((x'_1, a'_2, \dots, a'_{i-1}, a'_i, a'_{i+1}, \dots, a'_m), (a'_1, a'_2, \dots, a'_{i-1}, a'_i, a'_{i+1}, \dots, a'_m)) = 1. \quad (7)$$

Здесь a'_1, a'_2, \dots, a'_m – фиксированные компоненты вектора места работы, описанного в неправильных шкалах, x'_1 и x'_1 – переменные компоненты, $i = \overline{1, m}$. Для каждого значения переменной x'_i исследователь так подбирает значение переменной x'_i , чтобы соблюдалось условие равноценности мест работы $(x'_1, a'_2, \dots, a'_{i-1}, a'_i, a'_{i+1}, \dots, a'_m)$ и $(a'_1, a'_2, \dots, a'_{i-1}, a'_i, a'_{i+1}, \dots, a'_m)$ для данного человека, оценивающего место работы. В результате из опыта описываются зависимости $x'_i = h_i(x'_1)$, где $i = \overline{1, m}$.

Пользуясь зависимостями (1), (3), (5) и (6), перепишем условие (7) в виде:

$$g_1(x'_1)k_1 + g_2(a'_2)k_2 + \dots + g_{i-1}(a'_{i-1})k_{i-1} + g_i(a'_i)k_i + g_{i+1}(a'_{i+1})k_{i+1} + g_m(a'_m)k_m = g_1(a'_1)k_1 + g_2(a'_2)k_2 + \dots + g_{i-1}(a'_{i-1})k_{i-1} + g_i(x'_i)k_i + g_{i+1}(a'_{i+1})k_{i+1} + g_m(a'_m)k_m.$$

После упрощений имеем $g_1(x'_1)k_1 + g_i(a'_i)k_i = g_1(a'_1)k_1 + g_i(x'_i)k_i$. Решая последнее уравнение относительно x'_i , получаем $x'_i = g_i^{-1}(g_1(x'_1)k_1/k_i + g_i(a'_i) - g_1(a'_1)k_1/k_i)$. Таким образом,

$$h_i(x'_1) = g_i^{-1}(g_1(x'_1)k_1/k_i + g_i(a'_i) - g_1(a'_1)k_1/k_i). \quad (8)$$

Пусть $x'' = (x''_1, x''_2, \dots, x''_m)$ – вектор места работы, компоненты которого представлены в промежуточных шкалах, для которых $x''_1 = x'_1$, $h_i(x''_1) = x''_i$,

где $i = \overline{1, m}$. Введем функцию $H(x'') = x''$, задаваемую равенством

$$H(x''_1, x''_2, \dots, x''_m) = (x''_1, h_2(x''_2), \dots, h_m(x''_m)). \quad (9)$$

Она связывает векторы x' и x'' одного и того же места работы, представленного соответственно в неправильных и промежуточных шкалах. Человек, устанавливающий равноценность и неравноценность двух мест работы x'' и y'' , описанных в промежуточных шкалах, реализует своим поведением предикат

$$E''(x'', y'') = E'(H(x''), H(y'')). \quad (10)$$

Предположим, что два места работы $(x''_1, x''_2, \dots, x''_m)$ и $(h''_1, h''_2, \dots, h''_m)$, описанные в промежуточных шкалах, равноценны. Это означает, что

$$E''((x''_1, x''_2, \dots, x''_m), (h''_1, h''_2, \dots, h''_m)) = 1. \quad (11)$$

Согласно (9) и (10), это условие равносильно следующему:

$$E''((x''_1, h_2(x''_2), \dots, h_m(x''_m)), (h''_1, h_2(h''_2), \dots, h_m(h''_m))) = 1.$$

В свою очередь, согласно (5) и (6), последнее равенство равносильно условию:

$$E''((g_1(x''_1), g_2(x''_2), \dots, g_m(x''_m)), (g_1(h''_1), g_2(h''_2), \dots, g_m(h''_m))) = 1.$$

Далее, согласно (1) и (3), только что записанное условие переписываем в виде

$$g_1(x''_1)k_1, g_2(h_2(x''_2))k_2 + \dots + g_m(h_m(x''_m))k_m = g_1(h''_1)k_1, g_2(h_2(h''_2))k_2, \dots, g_m(h_m(h''_m))k_m.$$

После замены по (8) и упрощений в последнем равенстве, окончательно получаем:

$$g_1(x''_1) + g_1(x''_2) + \dots + g_1(x''_m) = g_1(h''_1) + g_1(h''_2) + \dots + g_1(h''_m). \quad (12)$$

Полагая

$$F''(x'') = g_1(x''_1) + g_1(x''_2) + \dots + g_1(x''_m), \quad (13)$$

имеем

$$E''(x'', y'') = D(F''(x''), F''(y'')). \quad (14)$$

Значения функции $F''(x'')$ можно принять в качестве степени привлекательности u места работы x'' , представленного в промежуточных шкалах. Приравнявая слагаемые, зависящие от переменных x_i и x''_i ($i = \overline{1, m}$) в выражениях (1) и (3), получаем $x_i k_i = g_1(x''_i)$, откуда

$$x_i = 1/k_i g_1(x''_i). \quad (15)$$

Выражение (15) показывает, что правильные шкалы для всех компонентов $x''_1, x''_2, \dots, x''_m$ места работы определяются единственной функцией g_1 с точностью до масштаба. Таким образом, для

отыскания правильных шкал описания входных сигналов объекта осталось определить из опыта вид функции g_1 .

Предположим, что два места работы

$$(x''_1, a''_2, \dots, a''_{i-1}, x''_i, a''_{i+1}, \dots, a''_m) = (a''_1, a''_2, \dots, a''_{i-1}, x''_i, a''_{i+1}, \dots, a''_m),$$

описанные в промежуточных шкалах, равноценны. Для каждого значения переменной x''_1 исследователь так подбирает значение переменной x''_i , чтобы только что сформулированное условие соблюдалось. В результате из опыта находится зависимость $x''_i = l(x''_1)$. Вид функции (1), вообще говоря, будет иным, если по-другому зафиксировать параметры a_1, a_2, \dots, a_m . С другой стороны, согласно (13) и (14), равноценность мест работы означает, что

$$g_1(x''_1) + g_1(a''_2) + \dots + g_1(a''_{i-1}) + g_1(x''_i) + g_1(a''_{i+1}) + \dots + g_1(a''_m) = g_1(a''_1) + g_1(a''_2) + \dots + g_1(a''_{i-1}) + g_1(x''_i) + g_1(a''_{i+1}) + \dots + g_1(a''_m).$$

После упрощений имеем

$$2g_1(x''_1) = g_1(a''_1) + g_1(x''_i).$$

Решаем последнее уравнение относительно x''_i : $x''_i = g_1^{-1}(2g_1(x''_1) - g_1(a''_1))$. Таким образом,

$$l_{a''_1}(x''_1) = g_1^{-1}(2g_1(x''_1) - g_1(a''_1)). \quad (16)$$

При функции l указан индекс a''_1 , поскольку она зависит от параметра a''_1 . Выражение (16) представляет собой функциональное уравнение, поскольку требуется по известной функции $l_{a''_1}$ отыскать функцию g_1 . Общий метод решения этого уравнения нам неизвестен. Однако в каждом конкретном случае его можно с определенной степенью приближения решить методом подбора функции g_1 . Суть метода заключается в следующем. Принимаем в роли g_1 какую-нибудь из стандартных монотонных функций: степенную, показательную, логарифмическую, степенной полином заданной степени и тому подобное. Подставляя выбранную функцию g_1 в правую часть равенства (16), получаем выражение для функции $l_{a''_1}$ при конкретном значении параметра a''_1 . Далее определяем такие значения констант, определяющих конкретный вид функции g_1 (например, значение показателя k для степенной функции x''_1^k), при которых имеет место наилучшее приближение правой части равенства (16) к уже известному графику функции $l_{a''_1}$.

Например, применяя в роли функции g_1 степенную функцию с произвольным показателем k $g_1(x''_1) = x''_1^k$, получаем согласно (16) следующее выражение для функции $l_{a''_1}$:

$$l_{a''_1}(x''_1) = \sqrt[k]{2(\xi''_1)^k - (a''_1)^k}. \quad (17)$$

При $a''_1 = 0$ имеем:

$$l_0(x''_1) = \sqrt[k]{2\xi''_1}. \quad (18)$$

Если экспериментально найденная функция l_0 не может быть признана прямой, проходящей через начало координатной системы $x''_1 x''_i$, то нужно попробовать какую-нибудь другую шкалу, а если все возможности такого рода исчерпаны, применить степенной полином или какой-либо другой способ многопараметрического задания функции g''_1 . Если же функция l_0 подходит, то по углу наклона прямой $x''_1 = \sqrt[k]{2\xi''}$ находим значение показателя степени k . Для окончательного заключения о приемлемости степенной шкалы следует еще удостовериться в том, что результаты вычислений по формуле (17) при ненулевых значениях параметра a''_1 также согласуются с экспериментальными кривыми $l_{a''}$.

3. Табулирование изоморфизмов линейного предиката

Рассмотрим модель компараторной идентификации в виде m -мерного линейного предиката:

$$E(x, y) = D(Ax, Ay), \quad (19)$$

$x, y \in R^n$; A — матрица размерности $m \times n$ и ранга m , $m \leq n$. Для такой модели разработаны методы параметрической идентификации матрицы A [2].

Из практики известно, что существуют объекты, моделируемые следующим предикатом:

$$E_1(\xi, \eta) = D(F(\xi), F(\eta)), \quad (20)$$

где F — нелинейная вектор-функция; $\xi, \eta \in R^n$; $F(\xi), F(\eta) \in R^m$. Эта нелинейность делает невозможным применение указанных выше параметрических методов идентификации. С другой стороны, недоступность выходных сигналов некоторых таких объектов для прямого измерения означает неприменимость к ним классических методов идентификации.

Найдем условие, достаточное для того, чтобы существовали шкалы измерения входных сигналов, делающие модель объекта линейной. Введем следующее ограничение на характеристическую функцию F предиката (19): для любых ξ, η существует такой изоморфизм φ предиката $E_1(\xi, \eta)$ и линейного предиката $E(x, y)$,

$$E_1(\xi, \eta) = E(\varphi(\xi), \varphi(\eta)) = E(x, y), \quad (21)$$

что

$$\varphi(\xi) = (\varphi_1(\xi), \varphi_2(\xi), \dots, \varphi_n(\xi)) = (x_1, x_2, \dots, x_n). \quad (22)$$

Если выполняются эти условия, то, зная вектор-функцию φ , то есть способ изменения шкал входных сигналов, можно перейти от нелинейной модели объекта E_1 к линейной E .

Целью настоящей работы является построение алгоритма нахождения вектор-функции φ . Поскольку φ находится из эксперимента, по точкам, то построим алгоритм табулирования φ . Начнем разработку алгоритма с определения размерности

m вектор-функции F . Возьмем произвольный входной сигнал ξ и подберем входной сигнал η так, чтобы при любом параметре k выполнялось неравенство $F(x) \neq F(k\eta)$, $k\eta = (k\eta_1, k\eta_2, \dots, k\eta_n)$. Затем подберем входной сигнал γ так, чтобы при любых параметрах k и l выполнялось неравенство $F(l\xi + k\eta) \neq F(\gamma)$. Продолжаем эту процедуру до тех пор, пока будут находиться выходные сигналы с таким свойством. Полученный набор будет одним из возможных базисов подпространства входных сигналов, которое содержит область значений характеристической вектор-функции F . Размерность этого подпространства будет совпадать с размерностью F . Возможны три, с практической точки зрения, различных случая: $m=1$, $2 \leq m < n$, $m=n$. Последний случай тривиален, поскольку при $m=n$ и выполнении условий (21) и (22) предикаты $E_1 = E = D$.

Рассмотрим случай $m < n$. Пусть предикаты E и E_1 связаны равенствами (21) и (22), причем $\varphi_i(\xi)$ — биекции, $\varphi_i: R \rightarrow R$. Пусть также из любого сочетания $n-1$ индексов от 1 до n можно выбрать m таких, что

$$\forall \xi \in R^n \exists \{\delta_i \in R\}_{i \in I}, E_1(\xi, \sum_{i \in I} \Delta_i) = 1, \quad (23)$$

где I — множество из m выбранных индексов, $\Delta_i \in R^n$, причем i -я координата Δ_i равна δ_i , а остальные, соответственно, δ_j^0 — такой величине, что $\varphi_j(\delta_j^0) = 0$. (Последнее условие несколько усиливает условие m -мерности предиката E , требуя, чтобы почти все наборы из m единичных векторов были для него базисными).

Процедура табулирования всех n функций φ_i аналогична. Рассмотрим процедуру табулирования функции φ_1 .

1. Определение нулей 0_i функций φ_i , шагов табулирования h_i (в областях реальных значений функций φ_i), верхних и нижних границ p_i^+ и p_i^- значений функций φ_i ($i = \overline{1, n}$).

Только в этом пункте требуется использование явных значений вектор-функции φ — величин 0_i , h_i , p_i^+ , p_i^- . Значениями p_i^+ и p_i^- примем соответственно максимальное и минимальное значения функций φ_i . В качестве нулей функций φ_i можно брать любые их значения, поскольку они определяют только положение системы координат, в которых находятся. С целью упрощения алгоритма табулирования примем $0_i = p_i^-$. Такой выбор 0_i располагает все остальные значения φ_i в положительной области. Величины шагов табулирования h_i зависят от необходимой точности модели; число получаемых экспериментальных точек k можно вычислить по формуле $k = ((p_i^+ - 0_i) / h_i) + 1$.

Найдем такие величины δ_i^0 , δ_i^h , δ_i^+ , что

$$\varphi(\delta_1^0, \delta_2^0, \dots, \delta_n^0) = (0_1, 0_2, \dots, 0_n),$$

$$\varphi(\delta_1^0, \dots, \delta_i^h, \dots, \delta_n^0) = (0, h_i, \dots, 0_n),$$

$$\varphi(\delta_1^0, \dots, \delta_i^+, \dots, \delta_n^0) = (0, \dots, p_i^+, \dots, 0_n), \quad i = \overline{1, n}.$$

Образуем вектор $\Delta_1^1 = (\delta_1^0, \delta_2^0, \dots, \delta_n^0)$.

2. Определение m индексов, не равных 1 (номеру табулируемой функции) из $n-1$ оставшихся, таких что выполняется условие (8). Пусть, например, искомые индексы – это номера от 2 до $m+1$.

3. Определение величин $\delta_i \in R, i = \overline{2, m+1}$ таких, что

$$E(\Delta_1^1, \sum_{i=2}^{m+1} \Delta_i) = 1, \quad (24)$$

где $\Delta_i \in R^n$ – векторы указанной в условии структуры.

4. Определение величины δ_1^2 :

4.1. Определение величины δ_1^2 такой, что

$$\varphi_1(\delta_1^2) = 2\varphi_1(\delta_1^1), \quad (25)$$

из условия

$$E_1(\overline{\Delta}_1^2, \Delta_1^2) = 1, \quad (26)$$

где

$$\overline{\Delta}_1^2 = (\delta_1^1, \delta_2, \dots, \delta_{m+1}, \delta_{m+2}^0, \dots, \delta_n^0). \quad (27)$$

Докажем, что равенства (26) и (27) выполняются одновременно. Согласно теореме об общем виде m -мерного линейного предиката E существует матрица A размера $m \times m$ и ранга m , такая что

$$\forall x, y \in R^n E(x, y) = D(Ax, Ay), \quad (28)$$

где D – предикат равенства.

Перейдем в равенстве (25) от предиката E_1 к предикату E , пользуясь условиями (22) и (28), а также представлением (23) функции φ :

$$\begin{aligned} 1 &= E_1(\Delta_1^1, \sum_{i=2}^{m+1} \Delta_i) = E(\varphi(\Delta_1^1), \varphi(\sum_{i=2}^{m+1} \Delta_i)) = \\ &= E((\varphi_1(\delta_1^1), O_2, \dots, O_n), (O_1, \varphi_2(\delta_2), \varphi_{m+1}(\delta_{m+1}), \\ &O_{m+2}, \dots, O_n)) = D(\varphi_1(\delta_1^1)A_1, \sum_{i=2}^{m+1} \varphi_i(\delta_i)A_i), \end{aligned}$$

где A_i – i -й столбец матрицы A . То есть мы показали, что

$$\sum_{i=2}^{m+1} \varphi_i(\delta_i)A_i = \varphi_1(\delta_1^1)A_1. \quad (29)$$

Сделаем аналогичный переход в равенстве (27):

$$\begin{aligned} E_1(\overline{\Delta}_1^2, \Delta_1^2) &= E(\varphi(\overline{\Delta}_1^2), \varphi(\Delta_1^2)) = E((\varphi_1(\delta_1^1), \varphi_2(\delta_2), \\ &\dots, \varphi_{m+1}(\delta_{m+1}), O_{m+2}, \dots, O_n), (\varphi_1(\delta_1^2), O_2, \dots, O_n)) = \\ &= D(\varphi_1(\delta_1^1)A_1 + \sum_{i=2}^{m+1} \varphi_i(\delta_i)A_i, \varphi_1(\delta_1^2)A_2) = \\ &= D(2\varphi_1(\delta_1^1)A_1, \varphi_1(\delta_1^2)A_1) = D(2\varphi_1(\delta_1^1), \varphi_1(\delta_1^2)). \end{aligned}$$

Мы воспользовались равенством (29) и тем, что вектор $A_1 \neq 0$. Случай $A_1 = 0$ интереса не представ-

ляет, так как тогда значения δ_1 и $\varphi_1(\delta_1)$ не влияли бы на предикаты E и E_1 .

Так как φ_1 – биективная функция, то существует величина δ_1^2 такая, что выполняется равенство (26), а значит, и равенство (27), что и требовалось доказать.

4.2. Определение величины δ_1^2 такой, что

$$\varphi_1(\delta_1^3) = 3\varphi_1(\delta_1^1), \quad (30)$$

из условия

$$E_1(\overline{\Delta}_1^3, \Delta_1^3) = 1, \quad (31)$$

где

$$D_1^3 = (\delta_1^3, \delta_2^0, \dots, \delta_n^0),$$

$$\overline{\Delta}_1^3 = (\delta_1^2, \delta_2, \dots, \delta_{m+1}, \delta_{m+2}^0, \dots, \delta_n^0).$$

Доказательство эквивалентности равенств (30) и (31) аналогично предыдущему.

Табулирование функции φ_1 продолжается до тех пор, пока не будет пройден весь интересующий нас диапазон. На шаге 4.k определяется величина δ_1^{k+1} такая, что

$$\varphi_1(\delta_1^{k+1}) = (k+1)\varphi_1(\delta_1^1), \quad (32)$$

из условия

$$E_1(\overline{\Delta}_1^{k+1}, \Delta_1^{k+1}) = 1, \quad (33)$$

где

$$\Delta_1^{k+1} = (\delta_1^{k+1}, \delta_2^0, \dots, \delta_n^0),$$

$$\overline{\Delta}_1^{k+1} = (\delta_1^k, \delta_2, \dots, \delta_{m+1}, \delta_{m+2}^0, \dots, \delta_n^0). \quad (34)$$

Общая таблица табулирования имеет вид ($i = \overline{1, n}$):

Таблица 1

$\varphi_i(x_i)$	0	h_i	$2h_i$...	Nh_i
x_i	δ_i^0	δ_i^1	δ_i^2	...	δ_i^N

Табулирование закончено.

Теперь рассмотрим частный случай, когда предикат E одномерен, $m = 1$:

$$\forall x, y \in R^n E(x, y) = D(ax, ay), \quad (35)$$

где $a \in R^n$; $ax = \sum_{i=1}^n a_i x_i$. Хотя бы два компонента

вектора a (например, a_1 и a_2 не равны нулю). Предикат E_1 связан с предикатом E равенством (21) и вектор-функция φ имеет вид (22) и, как и в случае $m < n$, является биекцией.

Тогда процедура табулирования вектор-функции φ следующая:

1. Все аналогично 1-му шагу общего случая $m < n$, за исключением того, что шаг табулирования h одинаков по всем компонентам $\varphi_1, i = \overline{1, n}$.

2. Определение величины δ_2^1 , равновеликой величине δ_1^1 , то есть такой, что

$$\varphi_2(\delta_2^1) = \varphi_1(\delta_1^1). \quad (36)$$

Для этого воспользуемся условием

$$E_1((\delta_1^1, \delta_2^0, \delta_3, \dots, \delta_n), (\delta_1^0, \delta_2^1, \delta_3, \dots, \delta_n)) = 1, \quad (37)$$

где $\delta_3, \dots, \delta_n$ – произвольные фиксированные величины. Действительно, перейдем к предикату E по формуле (21):

$$E((\varphi_1(\delta_1^1), 0_2, \varphi_3(\delta_3), \dots, \varphi_n(\delta_n)), (0_1, \varphi_2(\delta_2^1), \varphi_3(\delta_3), \dots, \varphi_n(\delta_n))) = 1. \quad (38)$$

Прежде чем воспользоваться представлением (20) предиката E , выполним его упрощение. Без ограничения общности можно считать, что если $a_i \neq 0$, то $a_i \neq 1$, $i = \overline{1, n}$. Действительно, если $a_i \neq 0$, то можно положить $\varphi'_i(\xi_i) = 1/a_i \varphi_i(\xi_i)$. Очевидно, что функции φ'_i образуют изоморфизм φ' предикатов E_1 и E' , такой что $\forall \xi, \eta \in R^n$ $E_1(\xi, \eta) = E'(\varphi'_1(\xi_1), \dots, \varphi'_n(\xi_n))$, причем вектор-функция φ' и предикат E' обладают всеми свойствами вектор-функции φ и предиката E_1 соответственно. Значит, будем считать, что $a_i = 1$ или $a_i = 0$, $i = \overline{1, n}$. Пользуясь последним условием, равенством (35) и тем, что $a_1 \neq 0$ и $a_2 \neq 0$, перейдем от (38) к следующему равенству:

$$\begin{aligned} \varphi_1(\delta_1^1) + 0 + \varphi_3(\delta_3) + \dots + \varphi_n(\delta_n) = \\ = 0 + \varphi_2(\delta_2^1) + \varphi_3(\delta_3) + \dots + \varphi_n(\delta_n). \end{aligned}$$

Величины, зависящие от $\delta_3, \dots, \delta_n$, взаимно уничтожаются и остается не зависящее от них равенство

$$\varphi_1(\delta_1^1) = \varphi_2(\delta_2^1), \quad (39)$$

что и требовалось показать.

3. Табулирование функции φ_1 .

3.1. Определение величины δ_1^2 такой, что

$$\varphi_1(\delta_1^2) = 2\varphi_1(\delta_1^1). \quad (40)$$

Для этого достаточно воспользоваться условием

$$E_1((\delta_1^1, \delta_1^2, \delta_3, \dots, \delta_n), (\delta_1^2, \delta_2^0, \delta_3, \dots, \delta_n)) = 1. \quad (41)$$

Действительно, используя равенства (21), (35) и условие $a_i = 1$ или $a_i \neq 0$, $i = \overline{1, n}$, получим:

$$\varphi_1(\delta_1^1) + \varphi_2(\delta_2^1) = \varphi_1(\delta_1^2).$$

Вместе с равенством (36) это дает требуемое соотношение (40).

3.2. Определение величины δ_1^3 такой, что

$$\varphi_1(\delta_1^3) = 3\varphi_1(\delta_1^1). \quad (42)$$

Для этого следует воспользоваться условием

$$E_1((\delta_1^2, \delta_1^3, \delta_3, \dots, \delta_n), (\delta_1^3, \delta_2^0, \delta_3, \dots, \delta_n)) = 1. \quad (43)$$

Доказательство аналогично предыдущему.

Табулирование продолжается до тех пор, пока не будет пройден весь интересующий нас диапазон, то есть до шага 3.N, когда

$$\delta_1^N \geq \delta_1^{p^+}, \delta_1^{N-1} < \delta_1^{p^+}. \quad (44)$$

4. Табулирование функций φ_1 , $i = \overline{2, n}$ таких, что $a_i \neq 0$. Для определения величины δ_i^k такой, что

$$\varphi_i(\delta_i^k) = kh, \quad (45)$$

следует воспользоваться условием

$$E((\delta_1^k, \delta_2, \dots, \delta_{i-1}, \delta_i^0, \delta_{i+1}, \dots, \delta_n), (\delta_1^0, \delta_2, \dots, \delta_{i-1}, \delta_i^k, \delta_{i+1}, \dots, \delta_n)) = 1, \quad (46)$$

$i = \overline{2, n}$, $k = \overline{1, N}$.

Доказательство такое же, как и в п. 3.1. Если $a_i = 0$, то значения δ_i и φ_i не влияют на предикаты E_1 и E . Общая таблица табулирования имеет вид:

Таблица 2

$\varphi_i(\xi_i)$	0	h	$2h$...	Nh
ξ_i	δ_i^0	δ_i^1	δ_i^2	...	δ_i^N

Табулирование закончено.

Теперь, имея результаты табулирования, можно проверить выполнение условий (21) и (22). Если предикат E обладает свойствами m -мерного линейного предиката, то эти условия выполняются. Условие (23) также можно проверить только после табулирования, используя известные предикаты E_1 и E . На практике достаточно найти хотя бы один m -мерный базис предиката E .

Выводы

Рассмотрены возможности применения компараторной идентификации при изучении социально-экономических процессов. Предложены компараторные модели процесса оценки человеком места работы в виде предиката эквивалентности и предиката порядка. Разработан функциональный метод шкалирования входных сигналов нелинейного объекта компараторной идентификации. Разработана процедура табулирования входных сигналов широкого класса нелинейных объектов компараторной идентификации, как одно-, так и многомерных. Сформулированы свойства нелинейного объекта компараторной идентификации, достаточные для его структурной идентификации методами, разработанными для линейных объектов.

Список литературы: 1. Бондаренко, М.Ф. Об общей теории компараторной идентификации [Текст] / М.Ф. Бондаренко, С.Ю. Шабанов-Кушнаренко, Ю.П. Шабанов-Кушнаренко // Бионика интеллекта: науч.-техн. журнал. – 2008. – № 2(68). – С. 13-22. 2. Бондаренко, М.Ф. Идентификация объектов, описываемых числовыми системами [Текст] / М.Ф. Бондаренко, С.Ю. Шабанов-Кушнаренко, Ю.П. Шабанов-Кушнаренко // Бионика интеллекта: науч.-техн. журнал. – 2008. – № 2(68). – С. 23-31. 3. Бондаренко, М.Ф. Разработка систем условий,

обеспечивающих существование линейного конечно-мерного оператора [Текст] / М.Ф. Бондаренко, С.Ю. Шабанов-Кушнарченко, Ю.П. Шабанов-Кушнарченко // Бионика интеллекта: науч.-техн. журнал. – 2009. – № 1 (70). – С. 11-17. 4. Бондаренко, М.Ф. Практические приложения компараторной идентификации линейных конечномерных объектов [Текст] / М.Ф. Бондаренко, С.Ю. Шабанов-Кушнарченко, Ю.П. Шабанов-Кушнарченко // Бионика интеллекта: науч.-техн. журнал. – 2009. – № 2 (71). – С. 5-12.

Поступила в редколлегию 17.09.2009

УДК 519.7

Методи ідентифікації механізму суб'єктивних соціально-економічних оцінок / М.Ф. Бондаренко, С.Ю. Шабанов-Кушнарченко, Ю.П. Шабанов-Кушнарченко // Біоніка інтелекту: наук.-техн. журнал. – 2009. – № 2 (71). – С. 24-30.

У статті розвивається теорія компараторної параметричної ідентифікації. Використовується модель у вигляді предиката порядку. Розглядаються методи ідентифікації суб'єктивних оцінок соціально-економічних процесів – функціональний метод шкалювання вхідних сигналів та метод табулювання ізоморфізмів лінійного предиката.

Бібліогр.: 4 найм.

UDC 519.7

Methods of the subjective social and economic estimations mechanism identification / M.F. Bondarenko, S.Yu. Shabanov-Kushnarenko, Yu.P. Shabanov-Kushnarenko // Bionika intellekta: Sci. Mag. – 2009. – № 2 (71). – P. 24-30.

In article the comparator parametrical identification theory is develops. The model in the form of an order predicate is used. Methods of identification of value judgment of social and economic processes – the entrance signals functional scaling method and the isomorphism of a linear predicate tabulation method are considered.

Ref.: 4 items.

УДК 519.7:007.52; 519.711.3



ДОСЛІДЖЕННЯ РОЗМІРНОСТІ ПРЕДМЕТНОГО ПРОСТОРУ В ЗАДАЧАХ МОДЕЛЮВАННЯ ОБ'ЄКТІВ У ВИГЛЯДІ РЕЛЯЦІЙНИХ МЕРЕЖ

І.Д. Вечірська

ХНУРЕ, м.Харків, Україна, ira_se@list.ru

В статті досліджено розмірність предметного простору в задачах, засобом розв'язання яких є реляційні мережі. Досліджені особливості побудови математичної моделі об'єкта у вигляді реляційної мережі. Наведено статистичні дані для степеня лінійного логічного перетворення та проведено їх аналіз в залежності від розмірності матриці ядра перетворення та її загального вигляду.

ЛІНІЙНЕ ЛОГІЧНЕ ПЕРЕТВОРЕННЯ, ОБЛАСТЬ ВИЗНАЧЕННЯ, РЕЛЯЦІЙНА МЕРЕЖА, ЯДРО ЛІНІЙНОГО ЛОГІЧНОГО ПЕРЕТВОРЕННЯ

Вступ

Сьогодення показує неупинний зріст темпів інформатизації суспільства. Наразі виникає нагальна потреба збереження та використання інформації в зручному вигляді. А оскільки для людини найзручнішою формою подання інформаційних повідомлень є природномовна, то особливо актуальним являється питання обробки саме природномовної інформації. Крім потреб щоденного спілкування людини, така необхідність обумовлена гострою потребою формалізувати та систематизувати накопичені протягом багатьох років знання, а головне зробити це ергономічно для подальшої обробки користувачем [1].

Формалізацією мовного феномену займаються вже не один десяток років. У цій галузі було запропоновано найрізноманітніші способи подання знань: від відомих традиційних засобів формальних граматики, логічних числень, лексикографічних числень і так далі до доволі несподіваного застосування принципів квантової механіки [2, 3]. Одним з універсальних засобів подання знань є реляційні мережі [4]. Універсальність у даному випадку обґрунтовано можливістю реалізації саме відношень, а не функцій, що дозволяє в свою чергу розв'язувати засобами однієї і тієї ж схеми прямі та обернені задачі.

Реляційна мережа складається з гілок та вузлів. Вузлам відповідають предметні змінні, а гілкам – відношення, що їх пов'язують. На мові алгебри предикатів, у межах якої й моделюють реляційні мережі, ці відношення записують у вигляді бінарного предикату. Тому для ефективного розв'язання різноманітних задач реляційною мережею необхідно вирішити питання подання багатомісцевого відношення у вигляді композиції бінарних. Зазвичай це питання вирішується за рахунок введення допоміжної змінної (допоміжною змінною в загальному випадку може бути порядковий номер), проте застосування такого способу бінаризації відношення не завжди є доцільним. Все залежить від структури самого вихідного відношення та областей визначення предметних змінних, які в нього входять.

В процесі розв'язання задачі реляційною мережею на деякі її вузли подають відомі знання, а мережа формує з них знання для інших вузлів. Мережа закінчує свою роботу, коли в усіх вузлах мережі два такти посліпль не змінюються значення. У ході роботи реляційної мережі по усім її гілкам інформація рухається у двох напрямках: прямому та зворотньому. Обробку знань на гілках реляційної мережі виконують лінійні логічні перетворення. Таким чином, метою даної статті є дослідження розмірності предметного простору в задачах моделювання реляційних мереж за рахунок вивчення властивостей лінійних логічних перетворень та закономірностей їх обчислення.

1. Математичний опис об'єкта

Для побудови математичної моделі природномовного об'єкта у вигляді реляційної мережі необхідно виявити усі предметні змінні та записати відношення, що їх зв'язує. Так, наприклад, початкове відношення, яке відображує зміну повних неприсвійних прикметників російської мови, містить 13 змінних і має вигляд [5]:

$$P(x_1, x_2, x_3, x_4, x_5, y_1, y_2, y_3, z, z_0, z_1, z_2, z_3).$$

Для того щоб бінаризувати початкове відношення, необхідно ввести допоміжні, так звані проміжні змінні. Для моделі словозміни повних неприсвійних прикметників проміжними змінними є змінні, що позначають контекст слова, тип словозміни, ліву та праву частини закінчення прикметників. Отже, математична модель у даному прикладі характеризується системою бінарних відношень у вигляді композиції 17-ти відповідних предикатів [5]:

$$\begin{aligned} P(x_1, x_2, x_3, x_4, x_5, r, s, y_1, y_2, y_3, z, z_0, z_1, z_2, z_3) = \\ = P_1(x_1, r) \wedge P_2(x_2, r) \wedge P_3(x_3, r) \wedge P_4(x_4, r) \wedge P_5(x_5, r) \wedge \\ \wedge P_6(z_l, r) \wedge P_7(z_{\text{п}}, r) \wedge P_8(z_l, z) \wedge P_9(z_{\text{п}}, z) \wedge P_{10}(z_l, z_0) \wedge \\ \wedge P_{11}(z_{\text{п}}, z_1) \wedge P_{12}(z_{\text{п}}, z_2) \wedge P_{13}(z_{\text{п}}, z_3) \wedge P_{14}(s, z_l) \wedge \\ \wedge P_{15}(y_1, s) \wedge P_{16}(y_2, s) \wedge P_{17}(y_3, s). \end{aligned}$$

Слід відзначити, що в розробці математичного опису об'єкта у вигляді реляційної мережі основна проблема полягає якраз у формулюванні загальних правил вибору проміжних змінних. Одним з шляхів розв'язання цієї задачі є виконання наступних дій: подати початкове відношення у вигляді ДДНФ, потім зібрати усі прості імпліканти у логічний добуток диз'юнкції змінних і далі за таблицями істинності шукати максимальні області для окремих змінних. Виходячи саме з таких міркувань, і обирають проміжні змінні.

Множиною вузлів реляційної мережі є множина усіх предметних змінних моделі, що описується. Множина гілок – це множина усіх бінарних предикатів, композиція яких складає вихідний предикат. На основі такого математичного опису можна розв'язувати клас задач, які щохвилини розв'язують люди в процесі спілкування, аналізуючи і синтезуючи повідомлення природною мовою. Для розв'язання таких задач вихідними даними для рівняння, яке описує математичну модель, візьмемо значення будь-яких наявних в ньому змінних-ознак. Зазначимо, що кожній змінній може відповідати якась підмножина області визначення. Тобто значення змінної не обов'язково задавати однозначно. Підмножина області визначення змінної називається знанням про значення змінної. Так, під час роботи зі введеними даними та рівнянням моделі реляційна мережа визначає знання про значення будь-яких змінних.

Таким чином, усі подібні задачі можна звести до того, щоб на основі рівняння моделі та вказаних підмножин значень деяких з її змінних визначити підмножини значень інших змінних цього рівняння.

2. Формальна постановка задачі

Опишемо сформульовану задачу у термінах алгебри предикатів. У загальному випадку результатом формального опису будь-якого об'єкта мовою алгебри предикатів завжди являється деякий предикат $H(x_1, x_2, \dots, x_n)$. Він позначає певне відношення H , яке є множиною значень усіх наборів предметів (x_1, x_2, \dots, x_n) , для яких виконується рівняння

$$H(x_1, x_2, \dots, x_n) = 1. \quad (1)$$

При цьому $H \subseteq S$, $(x_1, x_2, \dots, x_n) \in S$.

Вихідними даними такої задачі являються підмножини значень деяких із змінних рівняння (1):

$$x_{j_1} \in C_{j_1}, x_{j_2} \in C_{j_2}, \dots, x_{j_k} \in C_{j_k}, \quad (2)$$

де $1 \leq j \leq n$; $k \leq n$. Якщо $k = n$, для кожної змінної вказують множину її області визначення (стосовно моделі словозміни прикметників така задача навряд чи може виникнути). У загальному випадку це можуть бути довільні підмножини області визначення однієї або декількох змінних. Формула (2) виражає сукупність обмежень на значення вказаних змінних. Таким

чином, на основі зв'язку між усіма змінними, який виражається рівнянням (1), необхідно визначити, як вихідні дані обмежують значення інших змінних.

У вихідне обмеження (2) можна ввести довільні змінні, тобто існує можливість взаємного впливу змінних $x_{j_1}, x_{j_2}, \dots, x_{j_k}$, оскільки їх пов'язує рівняння (1). Тому задача полягає у визначенні множини можливих значень, які може приймати кожна із змінних рівняння, якщо значення деяких з них обмежені умовами (2). Для кожної змінної x_i позначимо таку множину як X_i^* , $(i = \overline{1, n})$. Таким чином, на основі рівняння (1) необхідно визначити множини $X_1^*, X_2^*, \dots, X_n^*$ за умови задання множин $X_{j_1}, X_{j_2}, \dots, X_{j_k}$. Виникає питання, як підставити множину значень однієї змінної у рівняння (1). Нехай, наприклад, $x_i \in C_i$ і $C_i(x_i : A_i)$ – предикат, який відповідає множині C_i , при цьому завжди виконується включення $C_i \subseteq A_i$. Для того щоб підставити множину значень C_i змінної x_i в рівняння, необхідно його ліву частину, тобто предикат $H(x_1, x_2, \dots, x_n)$, логічно домножити на предикат $C_i(x_i)$ а потім виключити змінну x_i з рівняння за допомогою квантора існування. В результаті отримаємо наступне рівняння: $\exists x_i (H(x_1, x_2, \dots, x_n) \wedge C_i(x_i)) = 1$. Якщо одночасно підставити у вираз значення декількох змінних, отримаємо своєрідне узагальнення цього способу. Нехай умови задачі задано формулою (2). Тоді в результаті підстановки усіх значень змінних у вихідне рівняння та після тотожних перетворень отримаємо наступне рівняння:

$$\begin{aligned} & \exists x_{j_1} \exists x_{j_2} \dots \exists x_{j_k} (H(x_1, x_2, \dots, x_n) \wedge \\ & \wedge C(x_{j_1}) C(x_{j_2}) \wedge \dots \wedge C(x_{j_k})) = 1. \end{aligned} \quad (3)$$

Зауважимо, що в останньому рівнянні присутні $(n-k)$ змінних.

Таким чином, усі можливі задачі вказаного типу можна звести до того, щоб на основі рівняння моделі та вказаних підмножин значень деяких її змінних визначити підмножини значень інших змінних цього рівняння.

Як уже було вказано, поняття «реляційна мережа» у загальному розумінні можна звести до поняття бінарної мережі за рахунок того, що знання обробляються в гілках між кожною парою вузлів мережі паралельно. Таким чином, рівняння (3) можна переписати, якщо замінити вихідне відношення $H(x_1, x_2, \dots, x_n)$ на композицію відповідних йому бінарних відношень. Розглянемо далі перехід до лінійних логічних перетворень.

3. Метод знаходження степеня лінійного логічного перетворення

Лінійним логічним перетворенням [6, 7] $L(P) = Q$ називається оператор L

$$Q(y) = \exists x \in A (K(x, y) \wedge P(x)), \quad (4)$$

який перетворює унарні предикати $P(x)$, що задані на множині A , в унарні предикати $Q(y)$, які задаються на множині B . У виразі (4) бінарний предикат $K(x,y)$ фіксований і заданий на декартовому добутку $A \times B$. Цей предикат називають ядром оператора L ; квантор існування в виразі означає логічну суму. Лінійне логічне перетворення задає перетворення однієї підмножини значень змінної x з областю визначення A , яку задано предикатом $P(x)$, у відповідну підмножину значень змінної y з областю визначення B , що задана предикатом $Q(y)$. Отже, обробку знань у гілках реляційної мережі здійснюють лінійні логічні перетворення.

В роботах [7], [8] було проведено дослідження дій над лінійними логічними перетвореннями, а саме знаходження степеня лінійного логічного перетворення (рис. 1). Виведено формулу для знаходження n -ого степеня лінійних логічних перетворень у прямому і зворотньому напрямку:

$$Q^{(n)}(y) = \bigwedge_{i=1}^n K_i Q(y), \text{ де } K_i = K = K(x,y)K(y,x),$$

$$P^{(n)}(x) = \bigwedge_{i=1}^n K'_i P(x), \text{ де } K'_i = K' = K(y,x)K(x,y).$$

Таким чином, розроблений метод знаходження степеня лінійного логічного перетворення $Q^{(n)}(y)$ можна розбити на наступні етапи. Спочатку необхідно знайти матрицю K , яка є суперпозицією ядер лінійних логічних перетворень з $P(x)$ в $Q(y)$ і, відповідно, з $Q(y)$ в $P'(x)$: $K = K(x,y)K(y,x)$.

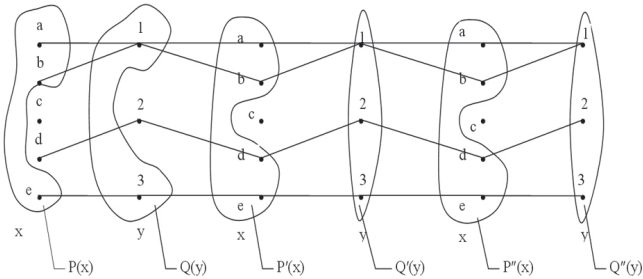


Рис. 1. Знаходження степеня лінійного логічного перетворення

Нехай ядро лінійного перетворення можна представити виразом $K(x,y) = \bigwedge_{j=1, \overline{1,n}}^{i=1, \overline{1,m}} a_{ij}$. Тоді матриця ядра зворотнього перетворення має вигляд $K(x,y) = \bigwedge_{j=1, \overline{1,n}}^{i=1, \overline{1,m}} a_{ji}$. А елементи матриці K можна виразити наступним чином:

$$b_{11} = a_{11} \vee a_{12} \vee \dots \vee a_{1,n-1} \vee a_{1n};$$

$$b_{12} = a_{11}a_{21} \vee a_{12}a_{22} \vee \dots \vee a_{1,n-1}a_{2,n-1} \vee a_{1n}a_{2n};$$

.....

$$b_{1,m-1} = a_{11}a_{m-1,1} \vee a_{12}a_{m-1,2} \vee \dots \vee a_{1,n-1}a_{m-1,n-1} \vee a_{1n}a_{m-1,n};$$

$$b_{1m} = a_{11}a_{m1} \vee a_{12}a_{m2} \vee \dots \vee a_{1,n-1}a_{m,n-1} \vee a_{1n}a_{mn};$$

$$b_{21} = a_{21}a_{11} \vee a_{22}a_{12} \vee \dots \vee a_{2,n-1}a_{1,n-1} \vee a_{2n}a_{1n};$$

$$b_{22} = a_{21} \vee a_{22} \vee \dots \vee a_{2,n-1} \vee a_{2n};$$

.....

$$b_{2,m-1} = a_{21}a_{m-1,1} \vee a_{22}a_{m-1,2} \vee \dots \vee a_{2,n-1}a_{m-1,n-1} \vee a_{2n}a_{m-1,n};$$

$$b_{2m} = a_{21}a_{m1} \vee a_{22}a_{m2} \vee \dots \vee a_{2,n-1}a_{m,n-1} \vee a_{2n}a_{mn};$$

$$b_{m-1,1} = a_{m-1,1}a_{11} \vee a_{m-1,2}a_{12} \vee \dots \vee a_{m-1,n-1}a_{1,n-1} \vee a_{m-1,n}a_{1n};$$

$$b_{m-1,2} = a_{m-1,1}a_{21} \vee a_{m-1,2}a_{22} \vee \dots \vee a_{m-1,n-1}a_{2,n-1} \vee a_{m-1,n}a_{2n};$$

.....

$$b_{m-1,m-1} = a_{m-1,1} \vee a_{m-1,2} \vee \dots \vee a_{m-1,n-1} \vee a_{m-1,n};$$

$$b_{m-1,m} = a_{m-1,1}a_{m1} \vee a_{m-1,2}a_{m2} \vee \dots \vee a_{m-1,n-1}a_{m,n-1} \vee a_{m-1,n}a_{mn};$$

$$b_{m1} = a_{m1}a_{11} \vee a_{m2}a_{12} \vee \dots \vee a_{m,n-1}a_{1,n-1} \vee a_{mn}a_{1n};$$

$$b_{m2} = a_{m1}a_{21} \vee a_{m2}a_{22} \vee \dots \vee a_{m,n-1}a_{2,n-1} \vee a_{mn}a_{2n};$$

.....

$$b_{m,m-1} = a_{m1}a_{m-1,1} \vee a_{m2}a_{m-1,2} \vee \dots \vee a_{m,n-1}a_{m-1,n-1} \vee a_{mn}a_{m-1,n};$$

$$b_{mm} = a_{m1} \vee a_{m2} \vee \dots \vee a_{m,n-1} \vee a_{mn}.$$

На наступному етапі необхідно знайти кон'юнкцію усіх n суперпозицій ядра лінійного логічного перетворення та вхідного вектора.

Таким чином, можна зробити висновок, що n -ий степінь лінійного логічного перетворення ($n \geq 1$) залежить від виду матриці K . А матриця K у свою чергу залежить тільки від області визначення змінної x і не залежить від області визначення змінної y . Звідси випливає, що крок, на якому степінь лінійного логічного перетворення в подальших діях не змінюється, безпосередньо залежить від розмірності області визначення змінної x .

4. Дослідження розмірності предметного простору

Було встановлено, що матриця K залежить тільки від області визначення вихідної змінної. Наведемо далі результати таких досліджень.

Було проведено аналіз матриць K різного розміру (2×2 , 3×3 , 4×4 та 5×5). Всього 33 620 496 матриць.

Результати наведено в табл. 1, де N – це номер кроку, на якому лінійне логічне перетворення стабілізується, тобто матриця ядра лінійного логічного перетворення не змінює своїх елементів при подальших перемноженнях.

Таблиця 1

Розмір матриці \ N	2x2	3x3	4x4	5x5
1-й крок	16	422	34324	11051342
2-й крок	0	90	29340	20645490
3-й крок	0	0	1872	1800000
4-й крок	0	0	0	57600
5-й крок	0	0	0	0
Всього	16	512	65 536	33554432

З таблиці видно, що кількість кроків, за які знаходиться кінцевий результат, прямо залежить від розмірності матриці K . Крім того, для матриць розміру $m \times m$ кількість кроків не перевищує m .

Слід відмітити, що у багатьох випадках при знаходженні матриці K як добутку прямої матриці $K(x,y)$ на транспоновану $K(y,x)$, за різних вихідних елементів матриць було отримано однакові K . У деяких випадках було отримано матрицю з нульовими рядком та стовпчиком. Таким чином, розмірність матриці автоматично ставала на порядок нижче. Тому не дивлячись на доволі вагомні цифри статистики, з проведення такого роду аналізу можна зробити висновок, що всього декілька видів матриць розміру $m \times m$ використовують m кроків для знаходження остаточного розв'язку. Так, для матриць розмірності 3×3 всі 3 кроки (останній третій крок було зроблено для перевірки вищеописаного критерію) для знаходження остаточного розв'язку використовували лише матриці трьох типів:

$$\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}.$$

В перетвореннях матриці K два кроки (результат було знайдено вже на першому кроці) використовували матриці шести типів:

- 1) одиничні матриці;
- 2) матриці з нульовими рядком та стовпчиком;

$$3) \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix};$$

$$4) \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix};$$

$$5) \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix};$$

$$6) \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Далі у табл. 2 запишемо кількість матриць, які співпадають.

5. Аналіз та перспективи проведених досліджень

Проведений аналіз статистичних досліджень відіграє важливу роль у теорії реляційних мереж. З наведених вище фактів можна зробити висновок, що швидкість роботи реляційної мережі залежить лише від розмірності підмножини вихідної змінної (чи змінних). Тобто в загальному випадку, чим більше значень предметної змінної подається на вхід мережі як початкові дані, тим більше кроків буде зроблено мережею до знаходження кінцевого результату.

Таблиця 2

Матриця	Кількість повторень	Матриця	Кількість повторень
$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$	1	$\begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix}$	24
$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	6	$\begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	24
$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	7	$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix}$	37
$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$	7	$\begin{pmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix}$	37
$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$	7	$\begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$	37
$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	12	$\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$	175
$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	12	$\begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix}$	30
$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$	12	$\begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$	30
$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix}$	24	$\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix}$	30

Раніше було доведено твердження [8] про те, що якщо при знаходженні степеня лінійного логічного перетворення на двох послідовних кроках значення перетворення повторюється, то це значення буде повторюватись також і на наступних кроках. Тобто якщо при знаходженні n -ого степеня лінійного логічного перетворення було отримано однакові результати на n -ому та $n-1$ -ому кроках, то цей результат отримаємо також і на наступних $n+1$ -ому, $n+2$ -ому і так далі кроках. Тоді таке лінійне перетворення і є шуканим. Таким чином, метод знаходження n -ого лінійного логічного перетворення дає нам критерій закінчення роботи логічної мережі і обґрунтовує його.

Проведені дослідження дозволяють визначити, що кількість ітерацій, які виконує логічна мережа до знаходження остаточного результату, не пере-

вищує розмірності підмножини області визначення вхідної змінної.

Аналіз проведено для випадку, коли на вхід подаються значення однієї змінної. Але необхідно вивчити взаємовплив даних, коли параметри задають одночасно на декількох полюсах. Крім того, необхідно оцінити розмірність задачі за час роботи всієї реляційної мережі, враховуючи її роботу в кожному вузлі в прямому та зворотньому напрямку. У подальших дослідженнях також є сенс зупинитись на самому вигляді елементів матриць, що формуються на кожному кроці, вивчити, як залежить кількість результуючих матриць від загального вигляду матриці та як залежить кількість матриць, які співпадають, від їх загального вигляду (тобто кількості та розташування нулів та одиниць матриці) на кожному кроці.

Висновки

Було досліджено розмірність предметного простору в задачах моделювання об'єктів засобами реляційних мереж. Наведено математичний опис об'єкта у вигляді реляційної мережі та метод знаходження степеня лінійного логічного перетворення. Статистику за результатами дослідження розмірності предметного простору в задачах, які доцільно розв'язувати за допомогою логічних мереж, розглянуто для областей визначення, які складають 2, 3, 4 та 5 значень. Детально проведено аналіз результатів дослідження ядра n -ого лінійного логічного перетворення, яке можна представити у вигляді матриці розміром 3×3 . А саме визначена необхідна кількість кроків для стабілізації лінійного логічного перетворення. Крім того, визначені 6 типів матриць, для яких результат знаходиться на першому кроці, та 3 типи матриць, для яких результат знаходиться на другому кроці. Також наведено статистичні дані по матрицям ядра лінійного логічного перетворення, що повторюються.

Список літератури: 1. *Бондаренко, М.Ф.* Основи теорії синтезу надшвидкодуючих структур мовних систем штучного інтелекту [Текст] / М.Ф. Бондаренко, З.Д. Коноплянко, Г.Г. Четвериков. – К.: ІЗМН, 1997. – 264 с. 2. *Широков, В.А.* Очерк основных принципов квантовой лингвистики [Текст] / В.А. Широков // Бионика интеллекта: науч.-техн. журнал. – 2007. – № 1(66). – С. 25-32. 3. *Широков, В.А.* Інформаційна теорія лексикографічних систем [Текст] / В.А. Широков. – К.: Довіра, 1998. – 331 с.

4. *Бондаренко, М.Ф.* О реляционных сетях как средство моделирования естественных языковых структур [Текст] / М.Ф. Бондаренко, И.Д. Вечирская, Г.Г. Четвериков, В.В. Токарев / Доклады Междунар. науч. конф. “Горизонты прикладной лингвистики и лингвистических технологий” (MegaLing’2008). – 22-28 сентября 2008, Украина, Крым, Партенит / редкол. В.А. Широков, С.С. Дикарева. – Симферополь: Изд-во „ДИАЙПИ”, 2008. – 372 с. 5. *Бондаренко, М.Ф.* Модели языка [Текст] / М.Ф. Бондаренко, В.А. Чикина, Ю.П. Шабанов–Кушнаренко // Бионика интеллекта: науч.-техн. журнал. – 2004. – № 1(61). – С. 27-37. 6. *Бондаренко, М.Ф.* О модифицированных категориях [Текст] / М.Ф. Бондаренко, З.В. Дударь, А.А. Иванилов, В.В. Маникин, Ю.П. Шабанов–Кушнаренко // Радиоэлектроника и информатика. – 2005. – № 1. – С. 87-99. 7. *Вечирская, И.Д.* О методе вычисления линейных логических преобразований [Текст] / И.Д. Вечирская // Бионика интеллекта: науч.-техн. журнал. – 2007. – № 2 (67). – С. 65 – 68. 8. *Вечирская, И.Д.* О методе нахождения n -ого линейного логического преобразования [Текст] / И.Д. Вечирская, Ю.П. Шабанов–Кушнаренко // Искусственный интеллект. – Донецк: Институт проблем искусственного интеллекта. – 2007. – № 3. – С. 382-389.

Поступила в редколлегию 08.09.2009

УДК 519.7:007.52; 519.711.3

Исследование размерности предметного пространства в задачах моделирования объектов в виде реляционных сетей / И.Д. Вечирская // Бионика интеллекта: науч.-техн. журнал. – 2009. – № 2 (71). – С. 31-35.

Исследована размерность предметного пространства в задачах моделирования объектов средствами реляционных сетей. Проведен статистический анализ общего вида ядер n -ого линейного логического преобразования от количества шагов, необходимых для стабилизации вычислений.

Табл. 2. Библиогр.: 8 назв.

UDC 519.7:007.52; 519.711.3

The research of space dimension in task of object modelling by relation nets / I.D. Vechirskaya // Bionics of Intelligence: Sci. Mag. – 2009. – № 2 (71). – P. 31-35.

Dimension of subject space in problems of object modelling is investigated by relation nets. The statistical analysis of general view of kernels of n -th linear logic transformation from quantity of the steps are requested for stabilisation of calculations is carried out.

The space dimension in task of object modelling by relation nets is investigated. The overview of kern of linear logical transformation and quantity of steps, which are required stabilization of computing are analyzed.

Tab. 2. Ref.: 8 items.

УДК 519.767



ИСПОЛЬЗОВАНИЕ МАТЕМАТИЧЕСКОГО АППАРАТА ПРЕДИКАТНЫХ КАТЕГОРИЙ ДЛЯ МОДЕЛИРОВАНИЯ СЕМАНТИКИ СВЕРХФРАЗОВЫХ ЕДИНСТВ

Н.Ф. Хайрова

НТУ «ХПИ», г. Харьков, Украина, nikhayv@vlink.kharkov.ua

Проведен анализ задач лингвистического процессора современных систем машинного перевода. Показано использование математического аппарата предикатных и модифицированных категорий для моделирования отношений в универсуме всевозможных элементов языковой системы. Разработана модель снятия лексической омонимии многозначных существительных сверхфразового единства.

ТЕОРИЯ ПРЕДИКАТНЫХ КАТЕГОРИЙ, СВЕРХФРАЗОВОЕ ЕДИНСТВО, МАШИННЫЙ ПЕРЕВОД, СЕМАНТИЧЕСКИЙ АНАЛИЗ

Введение

За последние пять-семь лет произошел качественный скачок в развитии систем автоматического перевода. Практически все современные системы трансферного типа позволяют получать перевод без грамматических (морфологических и контекстно-синтаксических) ошибок. Это связано с тем, что на поверхностных уровнях моделирования естественного языка — морфологическом и синтаксическом достигнуто достаточно много практических результатов.

Этап же семантического анализа по-прежнему остается одним из актуальных направлений исследования как в прикладной лингвистике, так и в области исследований искусственного интеллекта.

Как известно, под семантическим анализом в общем случае понимается представление значения входного текста в терминах некоторого формального языка, «понятного» ЭВМ. В большинстве современных систем машинного перевода задача семантического анализа сводится к выбору правильного значения переводного эквивалента многозначного слова.

1. Постановка задачи исследования

На мировом рынке сегодня представлено около тысячи коммерческих систем автоматического перевода, большинство из которых относятся к системам трансферного типа. В такого рода системах обычно для снятия семантической омонимии используется метод семантических фильтров. Этот легко реализуемый способ выбора переводного эквивалента многозначного слова, заключающийся в привлечении к переводу знаний о тематике переводимого текста. Его реализация обычно заключается в подключении заранее определенных пользователем предметных словарей. Использование данного метода на узкоспециализированных текстах (например, руководства по использованию) позволяет получить перевод довольно высокого качества. Но так как тексты редко бывают узкоспециализированными и раскрывают, как правило, несколько

микротем, подключение предметных словарей ко всему тексту нередко не только не улучшают качество перевода, но и несколько ухудшают его.

Можно показать, что привлечение к переводу знаний микротемы, раскрываемой в каждом отдельном сверхфразовом единстве, увеличивает вероятность правильного выбора переводных эквивалентов многозначного слова.

Для реализации данной задачи, для моделирования отношений в универсуме всевозможных элементов языковой системы предлагается использовать положения предикатной и модифицированной теории категорий.

2. Описание используемого метода

Обрабатываемые лингвистическим процессором объекты являются дискретными, конечными и детерминированными, что позволяет использовать при их обработке теорию категорий, включающую понятия предикатных категорий и модифицированных категорий [1, 2]. На вход системы подаются объекты различного уровня языковой системы x_1, x_2, \dots, x_n (морфемы, словоформы, словосочетания, семы, и так далее). Объекты, передающие информацию, берутся из конечных множеств X_1, X_2, \dots, X_n (множество морфем языка, множество словоформ словаря и так далее) причем $x_1 \in X_1, x_2 \in X_2, \dots, x_n \in X_n$. В результате работы лингвистического процессора на каждом уровне обработки языковой системы на выход поступает определенное множество объектов y_1, y_2, \dots, y_n , под которыми мы будем понимать морфемы, словоформы, словосочетания на языке перевода. Причем $y_1 \in Y_1, y_2 \in Y_2, \dots, y_n \in Y_n$. Понятно, что объекты y_1, y_2, \dots, y_n прямо зависят от объектов x_1, x_2, \dots, x_n . Это обозначает, что существует класс морфизмов, отображающий совокупность однотипных объектов друг на друга (морфизмов). Морфизмы представляют собой абстрактную направленную связь между объектами.

В [3] дается определение объектной категории C как совокупности однотипных математичес-

ких объектов X, Y (множеств, пространств, групп и так далее), для каждой пары из которых задано множество морфизмов (или стрелок) $Hom_C(X, Y)$, причем каждому морфизму соответствуют единственные X и Y . Категория C включает правило композиции морфизмов: для пары морфизмов $f \in Hom(X, Y)$ и $g \in Hom(Y, Z)$ определена композиция $f \circ g \in Hom(X, Z)$ и задает для каждого объекта X тождественный морфизм $id_x \in Hom(X, X)$.

В объектной категории множество объектов категории C обозначается $Ob C$. Если X является объектом категории C (C -object), то записывают $X \in Ob C$. Если f морфизм из объекта в объект Y : $X \xrightarrow{f} Y$, то объект X называют началом морфизма f , а объект Y — его концом. Так как для каждого морфизма $f \in Hom(X, Y)$ существует единственная пара объектов X, Y , такая что $X, Y \in Ob C$ и $f \in Hom_C(X, Y)$, то можно интерпретировать морфизмы как некоторые функции, областью определения которых X , а область значений — Y .

В качестве однотипных математических объектов категории будем использовать переменные предикаты, заданные на множествах X и Y . Предикатная категория $Pred$ [4] задается на некотором универсуме U . Из элементов универсума образуются подмножества X, Y, Z, \dots категории $Pred$, связанные с конкретной задачей аналитико-синтетической обработки лингвистического процессора системы. В роли множества объектов $Ob Pred$ используется система всех подмножеств универсума U . Предикат $P(x)$, заданный на множестве X , рассматривается как экземпляр объекта X . А предикат $Q(y)$, задан на множестве Y , рассматривается как экземпляр объекта Y . Морфизм $f \in Hom(X, Y)$, преобразующий экземпляры объекта X в экземпляры объекта Y , может быть представлен в виде линейного логического оператора $F_f(P) = Q$. Каждый такой оператор преобразует одноместные предикаты P в одноместные предикаты Q :

$$\exists x \in X (C_f(x, y) P(x)) = Q(y) \quad (1)$$

Предикат $C_f(x, y)$ задан на множестве $X \times Y$ и полностью определяет вид преобразования (1). В категории $Pred$ каждому морфизму $f \in Pred$ взаимно однозначно соответствует предикат $C_f(x, y)$ преобразования (1). Каждый морфизм категории $Pred$ можно задать, указав соответствующий ему предикат $C_f(x, y)$ на множестве $X \times Y$. Множество $Hom_{Pred}(X, Y)$ представляет собой совокупность преобразований вида (1) со всевозможными предикатами $C_f(x, y)$, заданными на всевозможных декартовых произведениях $X \times Y$ множеств $X, Y \subseteq U$.

3. Описание модели

В современных системах машинного перевода используется глубинный лингвистический процессор, задача которого на этапе семантической

обработки правильно определить переводной эквивалент многозначного слова [5].

Под универсумом элементов U будем понимать все возможные элементы языковой системы, используемые на различных этапах обработки глубинного лингвистического процессора (словоформы, морфемы, семы, словарные статьи переводных словарей и так далее.). Предикаты, определенные на декартовых произведениях множества объектов $Ob Pred$, характеризуют этапы аналитико-синтетической обработки естественно-языкового текста, поступающего на вход системы.

Под множеством объектов X будем понимать множество словоформ существительных N сверхфразового единства входного текста. Множество N , которое мы будем рассматривать, представляет собой некоторую, достаточно четко очерченную, совокупность существительных $N = \{x_i\}$, $1 \leq i \leq n$, полученную в результате обработки морфологических структур словоформ, выявленных на предыдущих этапах работы лингвистического процессора [6]. Например результатом графемного и морфологического анализа английской фразы: *after selling some of the surplus blood to hospitals* будет выделение следующих морфологических структур существительных:

6, $\{<(\text{surplus}), S, SG>\}$,

7, $\{<(\text{blood}) S, SG>\}$, ...

9, $\{<(\text{hospital}), S, PL>\}$, где 6, 7, 9 — порядковый номер словоформы во фразе; S — часть речи (существительное); SG, PL — соответственно единственное и множественное число.

Предикат $P(x)$, заданный на множестве N , рассматривается как экземпляр объекта N . Так как между множеством предикатов $P(x)$ и множеством элементов N существует взаимно однозначное соответствие, то эти два множества взаимозаменяемы. Взяв множество предикатов в роли объекта, можно использовать элементы этого множества в роли экземпляров объектов.

Введем множество Y — множество семантических областей, включающих понятия, выражаемые существительными обрабатываемого сверхфразового единства. Под понятием понимается совокупность суждений о каком-либо объекте, отражающем его сущность; мысль, являющаяся результатом познания объекта, выделяющая предметы некоторого класса по определенным общим и совокупным специфичным для них признакам. Понятие формируется в сфере мышления и имеет внеязыковую природу. Но поскольку мысль не может существовать вне слова, под понятием будем подразумевать лексическую единицу, представляемую значением переводного эквивалента данного существительного и выражающую определенное понятие.

Семантические области Y возможных переводных эквивалентов многозначных существительных

сверхфразовых единств являются подмножеством предметных областей специализированных словарей, используемых при переводе $Y \subset D$. Множество используемых специализированных словарей является конечным и достаточно четко очерченным $D = \{y_j\}, 1 \leq j \leq m$. Предикат $Q(y)$, заданный на множестве D , рассматривается как экземпляр объекта D . Так как между множеством предикатов $Q(y)$ и множеством элементов D существует взаимно однозначное соответствие, то эти два множества взаимозаменяемы.

Два множества $N = \{x_i\}$ и $D = \{y_j\}$ являются базовыми при использовании метода идентификации элементов смысла многозначных существительных сверхфразовых единств с использованием предикатных модифицированных категорий.

Ядром морфизма категории *Pred* будет служить предикат $C_f(x,y)$ (1), заданный на множестве $N \times D$. В предикатной категории *Pred* каждому морфизму $f \in \text{Pred}$ взаимно однозначно соответствует ядро $C_f(x,y)$, задающее отношение между существительными сверхфразового единства N и предметными словарями, отображающими семантические области понятия переводного эквивалента, D .

Множество N представляет собой множество всех многозначных существительных анализируемого сверхфразового единства входного текста. Под многозначным существительным будем понимать словоформу, которой на этапе морфологического анализа присвоен параметр S (часть речи существительное) и перевод данной словоформы есть в более чем одном подключенном словаре. Множество D представляет собой некоторую четко очерченную совокупность подключенных к переводу предметных, базовых и пользовательских словарей. Предполагается, что подключенные к переводу словари охватывают область семантического пространства существительных анализируемого сверхфразового единства переводимого текста. Наличие в подключенном словаре словарной статьи существительного множества N устанавливает морфизм $f \in \text{Pred}$. Рассмотрев все возможные пары из множества $N \times D$, получаем множество морфизмов $\text{Hom}_{\text{Pred}}(N,D)$, включающее все возможные отношения $C(x_i,y_j), 1 \leq i \leq n, 1 \leq j \leq m$.

4. Пример реализации модели

Рассмотрим сверхфразовое единство:

After selling some of the surplus blood to hospitals, the Red Cross has begun to destroy thousands of pints that have outlasted their shelf life. Directors of several Red Cross blood centers said their sites may discard as many as 1 of every 5 donations and the national total could easily reach tens of thousands.

На вход семантического анализа глубинного лингвистического процессора поступает множество $N = \{x_i\}, 1 \leq i \leq 8$, многозначных существитель-

ных: $x_1 = \text{blood}; x_2 = \text{hospital}; x_3 = \text{red}; x_4 = \text{cross}; x_5 = \text{shelf}; x_6 = \text{director}; x_7 = \text{center}; x_8 = \text{site}$.

Множество подключенных переводных словарей $D = \{y_j\}, 1 \leq j \leq m: y_1 = \text{General}; y_2 = \text{Law}; y_3 = \text{Biology}; y_4 = \text{Medical}; y_5 = \text{Economics}; y_6 = \text{Education}; y_7 = \text{Telecoms}; y_8 = \text{Polytechnic}$.

На рис. 1 показана категорная диаграмма, представляющая двудольный граф предиката $C(x,y)$.

Ядро морфизма категории *Pred*, заданного на декартовом произведении $N \times D$ множеств $N = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8\}$ и $D = \{y_1, y_2, y_3, y_4, y_5, y_6, y_7, y_8\}$, отображает отношения между элементами каждой пары x_i, y_j :

$$C(x,y) = (x^{x_1} \vee x^{x_2} \vee x^{x_3} \vee x^{x_4} \vee x^{x_5} \vee x^{x_6} \vee x^{x_7})y^{y_1} \vee \vee x^{x_5} y^{y_2} \vee (x^{x_3} \vee x^{x_4})y^{y_3} \vee (x^{x_3} \vee x^{x_4})y^{y_5} \vee x^{x_8} y^{y_6} \vee (x^{x_4} \vee x^{x_5} \vee x^{x_6} \vee x^{x_8})y^{y_7} \vee (x^{x_5} \vee x^{x_6} \vee x^{x_7})y^{y_8}. \quad (2)$$

Тогда согласно преобразованию (1) соответствующий линейный логический оператор будет записан в виде:

$$Q(y) = \exists x \in \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8\} (((x^{x_1} \vee x^{x_2} \vee x^{x_3} \vee \vee x^{x_4} \vee x^{x_5} \vee x^{x_6} \vee x^{x_7})y^{y_1} \vee x^{x_5} y^{y_2} \vee (x^{x_3} \vee x^{x_4})y^{y_3} \vee \vee y^{y_4} (x^{x_1} \vee x^{x_2} \vee x^{x_8}) \vee (x^{x_3} \vee x^{x_4})y^{y_5} \vee x^{x_8} y^{y_6} \vee (x^{x_4} \vee \vee x^{x_5} \vee x^{x_6} \vee x^{x_8})y^{y_7} \vee (x^{x_5} \vee x^{x_6} \vee x^{x_7})y^{y_8})P(x)). \quad (3)$$

Используя предикат узнавания предмета a по переменной x_i [7]:

$$x_i^a = \begin{cases} 1, & x_i = a \\ 0, & x_i \neq a \end{cases}, \quad (4)$$

где $i = \{1, 2, \dots, n\}$; a — любой элемент универсума, определяем реакцию морфизма на предикат $P(x) = x^{x_1} \vee x^{x_2}$, включающий существительные первой фразы сверхфразового единства *After selling some of the surplus blood to hospitals*:

$$Q(y) = \exists x \in \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8\} (((x^{x_1} \vee x^{x_2} \vee x^{x_3} \vee \vee x^{x_4} \vee x^{x_5} \vee x^{x_6} \vee x^{x_7})y^{y_1} \vee x^{x_5} y^{y_2} \vee (x^{x_3} \vee x^{x_4})y^{y_3} \vee \vee (x^{x_3} \vee x^{x_4})y^{y_3} \vee y^{y_4} (x^{x_1} \vee x^{x_2} \vee x^{x_8}) \vee (x^{x_3} \vee x^{x_4})y^{y_5} \vee \vee x^{x_8} y^{y_6} \vee (x^{x_4} \vee x^{x_5} \vee x^{x_6} \vee x^{x_8})y^{y_7} \vee \vee (x^{x_5} \vee x^{x_6} \vee \vee x^{x_7})y^{y_8}) (x^{x_1} \vee x^{x_2})) = y^{y_1} \vee y^{y_4}. \quad (5)$$

Для получения множества Q на графе (рис. 1) собираются вместе те элементы D , которые связаны с ребрами двудольного графа предиката $C(x,y)$ с элементами N , образующими множество $P(x) = x^{x_1} \vee x^{x_2}$. Множество $Q = \{y_1, y_2\}$ в рассматриваемом нами примере — это подключенные сло-

вари $y_1 =$ “General dictionary” и $y_4 =$ “Medical dictionary”.

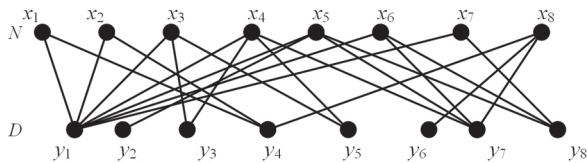


Рис. 1. Категорная диаграмма

Соответствующие переводные эквиваленты существительного x_1 — *кровь* (в базовом словаре “General dictionary” и в медицинском словаре — “Medical dictionary”); существительного x_2 — *госпиталь* (в базовом словаре “General dictionary” и в медицинском словаре — “Medical dictionary”). Используя полученные значения многозначных переводных эквивалентов, получаем наиболее правильный перевод рассматриваемой фразы: “После сбыта некоторых излишков крови в госпитали...”.

Выводы

Работа современных систем машинного перевода невозможна без реализации глубинного лингвистического процессора, осуществляющего аналитико-синтаксическую обработку текстов на естественном языке, использование которого на этапе семантического анализа требует формализовать выявление предметной области, раскрываемой в каждом сверхфразовом единстве или абзаце.

Использование предикатных и модифицированных категорий для моделирования отношений между многозначными существительными сверхфразовых единств и специализированными словарями позволяет создать легко реализуемую модель, снимающую многозначность переводных эквивалентов систем автоматического перевода. Так как математическим аппаратом данной модели является не базовая, а модифицированная предикатная теория категория, то данная модель может быть практически реализована не только средствами программного обеспечения, но и средствами аппаратного обеспечения, что в свою очередь обеспечит максимальное быстродействие.

Список литературы: 1. С. Мак Лейн [MacLane S.] Категории для работающего математика [текст]. — М.: Физматлит, 2004 [1998]. 2. Булкин, В.И. Использование метода декомпозиции бинарных предикатов при формализации интеллектуальной деятельности / В.И. Булкин, Н. Ф. Хайрова, Н. В. Шаронова // Вестн. Херсон. гос. техн. ун-та., Херсон, 2005.— N 1. С. 78-82. 3. D.E. Rydeheard, R.M. Burstall Computational Category Theory. — New York: Prentice Hall. — 1988. — XIII, 257 p. 4. Бондаренко, М.Ф. О модифицированных категориях [текст] / М.Ф. Бондаренко, З.В. Дударь, А.А.Иванов, В.В. Маникин, Ю.П. Шабанов-Кушнаренко // Радиоэлектроника и информатика. — Х.: Изд-во ХНУРЭ. — 2005.— № 1 — С. 87-99. 5. Хайрова, Н. Модель разбиения множества элементов смысла многозначных слов переводимого предложения в системах автоматического перевода [текст] / Н. Хайрова, Н. Шаронова // Бионика интеллекта: науч.-техн. журнал. — 2007. — № 2 (67). — С. 37-40. 6. Хайрова, Н.Ф. Машинный перевод [текст]: Учеб. пособие / Н.Ф. Хайрова, И.В. Замаруева. — Х.: Око, 1998. — 82 с. 7. Шабанов-Кушнаренко, Ю.П. Теория интеллекта: математические средства [текст] / Ю.П. Шабанов-Кушнаренко — Х.: Вища шк., 1984.— 143 с.

Поступила в редколлегию 18.09.2009

УДК 519.767

Використання математичного апарату предикативних категорій для моделювання семантики зверхфразових єдностей / Н.Ф. Хайрова // Біоніка інтелекту: наук.-техн. журнал. — 2009. — № 2 (71). — С. 36-39.

У пропонованій статті розглядаються завдання роботи лінгвістичного процесора в сучасних системах машинного перекладу. Показано використання математичного апарату предикативних і модифікованих категорій для моделювання зв'язків в універсумі всіляких елементів мовної системи. Запропонована модель дозволяє істотно зменшити лексичну омонімію багатозначних іменників зверхфразових єдностей.

Л. 1. Бібліогр.: 7 найм.

UDC 519.767

Use of a mathematical tools of predicate categories for modelling semantics of superphrases unities / N. Khairova // Bionics of Intelligence: Sci. Mag. — 2009. — № 2 (71). — P. 36-39.

The present article is on working out algorithms of linguistic processor of machine translation systems. This article reviews the method of modeling relations between every possible elements of linguistic system by mathematical tools of predicate categories. Proposed model allows to decrease lexical ambiguity of polysemantic nouns of superphrases unities.

Fig. 1. Ref.: 7 item.

УДК 658.012.011.56



МЕТОД И ТЕХНОЛОГИЯ ПРЕДСТАВЛЕНИЯ ЭТИМОЛОГИЧЕСКОГО СЛОВАРЯ УКРАИНСКОГО ЯЗЫКА В ЦИФРОВОЙ СРЕДЕ

И. В. Остапова

Украинский языково-информационный фонд НАН Украины, г. Киев, Украина,
iros@zeos.net

На основе формальной модели лексикографической системы этимологического словаря украинского языка разработана технология построения инструментальной системы для поддержки функционирования словаря в цифровой среде. Основное внимание уделяется механизму языковой индексации словаря.

ЛЕКСИКОГРАФИЧЕСКАЯ СИСТЕМА, ЭТИМОЛОГИЧЕСКИЙ СЛОВАРЬ, ИНДЕКС СЛОВАРЯ, ЛЕКСИКОГРАФИЧЕСКАЯ БАЗА ДАННЫХ, ПОЛЬЗОВАТЕЛЬСКИЙ ИНТЕРФЕЙС

Введение

В последнее десятилетие словари в цифровой форме, предназначенные для компьютерного пользователя (как на CD-дисках, так и в режиме on-line), получили не только широкое распространение, но и имеют тенденцию занять лидирующее положение в сфере лексикографических изданий. Компьютерные словари предоставляют, по преимуществу, справочно-лингвистическую информацию в виде, адекватно отвечающем уровню существующего запроса. Сказанное, однако, не исключает возможности извлечения и концептуальной языковой информации, «скрытой», «зашифрованной» в лексикографических структурах. Это справедливо не только по отношению к обычным пользователям (для справочной информации), но и для лингвистов (для дальнейших исследований) [1–3].

Очевидное преимущество лексикографических трудов в цифровой форме — это эффективность поиска требуемой информации и возможность проведения вычислительных экспериментов. Особенно это актуально для многотомных словарных изданий. Основным же побудительным мотивом в деле внедрения новых информационных технологий в лексикографии служит необходимость вывести на новый уровень языковедческие исследования [1, 6]. Чрезвычайно долгий цикл создания и модификации фундаментальных бумажных словарей приводит к тому, что представленная в них информация устаревает к моменту выхода книжного издания. Кроме того, сложившаяся технология подготовки словаря к печати делает невозможной внедрение в словарные структуры новых достижений теоретической лексикографии. Немаловажное значение имеет и предоставляемая цифровой средой возможность многократного использования однажды сформированных лексикографических структур и массивов многими профессионалами: лингвистами, лингвотехнологами и издателями.

1. Постановка задачи

Этимологические словари — это специальные лингвистические справочные системы, которые содержат информацию о генетических связях (этимологии) слов определённого языка или группы родственных языков. Этимологические словари используются как справочные издания для широкого круга пользователей, так и в качестве инструмента для дальнейших этимологических исследований.

Этимологический словарь украинского языка (далее — ЭСУЯ) представляет собой фундаментальный лексикографический труд, который создаётся в рамках проекта формирования национальной словарной базы Украины [4]. Первый том был издан в 1982 году, выход шестого тома ожидается в конце 2009 года; седьмой том по замыслу авторов будет представлять индекс ко всему словарному массиву. Для этимологических словарей наиболее действенным поисковым инструментом является индекс по языковой принадлежности слов, с которыми устанавливается генетическая связь в каждой словарной статье.

На сегодняшний день (на материале 5-ти вышедших из печати томов) представлены уже 232 различных языка, что было установлено в результате создания нами компьютерной лексикографической базы данных ЭСУЯ (речь об этом пойдет ниже). Для каждого языка в словаре необходимо создать отдельный индекс с идентификацией всех точных текстовых локализаций каждого слова данного языка. Для всего массива словаря предполагаемая размерность индекса — около 120 тысяч единиц. Трудоемкость работы по построению такого индекса столь велика, что задача создания его в «ручном» режиме представляется технологически неоправданной. Поэтому и была поставлена задача разработки специальной цифровой лексикографической среды, адаптированной к структурам ЭСУЯ и ориентированной на создание моноязычного индекса в автоматическом режиме.

Цифровая среда представляет собой качественно новый уровень сервиса для исследовательской работы с лингвистической информацией, представленной в словарной форме. В первую очередь это относится к индексным системам. Под индексацией словаря мы понимаем набор формализованных правил и процедур, на основании которых можно получить информацию об определённых классах языковых фактов, зафиксированных в словаре. Реализуются эти правила в форме пользовательских интерфейсов. Следует, однако, учитывать тот факт, что эффективность построения индексных схем для цифрового словаря возможна только в среде, достаточно формализованной.

При выполнении работы по созданию цифровой версии ЭСУЯ использовались методы, которые уже были успешно опробованы для решения подобных задач, в частности, для создания компьютерной лексикографической базы данных нового толкового словаря украинского языка [6].

2. Структура лексикографической системы этимологического словаря украинского языка

Мы рассматриваем словарь как информационную систему особого типа — лексикографическую. Согласно теории лексикографических систем это абстрактный языково-информационный объект, ориентированный на реализацию комплексного информационного описания лексико-грамматических структур определённого языка или совокупности языков [6].

Архитектура системы отвечает стандартной трёхуровневой архитектуре информационных систем ANSI/SPARK, согласно которой в информационной системе выделяются концептуальный, внутренний и внешний уровни представления данных [5].

В качестве концептуальной модели используется лексикографическая модель данных [6]. Ниже мы приводим её в несколько упрощённом виде:

$$\{I_0(D), V(I_0(D)), \beta, \delta[\beta], Red[V(I^Q(D))]\},$$

где D — объект моделирования — словарь украинского языка; $I_0(D) = \{x_i\}$ множество реестровых единиц словаря, в теории лексикографических систем его принято называть множеством *элементарных информационных единиц*; $V(I_0(D))$ — множество описаний (интерпретаций) элементарных информационных единиц, то есть текстов словарных статей: $V(I_0(D)) = \{V(x_i)\}$ — словарная статья с заголовковым словом (реестровой единицей) x_i ; β — множество структурных элементов, которые абстрагируются в результате анализа текста словаря; $\delta[\beta]$ — структура, порождаемая на β оператором δ , так что ограничение $\delta[\beta]$ на $V(x)$ порождает микроструктуру $\delta(x)$ словарной статьи; $Red[V(I_0(D))]$ — механизм *рекурсивной редукции лексикографической системы*. Он

даёт возможность последовательно выявлять всё более тонкие детали структуры лексикографической системы.

Изложенная модель представляет концептуальный уровень архитектуры и строится на основе анализа полиграфической версии ЭСУЯ: полиграфического оформления, организации текстов словарных статей, их структурных элементов, интерпретируемых как идентификаторы соответствующих элементов лексикографических структур β и $\delta(x)$.

В качестве базового структурного элемента лексикографической системы ЭСУЯ мы определяем *этимологический класс*, который представляет собой блок линейного текста словарной статьи, вычленимый по формальным признакам: структурная единица идентифицируется как этимологический класс, если в тексте словарной статьи можно выявить уникальные знаковые последовательности, используемые в качестве разделителей. Для ЭСУЯ нами выделены следующие типы этимологических классов: *класс реестрового слова* (обозначается *HEAD*), *класс дериватов* (*DER*), *класс славянских соответствий* (*SLAV*), *языковой класс* (*LANG*), *библиографический класс* (*BIBL*), *класс ссылок* (*LINK*). Каждый из этих классов имеет собственную текстовую структуру, что даёт возможность построения процедуры идентификации типа каждого этимологического класса в словарной статье по формальным признакам.

Класс реестрового слова и *языковой класс* составляют минимальную структуру словарной статьи. Этимологические классы других типов являются факультативными.

Проиллюстрируем изложенное на примере двух небольших словарных статей, достаточно репрезентативных с точки зрения структуры. Тексты приводим в форме, максимально приближенной к печатной версии.

Пример 1 (словарная статья с заголовковым словом **абетка**):

абетка, [*абетло*] Пі, *абетний* (заст.) «элементарный»;— власне українська назва азбуки, утворена за вимовою перших двох букв алфавіту (*a, be*), очевидно, під впливом назв *азбука, алфавет* і п. *abecadło* «тс.» (від вимови перших трьох букв *a, be, ce*).— *Sadn.* — *Aitz. VWb. I 42.*— Пор. **азбука, алфавіт**.

Этимологические классы для словарной статьи **абетка** (тексты классов подаются в угловых скобках):

HEAD ≡ <**абетка**>

DER ≡ <[*абетло*] Пі, *абетний* (заст.) «элементарный»>

LANG ≡ <власне українська назва азбуки, утворена за вимовою перших двох букв алфавіту (*a, be*),

очевидно, під впливом назв *азбука*, *альфабет* і п. abecadlo «тс.» (від вимови перших трьох букв a, b, c)»

BIBL ≡ <Sadn. — Aitz. VWb. I 42>

LINK ≡ <Пор. **азбука**, **алфавіт**>

Пример 2 (словарная статья с заголовковым словом **ва-банк**):

ва-банк (картярський термін) «ставка, що дорівнює всьому банку», (перен.) «ризкуючи всім»; — р. болг. *ва-банк*, п. *va banque*, ч. *vabank*, *vabanque*, слц. *vabank*, схв. *ва банк*; — запозичення з французької мови; фр. *va banque* (букв.) «іде банк» складається з дієслова *va* (3 ос. одн.), яке походить від лат. *vadit* «іде», спорідненого з двн. *watan* «переходити вбхід», дісл. *Vada*, дангл. *wadan* «тс.», і іменника *banque* «банк». — СІС 116; Шанский ЭСРЯ I 3,3; ССРЛЯ 2, 19; Dauzat 26, 737; Walde—Hofm. II 723—724. — Див. ще **банк**, **вада**.

Этимологические классы для словарной статьи **ва-банк**:

HEAD ≡ <**ва-банк** (картярський термін) «ставка, що дорівнює всьому банку», (перен.) «ризкуючи всім»>

SLAVIA ≡ <р. болг. *ва-банк*, п. *va banque*, ч. *vabank*, *vabanque*, слц. *vabank*, схв. *ва банк*>

LANG ≡ <запозичення з французької мови; фр. *va banque* (букв.) «іде банк» складається з дієслова *va* (3 ос. одн.), яке походить від лат. *vadit* «іде», спорідненого з двн. *watan* «переходити вбхід», дісл. *Vada*, дангл. *wadan* «тс.», і іменника *banque* «банк»>

BIBL ≡ <СІС 116; Шанский ЭСРЯ I 3,3; ССРЛЯ 2, 19; Dauzat 26, 737; Walde—Hofm. II 723—724>

*LINK*₁ ≡ <Див. ще **банк**, **вада**>

Приведём пример словарной статьи, минимальной как структурно, так и содержательно:

[**андріяк**] «опій»; — походження неясне.

Пример 3 (этимологические классы для словарной статьи **андріяк**):

HEAD ≡ <[**андріяк**] «опій»>

LANG ≡ <походження неясне>

В тексте каждого этимологического класса устанавливаются связи реестрового слова с определёнными словами других языков. Все эти слова, включая реестровые, мы называем *этимонами*. При анализе текстов этимологических классов было выявлено восемь параметров, посредством которых описываются этимоны: *маркер языковой принадлежности* (обозначим P_L), *ремарка к маркеру языковой принадлежности* (P_{RL}), *знаковое представление этимона* (P_A), *принадлежность к диалектной лексике* (P_D), *маркер омонимии* (P_O), *толкование* (P_S), *ремарка* (P_R), *библиография* (P_B). Мы перечислили параметры в том порядке, в котором они, как правило, следуют в тексте соответствующего этимологического класса. Два параметра являются обязательными: это P_L (*маркер языко-*

вой принадлежности) и P_A (*знаковое представление этимона*). Эти два параметра обеспечивают уникальность каждого этимона словарной статьи: этимоны с одинаковой знаковой формой могут иметь разную языковую принадлежность, или этимоны с одинаковой языковой принадлежностью могут иметь разные знаковые формы. Остальные параметры — факультативные. Для каждого параметра определена формальная процедура, которая позволяет вычленить соответствующий параметр из текста для каждого этимологического класса.

Набор параметров $\{P_L, P_{RL}, P_A, P_D, P_O, P_S, P_R, P_B\}$ мы называем *этимон-структурой* и обозначаем $ETYM(e_i)$, где e_i — соответствующий этимон; индекс i — порядковый номер данного этимона в тексте. Порядок следования параметров в этимон-структуре полагаем не существенным.

Не все параметры актуальны для каждого этимологического класса. Текст, идентифицируемый как этимологический класс, использует свое подмножество параметров; не каждый этимон обязан описываться полным набором параметров. Однако для достижения структурной однородности для каждого класса строится один тип этимон-структуры; если определённый параметр не задействован или не может быть выделен по формальным признакам, то его значению соответствует пустая строка текста. Этимон-структура строится только в том случае, если удалось вычленить P_A . Формально мы полагаем, что каждому этимологическому классу соответствует этимон-структура. Если языковой класс не имеет ни одного этимона (или не удалось его выявить формальной процедурой), то мы считаем его вырожденным этимологическим классом и ему соответствует пустая этимон-структура. Примером такого класса служит языковой класс для словарной статьи из примера 3.

Проиллюстрируем этимон-структуры на примерах текстов этимологических классов:

Пример 4 (этимон-структуры для класса реестрового слова):

HEAD (**абетка**) ≡ <абетка>

$ETYM(e_1) \equiv \{P_L = \langle \text{укр.} \rangle, P_A = \langle \text{абетка} \rangle\}$

HEAD (**ва-банк**) ≡ <**ва-банк** (картярський термін) «ставка, що дорівнює всьому банку», (перен.) «ризкуючи всім»>

$ETYM(e_1) \equiv \{P_L = \langle \text{укр.} \rangle, P_A = \langle \text{ва-банк} \rangle, P_R = \langle (\text{картярський термін}) \rangle, P_S = \langle \text{«ставка, що дорівнює всьому банку»} \rangle\}$

Пример 5 (этимон-структуры для класса дериватов):

DER (**абетка**) ≡ <[*абетло*] Пі, *абетний* (заст.) «элементарний»>

$ETYM(e_1) \equiv \{P_L = \langle \text{укр.} \rangle, P_A = \langle \text{абетло} \rangle, P_D = 1, P_B = \langle \text{Пі} \rangle\}$

$ETYM(e_2) \equiv \{\langle P_L = \langle \text{укр.} \rangle, P_A = \langle \text{абетний} \rangle, P_R = \langle (\text{заст.}) \rangle, P_S = \langle \text{«элементарний»} \rangle\}$

Параметр языковой принадлежности принимает значение «укр.» (украинский) по умолчанию, так как в этих текстах устанавливаются связи только со словами украинского языка. Параметр омонимии P_O для этимона e_1 принимает значение 1 (пример 5), так как квадратные скобки указывают на принадлежность слова к диалектной лексике. По умолчанию для всех этимонов значение этого параметра 0.

Пример 6 (этимон-структуры для класса славянских соответствий):

SLAV(ва-банк) \equiv <р. болг. *ва-банк*, п. *va banque*, ч. *vabank*, *vabanque*, слц. *vabank*, схв. *ва банк*>

В тексте этого этимологического класса устанавливается связь со словами русского (р.), болгарского (болг.), польского (п.), чешского (ч.), словацкого (слц.) и сербскохорватского (схв.) языков.

ETYM (e_1) \equiv {< P_L =<р.>, P_A =<ва-банк>}&

ETYM (e_2) \equiv {< P_L =<болг.>, P_A =<ва-банк>}&

ETYM (e_3) \equiv {< P_L =<п.>, P_A =<va banque>}&

ETYM (e_4) \equiv {< P_L =<ч.>, P_A =<vabank>}&

ETYM (e_5) \equiv {< P_L =<ч.>, P_A =<vabanque>}&

ETYM (e_6) \equiv {< P_L =<слц.>, P_A =<vabank>}&

ETYM (e_7) \equiv {< P_L =<схв.>, P_A =<>}&

Пример 7 (этимон-структуры для языкового класса):

LANG (ва-банк) \equiv <запозичення з французької мови; фр. *va banque* (букв.) «іде банк» складається з дієслова *va* (3 ос. одн.), яке походить від лат. *vadit* «іде», спорідненого з двн. *watan* «переходити вбхід», дісл. *Vada*, дангл. *wadan* «тс.», і іменника *banque* «банк»>

В приведенном тексте устанавливаются связи со словами французского (фр.), латинского (лат.), древнемецкого (двн.), древнеисландского (дісл.) и древнеанглийского (дангл.) языков.

ETYM (e_1) \equiv {< P_L =<фр.>, P_A =<va banque>, P_S =<«іде банк»>, P_R =<(букв.)>}&

ETYM (e_2) \equiv {< P_L =<фр.>, P_A =<va>, P_R =<(3 ос. одн.)>}&

ETYM (e_3) \equiv {< P_L =<лат.>, P_A =<vadit>, P_S =<«іде»>}&

ETYM (e_4) \equiv {< P_L =<двн.>, P_A =<watan>, P_S =<«переходити вбхід»>}&

ETYM (e_5) \equiv {< P_L =<дісл.>, P_A =<Vada>}&

ETYM (e_6) \equiv {< P_L =<дангл.>, P_A =<wadan>, P_S =<«тс.»>}&

ETYM (e_7) \equiv {< P_L =<фр.>, P_A =<banque>, P_S =<«банк»>}&

2. Парсинг этимологического словаря и создание его лексикографической базы данных

Основная проблема создания компьютерных словарей, исходя из их печатных версий, сводится к тому, каким образом сформировать соответствующую базу данных в автоматическом режиме не-

посредственно из текста словаря (парсинг). Опыт убеждает, что формирование нетривиальных лексикографических баз данных «вручную» из больших и сложных словарных текстов практически невозможно. Основной задачей парсинга является автоматическое выделение структурных элементов лексикографической системы непосредственно из текста словаря, поскольку именно они выполняют роль элементов лексикографической базы данных.

Парсинг ЭСУЯ был осуществлен следующим образом. Перед конверсией тексты всех томов были переведены в формат HTML и унифицированы как относительно структуры файлов, так и относительно знаковой системы. Разные тома словаря были подготовлены к печати различными издательскими технологиями. Первые три тома — в технологии монотайп, докомпьютерной. Поэтому печатные тексты сначала были отсканированы, распознаны программой FINEREADER, а затем вычитаны. Тексты 4-го и 5-го томов уже готовились в компьютерной издательской системе; для этого был разработан специальный набор компьютерных шрифтов, аналогичный используемому в 1, 2, 3 томах. Знаковая система словаря была унифицирована согласно кодировке UNICODE 3.0. Это позволило выполнить инвентаризацию символов алфавита для представления этимонов каждого языка.

Для связи между печатной и цифровой версиями словаря каждая словарная статья промаркирована следующим образом: номер тома, номер страницы начала текста, номер страницы конца текста.

Пример 8 (формат цифрового текста для статьи ва-банк; символ # используется для идентификации ударения):

@1 316 316

ва-ба#нк (картярський термін) «ставка, що дорівнює всьому банку», (перен.) «ризкуючи всім»; -- р. болг. <|>ва-ба#нк</|>, п. *va banque*, ч. *vabank*, *vabanque*, слц. *vabank*, схв. <|>ва банк</|>; -- запозичення з французької мови; фр. *va banque* (букв.) «іде банк» складається з дієслова *va* (3 ос. одн.), яке походить від лат. *vadit* «іде», спорідненого з двн. *watan* «переходити вбхід», дісл. *Vada*, дангл. *wadan* «тс.», і іменника *banque* «банк». -- СІС 116; Шанский ЭСРЯ I 3,3; ССРЛЯ 2, 19; Dauzat 26, 737; Walde--Hofm. II 723--724.-- Див. ще банк, ва#да.

В результате этих операций были получены специальным образом препарированные тексты томов этимологического словаря, полностью готовые для автоматической конверсии в лексикографическую базу данных.

Структура этой базы данных представляет ряд связанных таблиц, как это представлено на рис. 1.

Дадим краткую характеристику таблицам базы данных.

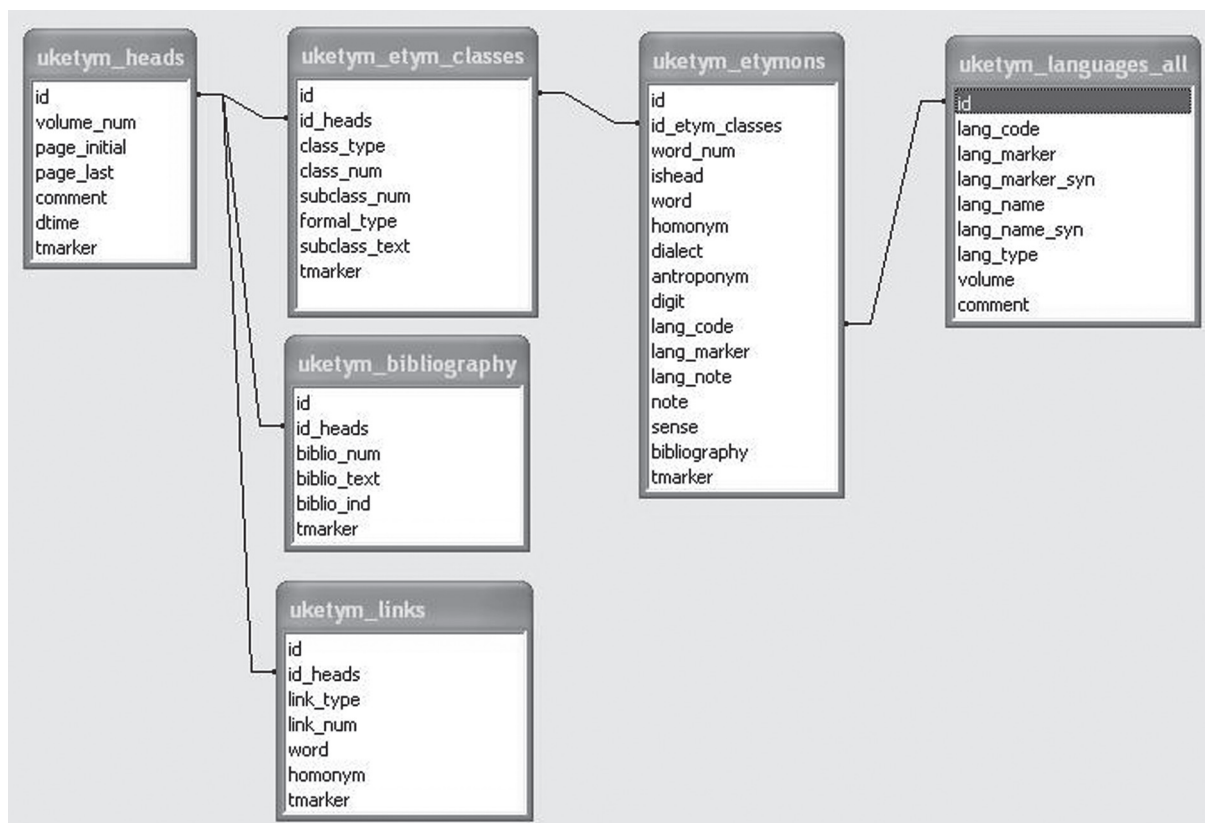


Рис. 1. Структура лексикографической базы данных этимологического словаря украинского языка

Таблица **uketym_heads** организует текст словарной статьи. Она хранит только идентификационный номер словарной статьи, номера тома, начальной и конечной страницы расположения текста словарной статьи в печатной версии.

Таблица языковых классов **uketym_etym_classes** хранит тексты словарных статей (за исключением библиографии и ссылок, которые организованы в отдельные таблицы).

Все межстатейные ссылки словаря организованы в таблицу **uketym_links**.

Библиография организована в таблице **uketym_bibliography**.

Таблица **uketym_etymons** содержит параметры этимон-структур.

В таблице **uketym_language_all** организована вся информация о языках, которые задействованы в словаре. На базе этой таблицы строятся пользовательские языковые регистры для индексирования. Приведём фрагмент этой таблицы:

id	Lang_code	lang_marker	lang_name	volume
127	127	п.	Польська	1
183	183	укр.	Українська	1
231	0	невизн.	невизначена	1

Как видим, в таблицу введен ещё один, «технологический», язык — неопределённый (мова «невизначена»). Это сделано из следующих соображений. Поскольку индексация производится автоматически, то не всегда согласно разработанно-

му алгоритму можно однозначно определить язык. В этом случае этимон маркируется неопределённым языком. По данному признаку из базы данных можно выбрать все подобные записи (например, сформировав языковой регистр, состоящий только из технологического языка) и затем произвести идентификацию языков, используя средства редактирования, предусмотренные в системе.

4. Функциональность инструментального этимологического комплекса

Для поддержки цифровой версии словаря построен инструментальный комплекс, который обеспечивает следующие основные функции [7]:

- 1) автоматическую конверсию текстов этимологического словаря в компьютерную базу данных;
- 2) традиционный вход в систему по реестровому слову и отображение текста словарной статьи;
- 3) редактирование любого структурного элемента словарной статьи;
- 4) построение этимон-структуры для словарной статьи в ручном режиме;
- 4) автоматическое построение этимон-структуры для словарной статьи;
- 5) создание словарной статьи с определенной структурой.

На рис. 2 показано одно из окон редактирования словарной статьи. На левой панели словарная статья представлена в виде дерева структурных элементов. Для каждого этимологического класса

выводится упорядоченный список этимонов, тем самым графическими средствами визуализируется глубина этимологического исследования. С помощью кнопок на средней панели структурные элементы можно добавлять, удалять и менять порядок их следования. Функции кнопок варьируются в зависимости от выбранного структурного элемента. Так, например, кнопка «Додати» (добавить) при выборе этимона позволяет добавить только этимон. Для каждого структурного элемента разработано своё окно редактирования, которое отражает специфику этого элемента. Для каждого этимона выводится и текст соответствующего этимологического класса, однако с запретом его редактирования. Это даёт возможность верифицировать параметризацию этимона, выполненную автоматически.

Для автоматического построения языковых индексов разработан специальный инструментарий, который позволяет:

1) в интерактивном режиме формировать любое количество языковых регистров на множестве всех языков словаря;

2) задавать спектры индексации, учитывая структуру словарной статьи.

На рис. 3 показано диалоговое окно пользователя для формирования языкового регистра.

Левая панель предназначена для выбора уже сформированных регистров в качестве неизменя-

емых шаблонов. Правая панель используется для редактирования существующих и формирования новых регистров.

В нашем примере мы сформировали новый регистр следующим образом: задали его имя «регистр_п_укр»; после проверки на уникальность новый регистр был занесён в список языковых регистров; в окно на левой панели загрузили регистр «Усі мови словника» (все языки словаря), который включает все задействованные в словаре языки; из списка были выбраны последовательно украинский и польский языки и перенесены в список на правой панели. Регистр «Усі мови словаря» может использоваться только в качестве шаблона. Предполагается в дальнейшем создать набор шаблонов со значимыми именами, например «Славянские языки», «Романские языки» и тому подобное.

На рис. 4 показано окно главного пользовательского интерфейса словаря с индексом, построенным по сформированному регистру.

На левой панели из предложенного набора языков был выбран польский (возможен выбор как всех языков в регистре, так и определённого подмножества языков). На правой панели в реестровое окно выведен список всех этимонов, которые идентифицированы как слова польского языка.

В окно реестра могут быть также выведены заголовковые слова тех словарных статей, в которых зафиксированы этимологические связи с поль-

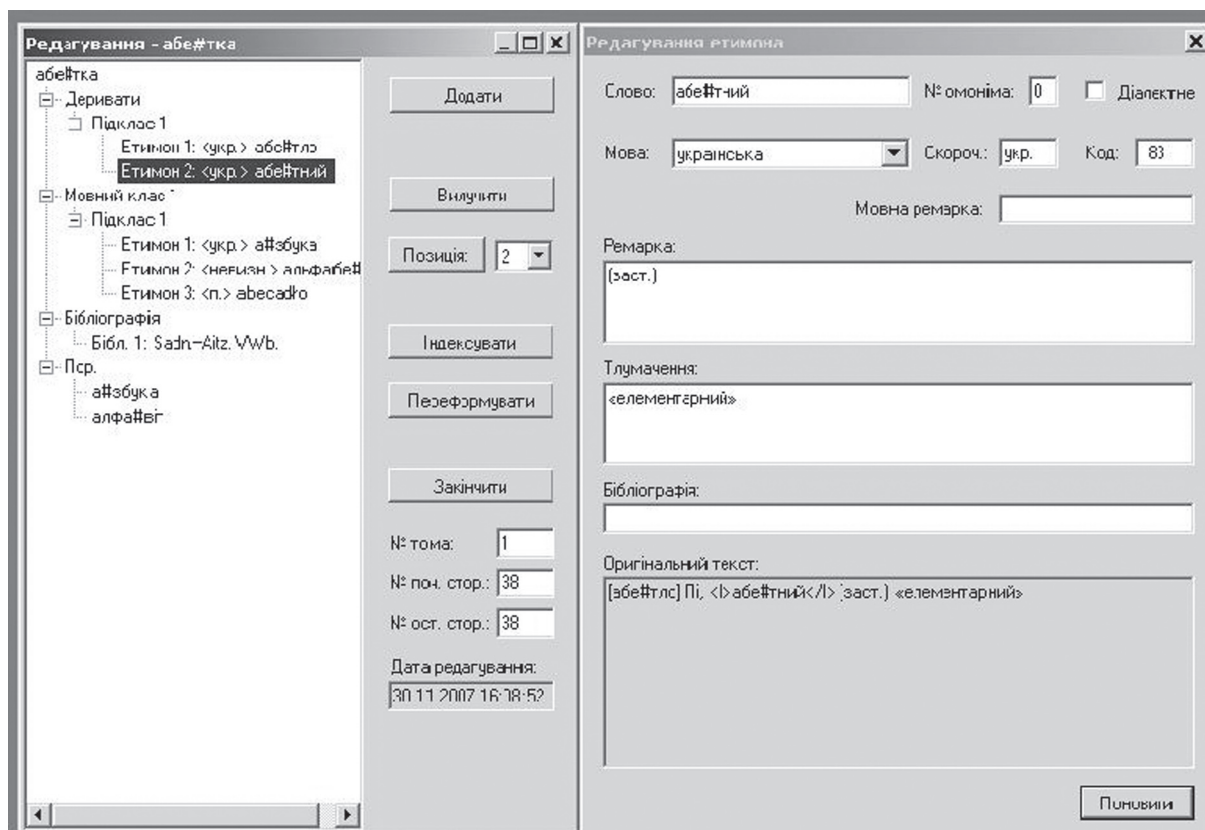


Рис. 2. Окно редактирования для словарной статьи с заголовковым словом **абетка**

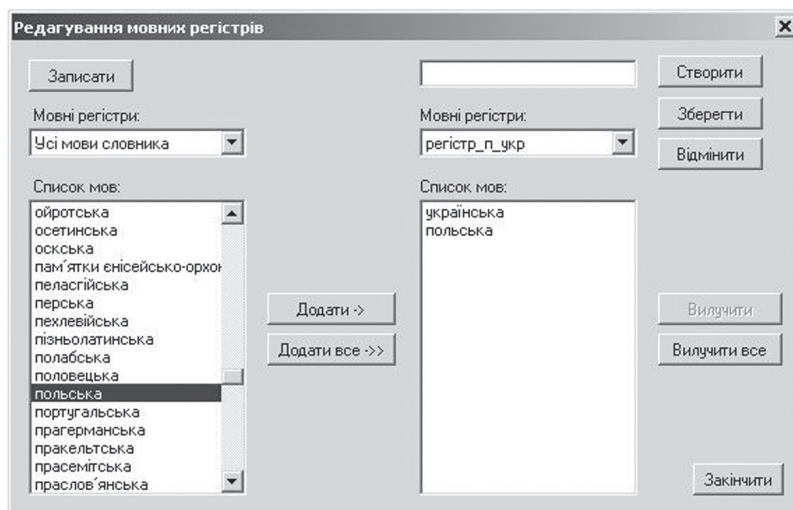


Рис. 3. Окно формирования языкового регистра

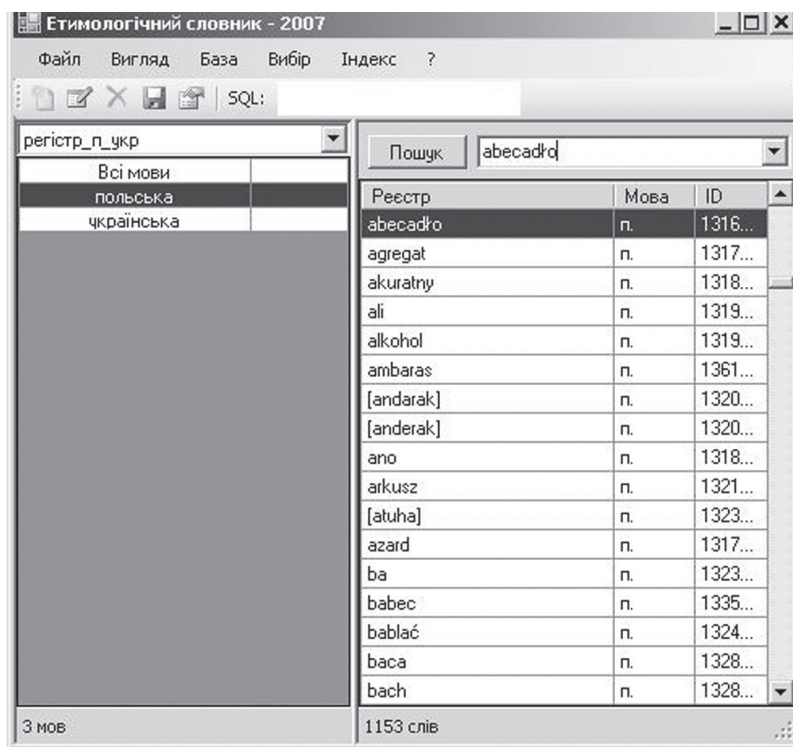


Рис. 4. Языковой индекс по заданному регистру

ским языком. Сформированный индекс по команде пользователя выводится в текстовый файл с указанием локализации каждого этимона.

Инструментальная система позволяет задать локализацию индексируемых элементов с точностью до структурного элемента — типа этимологического класса — словарной статьи (с помощью меню «Вибір» (Выбор) на верхней панели). В нашем случае был задан только *языковой класс*.

При активизации любого элемента реестра визуализируется текст словарной статьи.

Текст словарной статьи для вывода формируется из соответствующих полей базы данных. Полиграфическое оформление статьи практически сохранено полностью.

Выводы

Описанный метод представления этимологического словаря в цифровой среде дал возможность построить для совокупности словарных статей словаря соответствующую совокупность этимон-структур как формальных репрезентантов описаний генетических связей реестровых единиц. Все схемы индексации строятся только на основе этимон-структур. Такой подход обеспечивает как возможность построения структур, имплицитированных в текст словарных статей, так и отображения на цифровую среду аутентичного текста словаря, что делает цифровой словарь открытым для дальнейших интерпретаций.

Разработанные технологии и интерфейсы предлагаются как базовые для цифровых репрезентаций этимологических работ.

Список литературы: 1. Корпусна лінгвістика [Текст] / В. А. Широков, О. В. Бугаков, Т. О. Грязнухіна і др. ; відп. ред. В. А. Широков. — К. : Довіра, 2005. — 471 с. 2. *Григорович, Л. А.* Учебный словарь «NTC'American English Learner's Dictionary»: полиграфическая и электронная версии [Текст] / Л. А. Гончаров // Слово в словаре и дискурсе : Сб. науч. статей к 50-летию Харри Вальтера. — М. : ООО «Издательство “Элпис”», 2006. — 836 с. — С. 602–608. 3. *Большаков, И. А.* КроссЛексика — большой электронный словарь сочетаний и смысловых связей русских слов [Текст] / И. А. Большаков // Компьютерная лингвистика и интеллектуальные технологии : По материалам ежегодной Междунар. конф. «Диалог 2009» (Бекасово, 27–31 мая 2009 г.). Вып. 8(15). — М. : РГГУ, 2009. — XII, 620 с. — С. 45–50. 4. *Етимологічний словник української мови* : В 7 т. / Гол. ред. О. С. Мельничук. — Київ: Наукова думка, 1982–2006. Т. 1–5. 5. ANSI/X3/SPARK DBMS study group interim report. FDT-Bull. ACM SIGMOD. 1975. V. 7. № 2. 6. *Широков, В.А.* Елементи лексикографії [Текст] / В. А. Широков. — Київ: Довіра, 2005. — 304 с. 7. *Остапова, И. В.* Инструментальная лексикографическая система Этимологического словаря украинского языка [Текст] / И. В. Остапова, К. Н. Якименко // Прикладна лінгвістика та лінгвістичні технології : MegaLing-2007 : зб. наук. пр. — Київ: Довіра, 2008. — 511 с. — С. 276–291.

Поступила в редколлегию 01.10.2009

UDK 658.012.011.56

Метод і технологія представлення етимологічного словника української мови у цифровому середовищі / І. В. Остапова // Біоніка інтелекту: наук.-техн. журнал. — 2009. — № 2 (71). — С. 40-47.

Описано технологію побудови комп'ютерного етимологічного словника на основі методології лексикографічних систем. Словник розглядається як інформаційна система, архітектура якої відповідає стандартній архітектурі ANSI/SPARK. На концептуальному рівні використовується лексикографічна модель даних. Розроблений користувацький інтерфейс забезпечує доступ до кожного структурного елемента словника.

Л. 4. Бібл.: 7 найм.

UDK 658.012.011.56

Method and technology of the Ukrainian Etymological Dictionary representations in the digit mood / I. V. Ostapova // Bionics of Intelligence: Sci. Mag. — 2009. — № 2 (71). — P. 40-47.

On the base of methodology of the lexicographical systems the technology for the computer Etymological Dictionary forming is described. The dictionary is considered as the information systems of the standard architecture ANSI/X3/SPARK with the lexicographical model as the conceptual level of data representation. The user interface supports access for even structure element of the dictionary.

Fig. 4. Ref.: 7 items.

УДК 004.5:81'374.82



ИНСТРУМЕНТАЛЬНАЯ ЛЕКСИКОГРАФИЧЕСКАЯ СИСТЕМА В ОБЛАСТИ ФИЗИКИ МАГНИТНЫХ ЯВЛЕНИЙ

С.С. Дикарева¹, В.Н. Бержанский², С.Н. Полулях³, Е.В. Потапова⁴¹ТНУ им. В.И.Вернадского, Центр когнитивной и прикладной лингвистики, Симферополь, Украина²Таврический национальный университет им. В.И.Вернадского, Симферополь, Украина³Таврический национальный университет им. В.И.Вернадского, Симферополь, Украина⁴ТНУ им. В.И.Вернадского, Центр когнитивной и прикладной лингвистики, Симферополь, Украина

В статье представлены результаты работы по созданию редактора лексикографической системы трехязычного (русский-украинский-английский) словаря терминов по физике магнитных явлений. Разработана оригинальная лексикографическая параметризация термина. Лексикографическая база данных насчитывает более 900 словарных статей. Редактор лексикографической системы имеет функцию публикации содержимого лексикографической базы в виде HTML-сайта.

ЛЕКСИКОГРАФИЧЕСКАЯ СИСТЕМА, ЭЛЕКТРОННЫЙ СЛОВАРЬ, БАЗЫ ДАННЫХ, ПРОГРАММНЫЙ ИНТЕРФЕЙС

Введение

Электронный русско-украинско-английский «Словник термінів з фізики магнітних явищ» создан группой авторов совместно с Институтом магнетизма НАН и МОН Украины (г.Киев) в рамках «Державної програми розвитку й функціонування української мови на 2004-2010 роки». В основу данного проекта положен мультидисциплинарный подход к представлению научных знаний по физике магнитных явлений в электронных лексикографических системах [1, 2]. Принципы создания электронных лексикографических систем описаны в цикле работ В.А.Широкова [3, 4].

Проект направлен на систематизацию украинской физической терминологии на основе методов когнитивной и прикладной лингвистики [5, 2]. Последние десятилетия большое внимание уделяется проблемам терминологии и терминографии украинского языка. Создание многоязычной (русско-украинско-английской) лексикографической базы является актуальной задачей для многих предметных областей [6-8].

Целевой аудиторией являются студенты и преподаватели технических специальностей, научные сотрудники, технические переводчики.

Лексикографическая база данных на данный момент насчитывает более 900 словарных статей (эквиваленты на русском, украинском и английском языке, дефиниции терминов). В основу положены два печатных источника [9, 10].

1. Постановка задачи

Целью данной работы было создание инструментальной среды для ввода и редактирования информации в трехязычной лексикографической базе. В концептуальной модели системы помимо редактирования языковых реестров заложены следующие специфические возможности: двуязыч-

ный (русский-украинский) интерфейс редактора; ввод дефиниций термина на трех языках; наличие вариантов дефиниции; независимое редактирование информации о библиографических источниках, персоналиях, разделах физики магнитных явлений (ФМЯ); формирование пользовательской версии словаря в автоматическом режиме.

2. Структура лексикографической базы данных

Лексикографические параметры термина в созданной лексикографической системе разделены на две группы:

1. Параметры, описывающие термин внутри определенного языка (орфографический стандарт, грамматические характеристики, синонимы, дефиниция, пример контекстного употребления).

2. Параметры, содержащие информацию, универсальную для всех трех языков (семантические связи термина — гипонимы, гиперонимы; ассоциативные связи; персоналии; раздел ФМЯ).

В структуре лексикографической базы данных (ЛБД) выделены три языковые группы таблиц: английская, русская (рис. 1) и украинская, связанные друг с другом через таблицу `tbl_term`. Информация, не соотносимая с конкретным языком (родовидовые связи термина, персоналии, раздел ФМЯ) образуют отдельную группу таблиц, связанную с терминами конкретного языка через таблицу `tbl_term` (рис. 2).

На рис. 1. `tbl_term` — это общая для трех языков таблица переходов, `tbl_term_rus` — русский языковой реестр, `tbl_term_rus_synonym` — таблица для записи пар индексов терминов-синонимов для русского языка, `tbl_term_rus_details` — контекстные примеры употребления термина, `tbl_term_rus_descn` — текстовое описание термина.

Как отмечалось, в лексикографическую параметризацию термина включены гипонимы и ги-

пероним для данного термина, которые отражают связи термина типа «род-вид». Кроме того, включены ассоциативно связанные термины (по отношению к данному), определяемые экспертами исходя из содержания предметной области - физики магнитных явлений. Перечисленные выше типы отношений, в отличие от отношения синонимии, не являются внутриязыковым явлением и в базе данных хранятся в общей для всех языков таблице отношений (рис. 2).

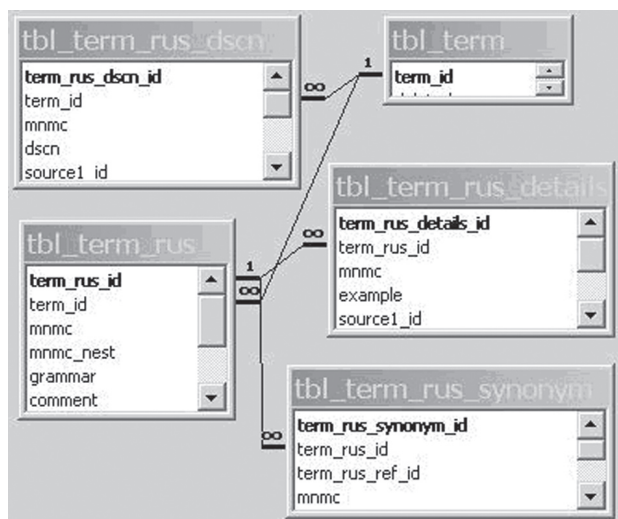


Рис. 1. Группа таблиц русского языка

На рис. 2 tbl_term – это общая для трех языков таблица переходов, tbl_term_rus, tbl_term_ukr, tbl_term_eng - русский, украинский и английский языковые реестры. tbl_relation_term – таблица для хранения пар индексов терминов, находящихся в определенном отношении (relation_type_id): гипонимы, гипероним, ассоциативно связанный термин. tbl_relation_type – реестр типов отношений. Таблица tbl_person_term связывает термины с персоналиями, tbl_subject_term – с разделами ФМЯ.

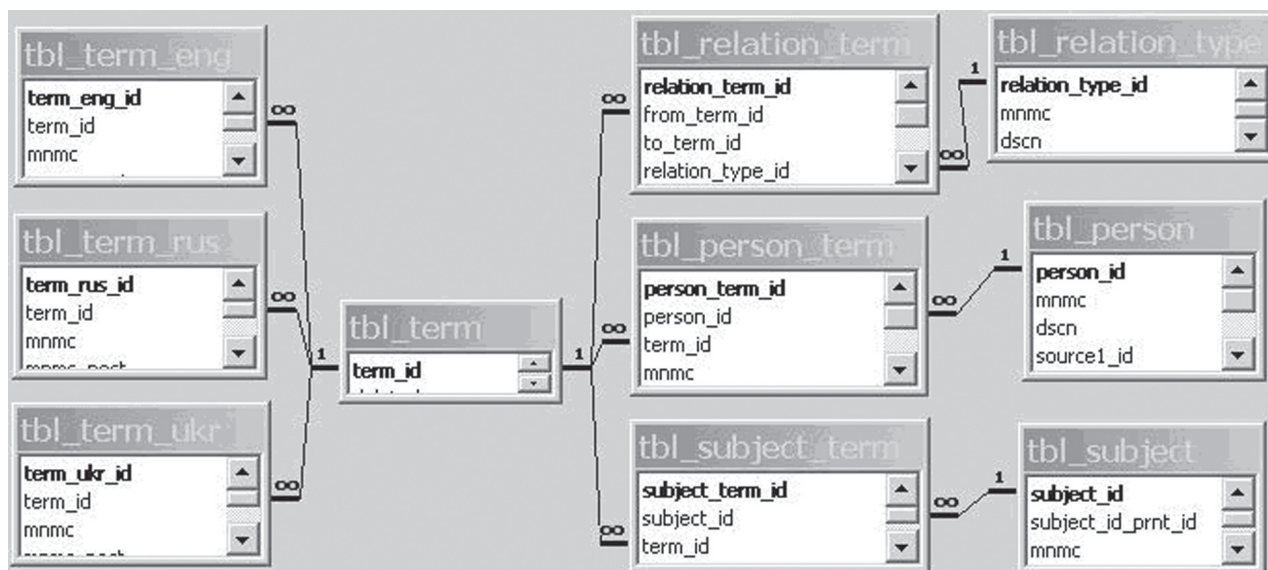


Рис. 2. Схема хранения связей термина в базе данных

3. Архитектура программного интерфейса редактора словаря

Работу с электронной лексикографической системой можно условно разделить на два режима: 1) работа редактора лексикографической базы – ввод данных в ЛБД и редактирование; 2) работа конечного пользователя словаря – поиск и просмотр информации, хранимой в лексикографической базе данных. Исходя из необходимости максимально упростить системные требования к пользовательскому компьютеру, оказалось целесообразно создать два интерфейса: редактор словаря (табл. 1) и версию словаря для конечного пользователя.

Редактор словаря реализован как локальное Windows-приложение (C#) с доступом к источнику данных – файлу словарной базы (ACCESS). Возможен удаленный доступ.

Таблица 1

Описание объектов и процедур редактора лексикографической базы данных

Группы объектов и процедуры	Назначение, функциональность
Управляющие элементы и объекты форм	Вывод информации ЛБД на экран, ввод запросов редактора, ввод и редактирование информации редактором-экспертом
Промежуточные объекты	Предварительная обработка массивов данных: хранения и преобразования вводимых и выводимых данных, формирования хеш-таблиц. Формирование SQL-запросов
Объекты и процедуры ввода/вывода	Установление связи с базой данных, редактирование и ввод новой информации по запросам (см. пред. шаг) в базу данных
Процедура генерации HTML кода	Генерация для каждого термина отдельной HTML страницы, заполненной информацией из ЛБД, с необходимыми гиперссылками для связи с другими страницами

Ступенчатая архитектура доступа к словарной базе данных (рис.3) имеет преимущество перед прямым доступом. Она позволяет модернизировать интерфейс редактора без вмешательства в другие группы объектов (табл. 1). Объекты ввода/вывода представляют собой отдельную библиотеку, которая в дальнейшем будет включена в сетевую версию редактора лексикографической базы.

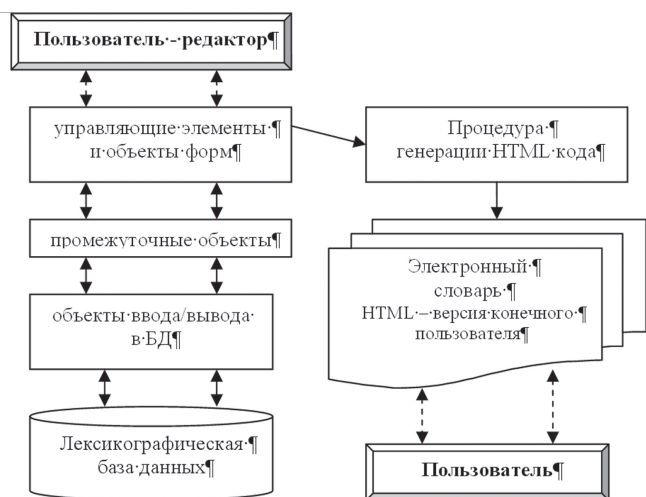


Рис. 3. Архитектура программного интерфейса редактора

Требования к компьютеру редактора словаря – операционная система Windows XP и последующих версий, Microsoft Office в конфигурации, включающей ACCESS.

Графический интерфейс (рис. 4.) редактора лексикографической базы разработан в соответствии с лексикографическими параметрами словаря. Меню приложения, панель инструментов, главный языковой реестр словаря (русский, украинский или английский – выбор зависит от языка запроса пользователя) присутствуют на экране в течение всего времени работы приложения. Также присутствует отображение искомого термина на трех языках (лексикографические параметры: орфографический стандарт, переводные эквиваленты).

Остальная информация группируется в отдельные вкладки следующим образом:

1. Описание (лексикографический параметр: дефиниция)
 - Текстовое описание.
 - Источник описания.
2. Примеры употребления
 - Пример употребления в тексте (отрывок из научной или научно-популярной статьи) (лексикографический параметр: контексты употребления).
 - Источник примера.
3. Дополнительно
 - Раздел ФМЯ.
 - Персоналии. Сведения о персоналиях, так или иначе связанных с данным термином.
4. Грамматика
 - Грамматическое описание термина в трех языках (лексикографический параметр: грамматикализация).

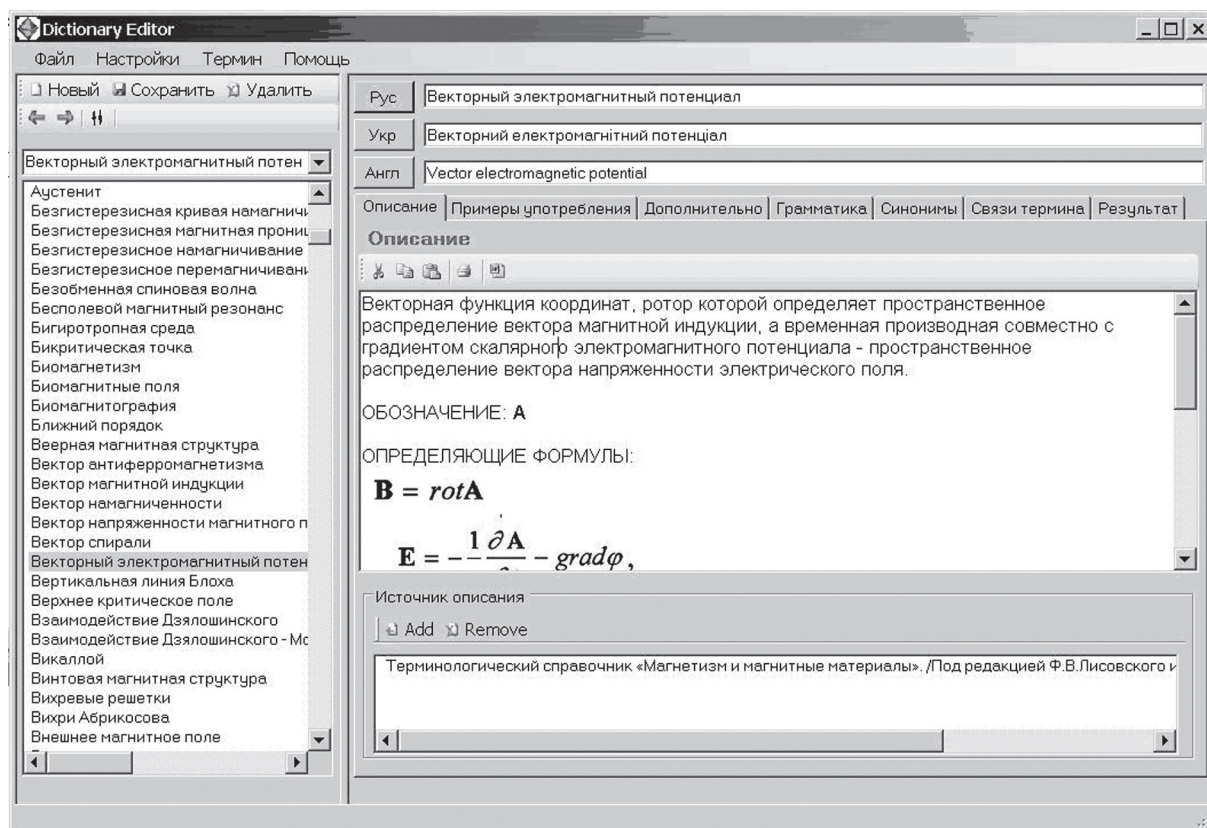


Рис. 4. Интерфейс эксперта. Вкладка «Описание»

- Комментарий (лексикографический параметр: ремарки).

5. Синонимы (лексикографический параметр: варианты и синонимы).

6. Связи термина – информация, необходимая для построения карты термина.

- Гипероним (лексикографический параметр: гипероним (родовое понятие)).

- Гипонимы (лексикографический параметр: гипонимы (видовые понятия)).

- Термины, ассоциативно связанные с данным термином (лексикографический параметр: ассоциативный ряд).

7. Результат – словарная статья термина (в формате HTML), построенная на основе информации, введенной в ЛБД на предыдущих вкладках.

В приложении предусмотрен выбор языка интерфейса пользователя: русский, украинский, английский. Он осуществляется с помощью команд Главного меню: **Настройки>Украинский интерфейс** (Русский, Английский там же). Полный перечень команд Главного меню и их назначение содержит табл. 2.

Таблица 2

Команды Главного меню

Команда меню	Назначение
Файл> База данных	В диалоговом окне указать путь к базе данных
Файл> Обновить	Обновить главный реестр терминов
Файл> Генерировать HTML	Запуск процедуры генерации набора HTML-страниц для создания версии словаря для конечного пользователя
Файл> Выход	Выход из программы
Настройки> Русский интерфейс	Выбор языка интерфейса
Настройки> Украинский интерфейс	Выбор языка интерфейса
Настройки> Английский интерфейс	Выбор языка интерфейса
Настройки> Источники	Открывает диалог для добавления или редактирования выходных данных источников описаний и текстовых примеров
Настройки> Персоналии	Открывает диалог для добавления или редактирования сведений о персоналиях
Настройки> Разделы ФМЯ	Открывает диалог для добавления или редактирования разделов ФМЯ
Термин> Новый...	Диалог для добавления нового термина в словарь
Термин> Сохранить	Сохраняет всю информацию для выбранного на данный момент термина
Термин> Удалить	Удаляет выбранный термин
Помощь> О программе	Демонстрирует сведения о проекте

На любом этапе работы над словарем содержимое словаря может быть опубликовано в электронном виде. Создание версии электронного терминологического словаря для конечного пользователя происходит по команде **Файл>Генерировать HTML** из меню редактора словаря. Запускается процедура, генерирующая HTML код (см. рис. 3) для отображения данных, находящихся в лексикографической базе на момент запуска процедуры. Вся информация лексикографической базы отображается в виде отдельных для каждого термина HTML-страниц и сохраняется в отдельную папку. К этому набору страниц были добавлены служебные файлы (собственно, **Интерфейс конечного пользователя**) для Главной страницы, Алфавитного указателя (на 3-х языках), Оглавление разделов ФМЯ и некоторые служебные файлы для оформления сайта и организации поиска.

4. Интерфейс конечного пользователя

Интерфейс конечного пользователя представляет собой сайт, построенный на основе DHTML и Java-script, что позволяет минимизировать требования к компьютеру конечного пользователя. Пользовательская версия словаря может распространяться на лазерных дисках или размещаться в Интернете. Для установки пользовательской версии на локальном компьютере пользователя достаточно, чтобы на нем была установлена операционная система Windows.

В интерфейсе конечного пользователя предусмотрена группировка терминов по разделам ФМЯ и по Персоналиям (рис. 5). В отличие от редактора здесь сразу приводятся все три языковые реестры, причем в первую колонку помещается реестр на языке алфавитного указателя.

Словарная статья термина (рис. 6) помимо информации из базы данных содержит определенное количество гиперссылок, связывающих ее с другими словарными статьями, а также с источниками в Интернете (если был указан Интернет адрес). Каждая словарная статья открывается в отдельном окне, что дает возможность одновременно изучать информацию по нескольким терминам.

Часть словарной статьи, содержащая информацию о связях термина, представляет собой электронную интерактивную карту термина в табличном виде (рис. 7). Термины, связанные с данным, приводятся сразу на трех языках и представляют собой гиперссылки для быстрого перехода к словарной статье выбранного термина.

Карта термина отражает участок дерева концептов предметной области [2] и позволяет воспринимать не отдельные статичные образы, а смысловые схемы, логические связи, характер взаимодействия между предметами и явлениями. Графическое отображение карты термина находится в стадии разработки.

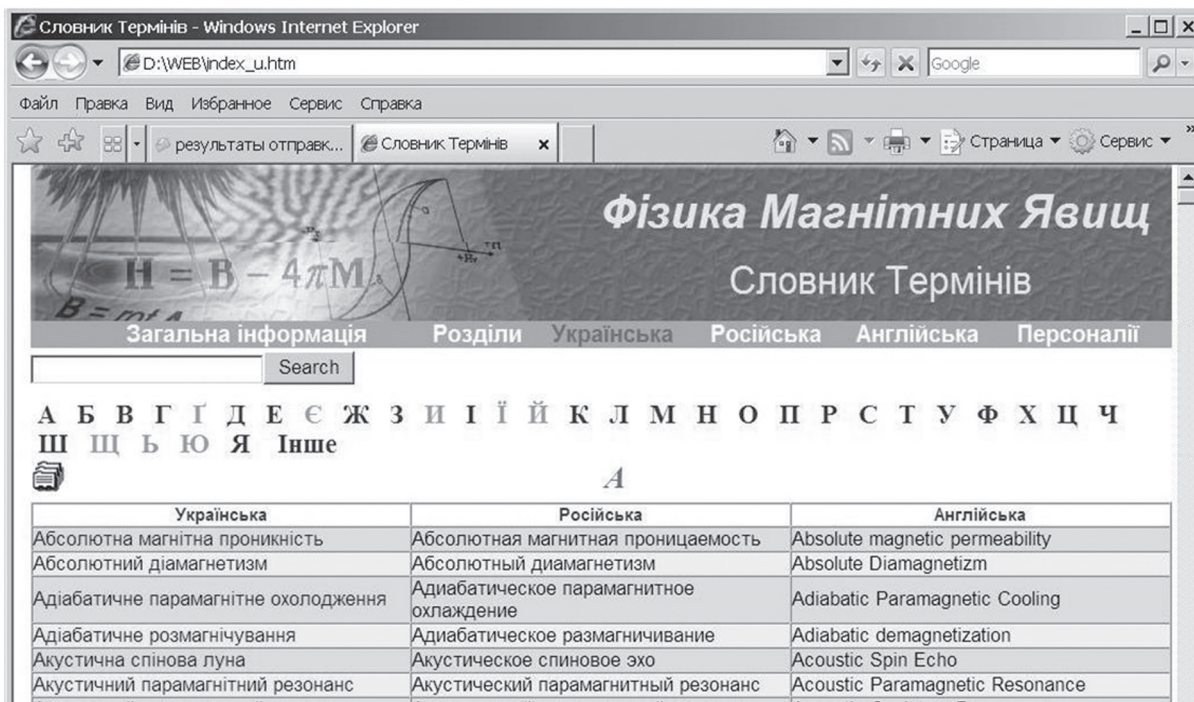


Рис. 5. Інтерфейс користувача – український алфавітний указатель

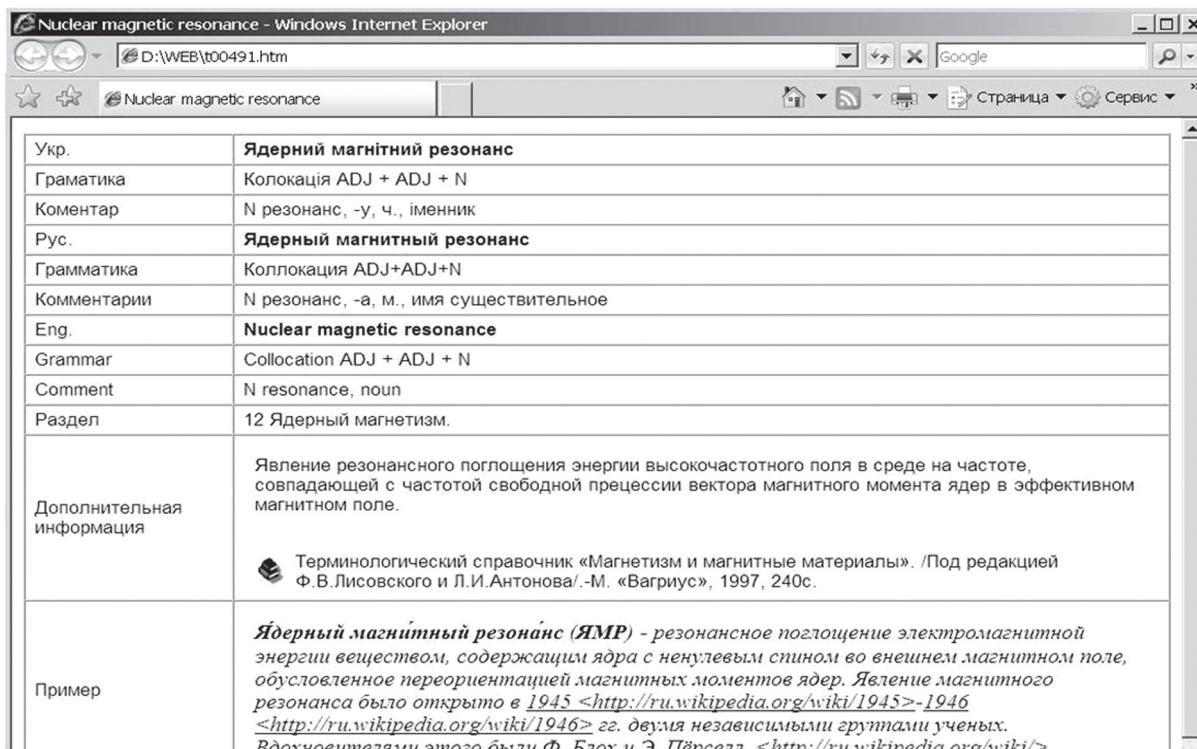


Рис. 6. Інтерфейс користувача. Словарна стаття терміна (частина 1)

Заклучение

Разработанная инструментальная система (редактор) позволяет пополнять и редактировать содержимое лексикографической базы данных, а затем публиковать его в виде веб-сайта, построенного на основе HTML и Java-script. Этот сайт представляет собой пользовательскую версию словаря и может быть размещен в Интернете или записан на

лазерный диск. Редактор и пользовательская версия имеют двуязычный интерфейс (русский, украинский). Программный интерфейс и алгоритм описания терминов не зависят от предметной области и могут быть использованы для создания трехязычных словарей других предметных областей.

Авторы приносят глубокую благодарность директору Института магнетизма НАН и МОН Ук-

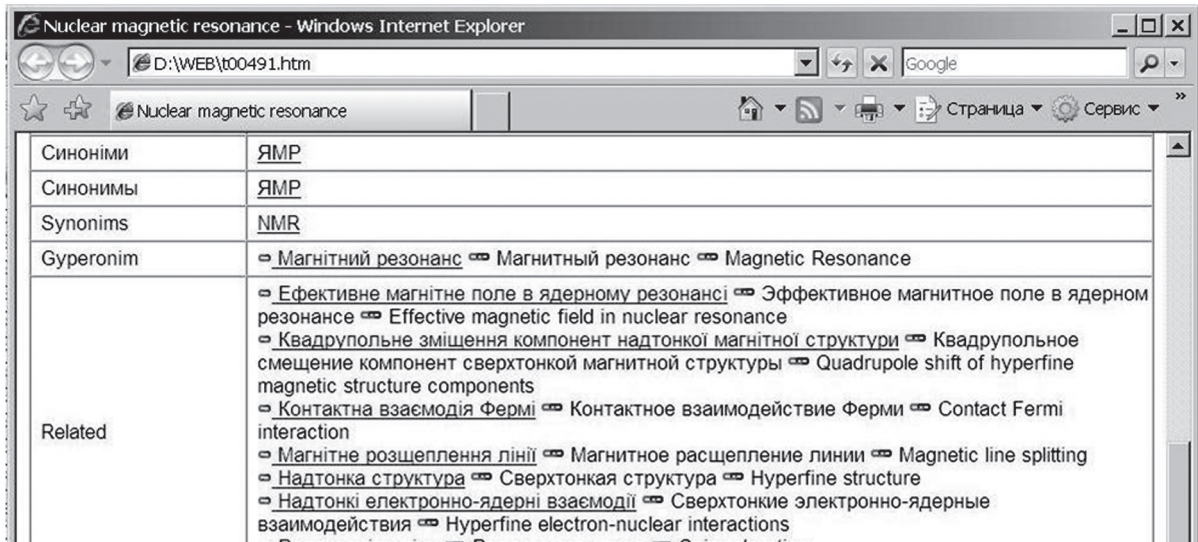


Рис. 7. Інтерфейс користувача. Словарна стаття терміна (частина 2 – карта терміна)

раины академику НАН Украины В.Г.Барьяхтару и директору Украинского языково-информационного фонда НАН Украины член-корреспонденту В.А.Широкову за постоянный интерес к данному проекту и ценные советы.

Список литературы: 1. Бержанский, В.Н. Формирование онтологии на материале терминосистемы физики магнитных явлений [Текст] / В.Н.Бержанский, Е.В.Потапова, О.Ю. Салюк // Прикладна лінгвістика та лінгвістичні технології: MegaLing 2007: сб.науч. тр./ отв. ред. В.А.Широков – К.: Довіра, 2008. С.61-67. 2. Барьяхтар, В.Г. Ментальные пространства физики магнетизма в информационно-коммуникативном обществе знаний [Текст]/ В.Г. Барьяхтар, В.Н.Бержанский, С.С.Дикарева // MegaLing'2008 Горизонти прикладної лінгвістики та лінгвістичних технологій: тез. науч. конф. (вересень 2008) / отв. ред. В.А.Широков. – Сімферополь, ДИ-АЙ-ПИ, 2008. – С. 350 – 352. 3. Широков, В.А. Інформаційна теорія лексикографічних систем [Текст] / В.А.Широков. - Київ, Довіра, 1998. – 331с. 4. Широков, В.А. Элементы лексикографии [Текст] / В.А. Широков. – К.: Довіра, 2005. 5. Герд А.С. Научно-техническая лексикография [Текст] / А.С. Герд // Прикладное языкознание. – Спб, 1996. – 308 с. 6. Міжнародна наукова конференція “Проблеми української термінології” [Електронний ресурс] / Національний університет “Львівська політехніка”. – Режим доступа : http://www.lp.edu.ua/tc.terminology/TK_konf.htm. – 06.06.2009 г. – Загл. с экрана. 7. Технічний комітет стандартизації науково-технічної термінології [Електронний ресурс] / МОН України. – Режим доступа : <http://www.lp.edu.ua/tc.terminology> 25.03.2009 г. – Загл. с экрана. 8. Українська термінологія і сучасність [Текст]: збірник наукових праць. Випуск VII / Відп. ред. Л.О.Симоненко. – Київ: КНЕУ, 2007. – 207 с. 9. Физика твердого тела. Энциклопедический словарь: в 2 т. [Текст] / Под ред. В. Г. Барьяхтара. Т. 1.– Киев: Наукова думка, 1996. – 656 с., Т. 2. – Киев: Наукова думка, 1998. – 648 с. 10. Терминологический справочник «Магнетизм и магнитные

материалы» [Текст] / Под ред. Ф. В. Лисовского и Л. И. Антонова. – М.: «Вагриус», 1997. – 240 с.

Поступила в редколлегию 06.10.2009

УДК 004.5:81'374.82

Інструментальна лексикографічна система в галузі фізики магнітних явищ / С. С. Дікарева, В. Н. Бержанський, С. Н. Полулях, Є. В. Потапова // Біоніка інтелекту: наук.-техн. журнал. – 2009. – № 2 (71). – С. 48-53.

Розроблена інструментальна система (редактор) дозволяє поповнювати й редагувати вміст лексикографічної бази даних тримовного (російська-українська-англійська мови) словника термінів з фізики магнітних явищ, а потім публікувати вміст словника у вигляді веб-сайта, побудованого на основі HTML і Java-script. На основі методів когнітивної й прикладної лінгвістики розроблена оригінальна лексикографічна параметризація терміна. Програмний інтерфейс і алгоритм опису термінів не залежать від предметної галузі й можуть бути використані для створення тримовних словників інших предметних галузей.

Табл. 2. Іл. 7. Бібліогр.: 10 найм.

UDK 004.5:81'374.82

Instrument lexicographical system in the field of physics of magnetic phenomena /S.S. Dikareva, V.N. Berzhanskiy, S.N. Polulyakh, E.V. Potapova // Bionics of Intelligence: Sci. Mag. – 2009. – № 2 (71). – P. 48-53.

The developed instrument system (editor) makes it possible to supplement and to edit contents of the lexicographical data base of the three lingual (Russian-Ukrainian- English) dictionary of terms in physics of magnetic phenomena and then publish contents of the dictionary in a form of the HTML and Java-script Web site. On the basis of cognitive and applied linguistics methods the original lexico-graphical parametrization of term was developed. The program interface and the algorithm of the terms description does not depend on subject area and they can be used for creating the three lingual dictionaries of other subject areas.

Fig. 7. Tabl. 2. Ref.: 10 items.

УДК 004.421



ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ ІНВЕРСНОГО КОНТЕКСТНО-АСОЦІАТИВНОГО МЕТОДУ АВТОМАТИЗОВАНОЇ ОРФОКОРЕКЦІЇ

Т.М. Заболотня

НТУУ «КПІ», м. Київ, Україна, tatiana104@yandex.ru

Запропоновано алгоритм, який реалізує інверсний контекстно-асоціативний метод автоматизованого виправлення орфографічних помилок. Експериментально досліджено ефективність використання інверсного контекстно-асоціативного методу за критеріями швидкодії та точності роботи відповідного програмного забезпечення. Визначені рекомендовані значення параметрів алгоритму виправлення помилок, встановлення яких забезпечує ефективну роботу програмного орфокоректора.

ЛІНГВІСТИЧНЕ ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ, АВТОМАТИЗОВАНА ОРФОКОРЕКЦІЯ ПРИРОДНОМОВНИХ ТЕКСТОВИХ ОБ'ЄКТІВ

Вступ

Від моменту виникнення систем автоматичної обробки текстів (АОТ) і до теперішнього часу їх важливою задачею є перевірка орфографії в текстових даних. З огляду на постійне збільшення обсягу електронної документації, забезпечення ефективної комп'ютеризованої обробки природномовних об'єктів неможливе без підвищення продуктивності програмних орфокоректорів, особливо за критеріями точності та швидкості обробки вхідних даних [1].

Дослідження останніх вітчизняних та зарубіжних робіт у галузі автоматичної обробки природномовних текстів показало, що підвищенню ефективності автоматизованої орфокорекції значною мірою сприяє модифікація існуючих алгоритмів виправлення помилок у напрямку залучення контекстної інформації та лексико-семантичних словникових ресурсів для її обробки. Проведення аналізу асоціативних зв'язків між словами уможливило коректне звуження множини гіпотез, які мають бути оброблені під час визначення варіантів виправлення, що позитивно впливає на ефективність функціонування систем АОТ в цілому. Але на даний момент в існуючих програмних засобах орфокорекції методика врахування контексту спотвореного слова за допомогою семантичного інструментарію є недостатньо розробленою, а отже, дослідження даного питання видається актуальним.

1. Постановка задачі

Метою даної роботи є експериментальне дослідження ефективності інверсного контекстно-асоціативного методу автоматизованої орфокорекції, сформульованого та теоретично обґрунтованого у [2].

У відповідності до поставленої мети **задачами дослідження** є:

- розробка алгоритму визначення варіантів виправлення орфографічних помилок, що реалізує інверсний контекстно-асоціативний метод орфокорекції;

- дослідження ефективності роботи побудованого програмного коректора за критеріями точності та швидкості обробки текстових даних.

2. Основні теоретичні відомості щодо забезпечення автоматизованої контекстно-асоціативної орфокорекції текстових даних

2.1. Схема автоматизованого виправлення орфографічних помилок

Загальноприйнята схема автоматизованої корекції спотвореного слова [3, 4] передбачає реалізацію:

- етапу висунення гіпотез (вірогідних варіантів виправлення помилки) і

- етапу перевірки гіпотез та ухвалення однієї (декількох) з них як виправлення, що пропонується програмою до внесення.

На першому етапі послідовно виконуються підбір первинної множини варіантів виправлення із словника та попередня фільтрація її вмісту; на другому етапі – перевірка гіпотез на подібність до спотвореного слова за певними критеріями. Методи висунення та перевірки гіпотез виправлення по своїй суті передбачають фільтрацію заданої множини слів, адже в результаті застосування кожного з них відбувається звуження поточної множини варіантів корекції спотвореного слова.

З огляду на вищенаведене будемо вважати процес визначення варіантів виправлення таким, що складається із застосування композиції функцій фільтрації до множини слів W_{dict} , яка міститься у словнику [2]. Фільтром множини W_x називатимемо функцію $filter: W_x \rightarrow W_y$, за допомогою якої з елементів W_x формується множина слів W_y , що відповідають певному критерію схожості зі спотвореним словом ($W_y \subseteq W_x$) [2].

$$filter: W_x \rightarrow W_y, W_y \subseteq W_x, \quad (1)$$

де W_x, W_y – множини природномовних слів.

Позначимо послідовність фільтрів як

$$\begin{aligned} FILTERS = \\ = f_m \circ f_{m-1} \circ \dots \circ f_i \circ \dots \circ f_2 \circ f_1: W_{dict} \rightarrow W_{retr}, \quad (2) \\ m > 1, \end{aligned}$$

де $f_i: W_{i-1} \rightarrow W_i$ ($i = 2, \dots, m$) – фільтр множини слів, отриманої у результаті виконання f_{i-1} (для f_1

– множини W_{dict}); W_{retr} – множина слів, визначених коректором як можливі варіанти виправлення за ознаками їх близькості до спотвореного слова.

Під *точністю* машинної орфографічної корекції спотвореного слова будемо розуміти відношення числа запропонованих орфокоректором вірних варіантів написання слова (це одиниця або нуль) до загальної кількості підібраних слів.

$$PRECISION = \frac{|W_{corr} \cap W_{retr}|}{|W_{retr}|}, \quad (3)$$

де W_{corr} – множина вірних варіантів корекції спотвореного слова у словнику [2].

Відповідно до формули (3), для того, щоб досягти високого показника точності роботи орфокоректора, необхідно, по-перше, забезпечити постійне входження вірного слова до сформованого масиву варіантів виправлення ($|W_{corr} \cap W_{retr}| = 1$), а по-друге, – зменшити загальну кількість слів, які пропонуються програмою як найбільш вірогідні кандидати виправлення помилки (W_{retr}).

2.2. Місце семантичної складової у схемі виправлення орфографічних помилок

Згідно з класичною послідовністю обробки текстових даних (морфологічний, синтаксичний та семантичний аналіз) семантичні фільтри набору гіпотез мають стояти наприкінці композиції *FILTERS* (див.(2)). Відповідно ж до сучасних тенденцій щодо зміни загальноприйнятого порядку етапів обробки текстів [3, 5] можливим є підвищення ефективності програмного орфокоректора у випадку перенесення перевірки гіпотез за семантичними критеріями ближче до початку *FILTERS*.

Теоретичні дослідження впливу зміни місця семантичної складової у схемі орфокорекції на точність та швидкодію програмного коректора показали такі результати [2]:

– зміна розташування семантичного фільтру f_{cont} у схемі орфокорекції не впливає на точність роботи коректора (*PRECISION*);

– застосування семантичного фільтру f_{cont} під час висунення гіпотез виправлення забезпечує більш швидке отримання результату роботи орфокоректора, ніж його використання для остаточної перевірки множини гіпотез за умови виконання нерівності $t_{f_{cont}(W_{dict})} \leq t_{f_{cont}(W_m)}$, де W_m – результат фільтрації множини W_{dict} із використанням композиції функцій $f_m \circ \dots \circ f_2 \circ f_1 : W_{dict} \rightarrow W_m$ (для $i=1$ роль W_m виконує безпосередньо W_{dict}).

У даному випадку f_{cont} – це функція, яка застосовується для відбору із вихідного набору слів тих словоформ, що узгоджені з контекстним оточенням спотвореного слова за змістом.

Факт підвищення швидкості пошуку варіантів виправлення при збереженні точності роботи орфокоректора у разі перенесення семантичної скла-

дової схеми корекції на її початковий етап [2] і ліг в основу інверсного контекстно-асоціативного методу автоматизованої орфокорекції.

2.3. Інверсний контекстно-асоціативний метод та узагальнений алгоритм автоматизованого виправлення орфографічних помилок

Для забезпечення орфокоректора даними про семантично-асоціативні зв'язки між словами природної мови використано онтологічний словниковий ресурс у формі орієнтованого графу G , вершинами якого є лексеми природної мови, поєднані лексико-семантичними відношеннями [5, 6].

З огляду на те, що із збільшенням дистанції між вершинами графу словника сила семантичного зв'язку між ними швидко зменшується [5], слова, відстань від яких до контексту за структурою графу G перевищує певний поріг *maxdist*, вважатимемо нескінченно віддаленими від нього і не включатимемо їх до множини гіпотез.

Отже, згідно з інверсним контекстно-асоціативним методом процедура автоматизованої орфокорекції являє собою послідовне виконання таких дій:

- 1) встановлення радіуса пошуку r рівним мінімально припустимому значенню;
- 2) висунення гіпотез виправлення за ознакою семантичної близькості до контекстного оточення спотвореного слова;
- 3) перевірка гіпотез виправлення на подібність до спотвореного слова за формальними ознаками;
- 4) збільшення радіуса пошуку гіпотез виправлення та перехід до п.2 даного методу у випадку, якщо, по-перше, $r < \text{maxdist}$, а по-друге, якщо на заданій відстані r від вершин графу словника, котрі відповідають словам контексту, не знайдено жодного слова, яке задовольнило б усім критеріям схожості зі спотвореним словом; в іншому разі – закінчення пошуку варіантів виправлення [2].

Важливою особливістю методу є його ітераційний характер. Він дозволяє зменшити кількість дій щодо обробки слів під час орфокорекції, і тим самим підвищити швидкість її виконання.

На основі інверсного контекстно-асоціативного методу організації роботи орфокоректора процес корекції спотворених слів пропонується упорядкувати таким чином, як це описано нижче (див. рис. 1).

Крок 1. Визначення елементів контекстного оточення, які будуть враховуватися автокоректором для виправлення спотвореного слова: від того, які слова будуть відібрані з контексту на цьому кроці, залежить хід подальшої корекції помилок та ефективність застосування методу в цілому (див. блоки 2, 3, 6 рис. 1).

Множина слів контексту $W_{context}$ може містити слова $W_{dict_context} \subset W_{dict}$ та слова $W_{error_context}$, яких у словнику немає (будемо вважати їх такими, що містять помилку).

Якщо $W_{error_context} \neq \emptyset$ (див. блок 3 рис. 1), подальший хід корекції помилок зводиться до пере-

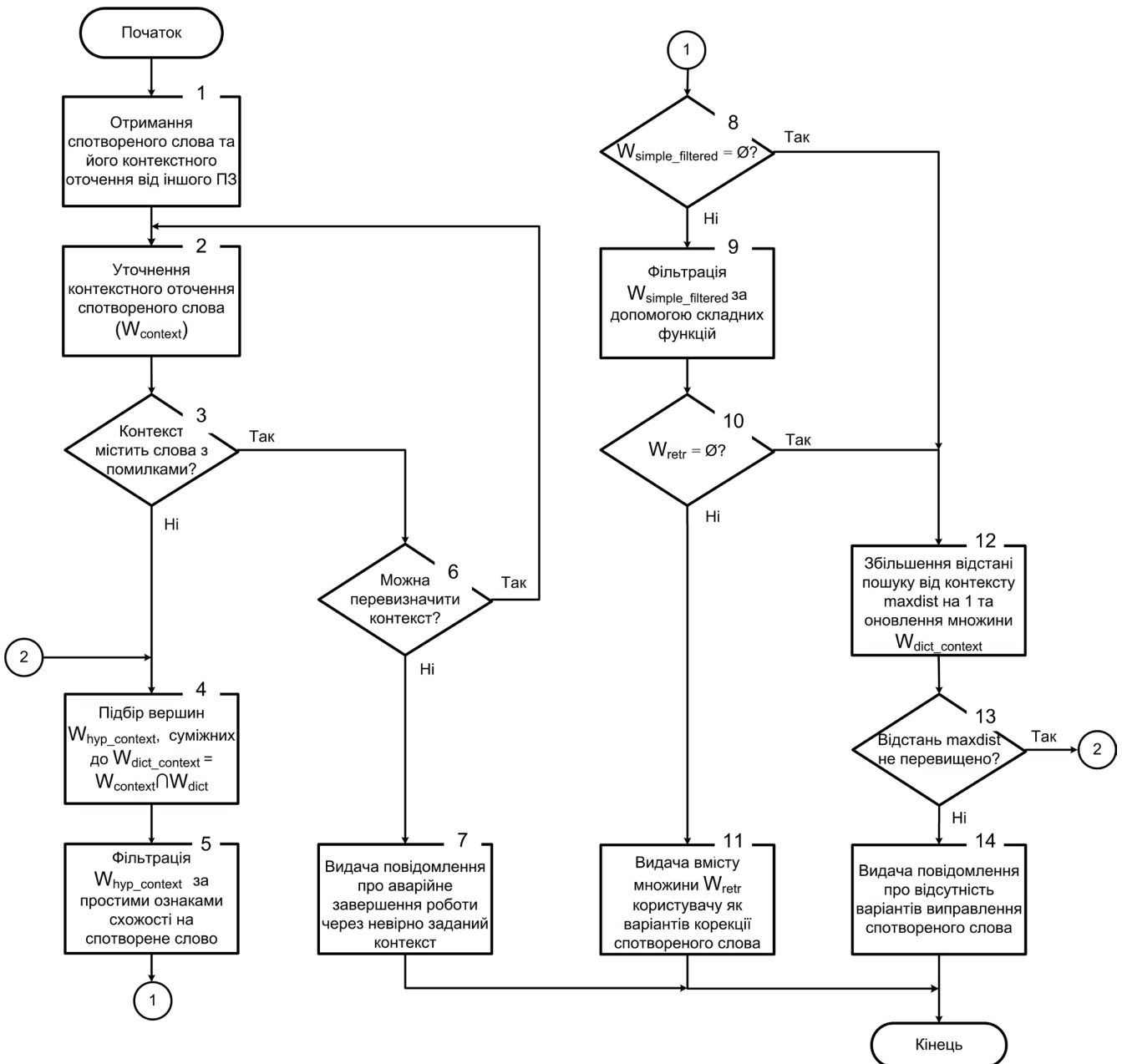


Рис. 1. Схема узагальненого алгоритму визначення варіантів виправлення

вірки можливості перевизначення контексту (див. блоки 2, 6 рис.1). Для систем реального часу будь-яка додаткова обробка слів контекстного оточення, які містять помилку, є неприпустимою. У такому разі умова $W_{error_context} \neq \emptyset$ є критерієм завершення процесу корекції конкретного слова.

Крок 2. Проведення підбору слів $W_{hyp_context}$, які є суміжними за структурою словника до елементів множини $W_{dict_context}$ (див. блок 4 рис.1).

$$\begin{aligned}
 W_{hyp_context} &\subset W_{dict} \setminus W_{dict_context} \\
 \forall w \in W_{hyp_context} \exists v \in W_{dict_context} : (v, w) \in E
 \end{aligned}
 \quad (4)$$

Відібрані таким чином слова є близькими за змістом до слів контекстного оточення спотвореного слова $error_word$. Але крім семантичної складової, необхідно враховувати і формальний бік схожості слів. Тому наступним кроком є

Крок 3. Фільтрація $W_{hyp_context}$ за простими ознаками схожості на спотворене слово (див. блок 5 рис.1). Відбір слів відбувається за такими критеріями, які дозволяють швидко визначити, які слова точно не є варіантами виправлення спотвореного слова. Це можуть бути: критерій альфакоду, критерій довжини слова, критерій збігу першої літери слова тощо [4].

Крок 4. Ухвалення гіпотез(и) як підказки до виправлення (блок 9 рис.1). Функції, застосування яких до множини гіпотез передбачено даним кроком, дозволяють однозначно визначати слова, які схожі (згідно з обраними ознаками) на спотворене слово і можуть бути вірним варіантом його написання.

Якщо пошук серед сусідніх до контекстного оточення слів не дав позитивних результатів, коло можливих варіантів можна розширити

$W_{dict_context} = W_{dict_context} \cup W_{hyp_context}$ і повторити кроки 2-4 алгоритму знову (див. блоки 8, 10, 12 рис.1). Даний процес триватиме, поки не будуть отримані позитивні результати ($PRECISION > 0$), або поки відстань до контексту спотвореного слова, на якій проводиться пошук варіантів виправлення у словнику, не перевищить припустимий поріг $maxdist$ (блок 13 рис.1). Ітеративний характер алгоритму, що пропонується, сприяє підвищенню швидкості роботи орфокоректора, оскільки аналізується виключно необхідний мінімум слів з околу контексту.

Перш, ніж проводити практичне дослідження ефективності методу орфокорекції з урахуванням семантично-асоціативних зв'язків слів природної мови, необхідним є обрання способу кількісного оцінювання відповідності за змістом гіпотези виправлення та контекстного оточення спотвореного слова.

2.4. Кількісне оцінювання семантичної близькості гіпотези виправлення контексту спотвореного слова

У даній статті дослідження ефективності інверсного контекстно-асоціативного методу орфокорекції проводиться із застосуванням двох мір семантичної близькості варіанта виправлення контексту спотвореного слова під час автоматизованої орфокорекції:

– *Міри 1*, яка використовується у практиці машинного перекладу та синтактико-семантичного аналізу текстів [5, 7] і обчислюється як обернена величина сумарної довжини найкоротших шляхів від даного слова до усіх слів контекстного оточення за структурою лексико-семантичного ресурсу;

– *Міри 2*, запропонованої у [6], яка дає змогу оцінювати близькість слова та контексту з огляду на наявність зв'язку цього слова хоча б з одним елементом його оточення і є оберненою величиною мінімальної з довжин найкоротших шляхів від даного слова до елементів контексту.

Основна відмінність *Міри 1* та *Міри 2* полягає у тому, що для обчислення сумарної довжини найкоротших шляхів від слова до елементів контексту у будь-якому випадку необхідно задіяти весь окіл контексту на відстані $maxdist$ (адже, наприклад, слово, яке має шляхи до контексту довжиною 4 та 1 згідно з *Міри 1*, є більш семантично близьким до нього, ніж слово, яке має шляхи довжиною 3 та 3). Процедуру ж визначення слів із *Мірою 2* можна організувати таким чином (вибрати такий напрям відбору слів із словника), що для її успішного завершення достатньо обробити вершини графу G , які лежать у радіусі $R_{min} = \min(R)$ від контексту, де:

$$R = \{r \in \mathbb{N} \mid (r \leq maxdist) \wedge \wedge(((\bigcup_i \sigma x_i^r \cup \sigma x_i^{-r}) \cap W_x) \neq \emptyset)\}, \quad (5a)$$

$$R = \{r \in \mathbb{N} \mid (r \leq maxdist) \wedge \wedge(F_A(\bigcup_i \sigma x_i^r \cup \sigma x_i^{-r}) \neq \emptyset)\}. \quad (5b)$$

У формулі (5a) $W_x \subset W_{dict}$ – результат попереднього відбору гіпотез, які потрібно перевірити на семантичну відповідність контексту. Якщо ж процес корекції починається із пошуку семантично пов'язаних з контекстом слів, доцільно використовувати формулу 5б, де F_A – абстрактна функція перевірки елементів заданої множини слів на відповідність іншим критеріям схожості із спотвореним словом. В обох записях x_i – слово контекстного оточення; $\sigma x_i^r, \sigma x_i^{-r}$ – відображення r -го ступеня вершини x_i графу G (пряме та зворотне).

3. Аналіз результатів експериментальних досліджень ефективності інверсного контекстно-асоціативного методу автоматизованої орфокорекції

Для експериментальної апробації ефективності інверсного контекстно-асоціативного методу орфокорекції використано масив словосполучень, які є запитами користувача до інформаційно-пошукової системи [8] та характеризуються різною потужністю множин слів, що складають контекст спотвореного слова; різною кількістю помилок, припущених у слові (1 та 2 помилки); різною силою семантичного зв'язку контексту із спотвореним словом.

Для підтвердження досягнення найкращих показників роботи коректора за умови застосування фільтрів до вмісту лексико-семантичного словника в *інверсному* порядку проаналізуємо результати функціонування відповідного програмного забезпечення у випадках, коли алгоритмом його роботи передбачено:

1) використання *Міри 1* та *Міри 2* сили семантичного зв'язку контексту спотвореного слова та варіантів виправлення;

2) використання семантичної функції f_{cont} на початку та наприкінці послідовності фільтрів (*прямий* та *інверсний* порядок фільтрації);

3) проведення спроб виправлення *одно-* та *двократних* помилок;

4) встановлення радіуса околу контекстного оточення спотвореного слова за структурою графа словника рівним від 1 до 5 переходів ($maxdist = 1..5$).

Показники ефективності роботи створеного орфокоректора порівняно із аналогічними показниками модуля, вбудованого до пакету MS Word, функціональність якого найчастіше використовується для обробки текстів. Робота коректора базується на використанні WordNet 3.0 для англійської мови, оскільки кінцевого варіанта локалізації словника даного формату для української мови поки що не існує.

Різноманітність вхідних текстових даних створює незручності при оцінюванні часових показників роботи коректора в секундах (чи інших абсолютних одиницях виміру часу). Тому результати вимірювань часу орфокорекції наведено у вигляді

відношення часу роботи коректора при заданих вихідних умовах (t) до відповідного показника роботи коректора, коли він виконує виправлення *однократних* помилок із застосуванням фільтрів вмісту словника в *прямому* порядку ($t_{\text{баз}}$).

$$K_t = t * 100 / t_{\text{баз}}. \quad (6)$$

Першим показником ефективності роботи програмних засобів орфокорекції, на основі значень якого можна робити висновки щодо швидкості виправлення помилок, є час, необхідний для виконання семантичної функції f_{cont} . У статті [2] показано, що для забезпечення більш швидкої роботи коректора при перенесенні f_{cont} до початку послідовності *FILTERS*, необхідним є виконання нерівності $t_{f_{\text{cont}}(W_{\text{dict}})} \leq t_{f_{\text{cont}}(W_m)}$, де W_{dict}, W_m – множини слів, які містить словник, та слів, які передаються на обробку функції f_{cont} , якщо остання стоїть наприкінці послідовності фільтрів. Перевіримо справедливості даного твердження. На графіках 2а та 2б наведені результати вимірювання часу виконання f_{cont} , в основі роботи якої лежать *Mipa 1* та *Mipa 2* семантичної близькості варіантів виправлення та контексту спотвореного слова.

Як можна побачити на графіку 2а, у межах виправлення однократних помилок при обробці околу контексту на відстані менше трьох переходів за структурою графа словника виконання функції f_{cont} є швидшим для інверсного порядку фільтрів. Дана закономірність підтверджує теоретичні відомості щодо стрімкого зменшення сили семантичного зв'язку при збільшенні дистанції між словами, яке приводить до зростання кількості нерелевантних оброблюваних гіпотез виправлення та погіршення швидкості роботи програмних засобів орфокорекції.

При спробі виправлення двократних помилок (за умови застосування фільтрів вмісту словника в інверсному порядку) час роботи f_{cont} практично збі-

гається з аналогічними показниками, отриманими для даної функції у межах корекції слів з однократною помилкою. При застосуванні фільтрів у прямому порядку час виконання f_{cont} для виправлення двократних помилок є більшим, ніж час роботи f_{cont} для корекції слів з однократними помилками.

На відміну від *Mipa 1*, використання *Mipa 2* семантичної близькості варіантів виправлення та контекстного оточення спотвореного слова при проведенні фільтрації вмісту словникового ресурсу в інверсному порядку (див. рис. 2б) приводить до покращення часових характеристик роботи функції f_{cont} (порівняно із застосуванням f_{cont} наприкінці процесу фільтрації) незалежно від радіуса пошуку варіантів виправлення у графі словника.

Робота f_{cont} (з *Mipoyu 2*) при виправленні двократних помилок у разі застосування фільтрів в інверсному порядку займає приблизно однаковий час з роботою семантичної функції для корекції слів, які мають однократні помилки. Час же виконання f_{cont} за умови прямого порядку фільтрації у випадку виправлення двократних помилок є більшим, ніж при виправленні однократних спотворень.

Таким чином, можна зробити висновок про те, що у разі перенесення на етап висунення гіпотез семантична функція виконується швидше, ніж тоді, коли вона входить до послідовності фільтрів етапу перевірки гіпотез (для *Mipa 1* та *Mipa 2*). Отже, умова $t_{f_{\text{cont}}(W_{\text{dict}})} \leq t_{f_{\text{cont}}(W_m)}$ виконана, і загальний час проведення орфокорекції повинен бути меншим при застосуванні фільтрів до вмісту словникового ресурсу в інверсному порядку. На основі рис. 3 визначимо, чи виконується дане твердження.

Якщо порівняти графіки, подані на рис. 3 та 2а, можна помітити, що вони мають багато спільного: орфокоректор працює швидше у разі інверсної послідовності застосування фільтрів до вмісту словника на відстані пошуку гіпотез, яка не пере-

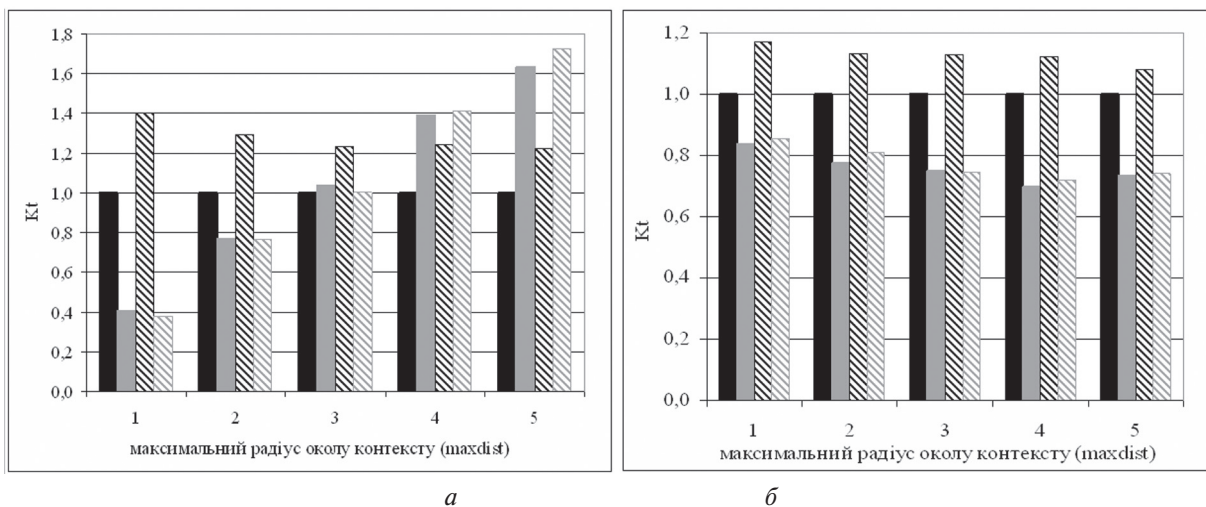


Рис. 2. Графік залежності відносного часу виконання f_{cont} від особливостей реалізації алгоритму орфокорекції (для *Mipa 1* (а) та *Mipa 2* (б)):

- – прямий порядок, 1 помилка; ▒ – інверсний порядок, 1 помилка;
- ▨ – прямий порядок, 2 помилки; ▩ – інверсний порядок, 2 помилки

вище 4 переходи за структурою графа; час, який витрачається на проведення спроби виправлення двократних помилок, є практично однаковим з часом, потрібним для визначення варіантів виправлення однократних помилок (для інверсного порядку фільтрації).

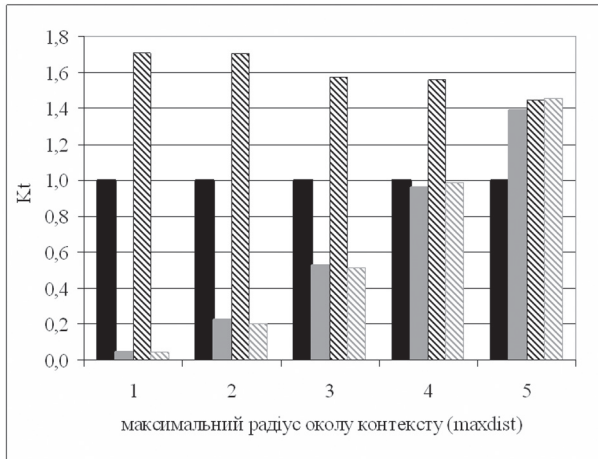


Рис. 3. Графік залежності відносного загального часу орфококорекції від особливостей реалізації її алгоритму (для *Mipu 1*):

- – прямий порядок, 1 помилка; ■ – інверсний порядок, 1 помилка; ▨ – прямий порядок, 2 помилки; ▩ – інверсний порядок, 2 помилки

Таким чином, виходячи з наведеного на рис. 3 графіка, можна стверджувати, що перенесення семантичної функції (яка використовує *Mipu 1* близькості слів заданому контексту) на початок послідовності фільтрів вмісту словника сприяє підвищенню загальної швидкодії орфококоректора.

Всі тенденції, згадані вище для рис. 3а, також мають місце і у результатах дослідження загального часу орфококорекції за умови використання *Mipu 2*.

Порівнюємо тепер найкращі часові показники роботи орфококоректора із застосуванням кожної з двох мір семантичної близькості заданого слова до контексту та аналогічний показник роботи модуля корекції помилок, який входить до складу редактора MS Word (див. рис. 4). За $t_{\text{баз}}$ (див. (6)) прийнято час орфококорекції із застосуванням *Mipu 1* семантичної близькості слів до контексту. Як видно з рис. 4, використання *Mipu 2* забезпечує найшвидшу роботу орфококоректора, а MS Word демонструє гірший результат.

Таким чином, на основі рис. 2–4 можна зробити висновки про досягнення найкращих часових показників роботи програмних засобів орфококорекції за таких умов їх функціонування:

- при виконанні фільтрації вмісту словника в *інверсному* порядку;
- при залученні для реалізації f_{cont} *Mipu 2* семантичної близькості варіантів виправлення та контексту спотвореного слова;
- при неперевищенні радіусом околу контексту, в якому здійснюється пошук варіантів виправлення, значення $\text{maxdist}=3$.

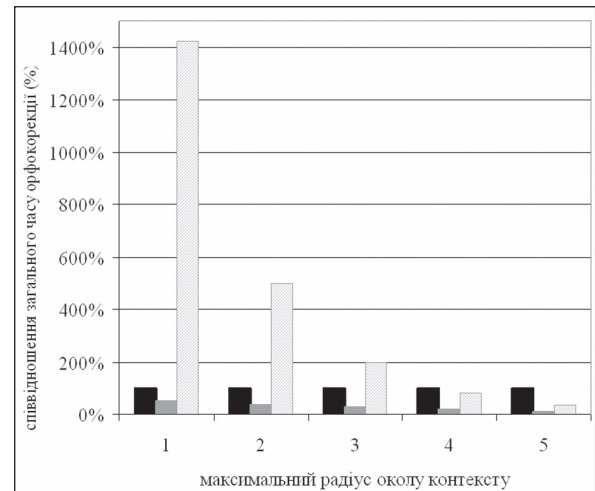


Рис. 4. Графік залежності відносного загального часу орфококорекції від особливостей реалізації її алгоритму (для *Mipu 1*): ■ – *Mipu 1*; ■ – *Mipu 2*; ▨ – MS Word

Крім того, показано, що при виконанні наведених вище умов роботи орфококоректора час на спробу виправлення однократних та двократних помилок є майже однаковим.

Проаналізуємо тепер, яким чином змінюється точність орфококорекції в залежності від різних вихідних умов проведення останньої (див. рис. 5).

Практичні дослідження підтвердили справедливості теоретично доведеної комутативності композиції функцій *filter*: від зміни порядку застосування фільтрів точність орфококорекції не змінюється. Разом з тим, очевидно є перевага розроблених програмних засобів орфококорекції за критерієм точності над модулем виправлення помилок MS Word. Найімовірнішою причиною таких результатів є припущення про те, що останній не реалізує перевірки варіантів виправлення на семантичну узгодженість з контекстом спотвореного слова [1]. Так, наприклад, спотворене слово «*fyrn*» в залежності від контексту розробленим коректором виправляється на «*farm*» (при контексті «*animals grow*»), «*form*» (контекст – «*geometrical*») або «*firm*» (контекст – «*business*»). Модуль орфококорекції MS Word пропонує у всіх трьох випадках однакову множину варіантів виправлення: «*farm, form, firm*».

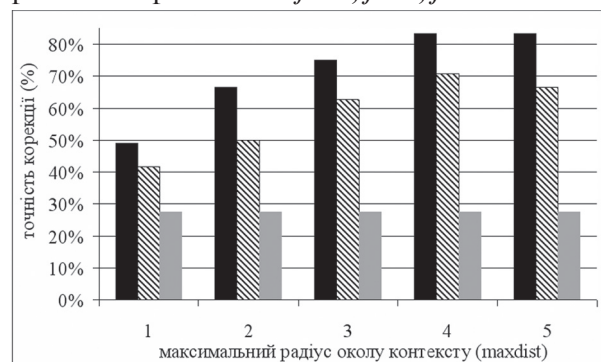


Рис. 5. Графік залежності точності роботи програмних засобів від реалізації алгоритму орфококорекції:

- – виправлення 1 помилки;
- ▨ – виправлення 2 помилок; ■ – MS Word

Факт, що розроблене ПЗ, яке реалізує інверсний контекстно-асоціативний метод автоматизованої орфокоорекції, не характеризується 100% точністю роботи, пояснюється тим, що, з одного боку, при виправленні однократних помилок слова, які мають двократні помилки, залишаються невиправленими, а з іншого боку, спроба виправлення двократних помилок спричиняє додавання до результатів зайвих слів у випадку коорекції слів, які мають однократну помилку.

Можна відмітити ще одну важливу закономірність: точність виправлення помилок перестає зростати при $maxdist > 4$. Отже, з точки зору досягнення найвищої точності роботи орфокооректора збільшувати далі радіус пошуку гіпотез недоцільно.

Якщо порівняти рівні точності орфокоорекції, які забезпечують реалізації алгоритмів на основі *Miru1* та *Miru2* семантичної близькості варіантів виправлення та контексту спотвореного слова, можна зробити висновок про те, що робота кооректора із залученням модифікованого словника (для виправлення одно- та двократних помилок) для обох мір є практично однаково точною (тому у статті наведено графік тільки для *Miru1* — див.рис.5).

Висновки

Таким чином, результати експериментального дослідження ефективності інверсного контекстно-асоціативного методу автоматизованої орфокоорекції показали, що кооректор працює краще за показниками точності та швидкості, якщо алгоритмом його роботи передбачено застосування фільтрів до вмісту словника в *інверсному* порядку (починаючи з семантичного фільтра), а також використання *Miru2* семантичної близькості варіантів виправлення та контексту спотвореного слова для реалізації семантичного фільтра f_{cont} . Крім того емпірично отримані дані показали недоцільність перевищення відстані $maxdist=4$ при проведенні пошуку гіпотез у лексико-семантичному словнику, а також практично однакову швидкість та точність роботи кооректора при спробах виправлення одного та двократних помилок.

Отримані рекомендовані значення параметрів алгоритму виправлення помилок можуть бути уточнені при побудові програмних орфокооректорів для роботи зі спеціалізованими текстовими даними.

Список літератури: 1. Лавошникова, Э.К. О компьютерной коррекции «популярных» ошибок в текстах на русском языке [текст] / Э.К. Лавошникова // НТИ, с. 2. — 2003. — № 9. — С. 28–34. 2. Заболотня, Т.М. Инверсный контекстно-асоциативный метод автоматизованої орфокоорекції [текст] / Т.М. Заболотня, А.Ю. Михайлюк, О.С. Михайлюк // Искусственный интеллект. — 2008. — № 3. — С. 78–88. 3. Kukich K. Techniques for Automatically Correcting Words in Text // ACM Computing Surveys. — 1992. — Vol. 24, № 4. — P. 377–439. 4. Файн, В.С. Машинное понимание текстов

с ошибками / В.С. Файн, Л.И. Рубанов. — М.: Наука, 1991. — 151 с. 5. Марченко, О.О. Алгоритмы семантического анализа природномовных текстов [текст] : Дис. ... канд. физ.-мат. наук: 01.05.01/ КНУ ім. Тараса Шевченка / О.О. Марченко. — К., 2005. — 17 с. 6. Заболотня, Т.М. Оптимізація процесу контекстноорієнтованої орфокоорекції шляхом спрощення обчислення міри семантичної близькості слів [текст] / Т.М. Заболотня // Проблеми інформатизації та управління: Зб. наук. праць. — Вип. 3 (21). — К.: НАУ, 2007. — С. 55–59. 7. Скороходько Э.Ф. Семантические сети и автоматическая обработка текста [текст] / Э.Ф. Скороходько. — К.: Наукова думка, 1983. — 217 с. 8. Заболотня, Т.М. Инверсный контекстно-асоциативный метод та програмні засоби автоматизованої орфокоорекції природномовних текстових об'єктів [текст] : автореф. дис. ... канд. техн. наук: 05.13.05 / НТУУ «КПІ». / Т.М. Заболотня. — К., 2008. — 21 с.

Надійшла до редколегії 12.10.2009

УДК 004.421

Исследование эффективности инверсного контекстно-асоциативного метода автоматизированной орфокоорекции / Т.Н. Заболотня // Бионика интеллекта: науч.-техн. журнал. — 2009. — № 2 (71). — С. 54–60.

В статье предложен обобщенный алгоритм, реализующий инверсный контекстно-ассоциативный метод исправления орфографических ошибок, который обеспечивает повышение эффективности работы орфокооректора по критериям скорости и точности обработки текстовых данных. Проанализированы результаты функционирования соответствующего программного обеспечения в случаях, когда алгоритмом его работы предусмотрено: использование двух разных способов количественной оценки силы семантической связи контекста искаженного слова и вариантов исправления; прямой и инверсный порядок фильтрации содержимого словаря; проведение попыток коррекции одно- и двукратных ошибок; ограничение круга поиска гипотез по структуре графа словаря. Определены рекомендованные значения параметров алгоритма орфокоорекции, установление которых обеспечивает эффективную работу соответствующих программных средств.

Ил. 5. Библиогр.: 8 назв.

UDC 004.421

Investigation of the inverse context-associative method of automatized spelling correction efficiency / T. Zabolotnia // Bionics of Intelligence: Sci. Mag. — 2009. — № 2 (71). — P. 54–60.

In the given article the generalized algorithm that implements the inverse context-associative method of automatized spelling correction, which provides the increase of software corrector's effectiveness on the criteria of speed and accuracy of textual data processing is proposed. The results of the appropriate software functioning in cases where the algorithm of its work provides: usage of two different ways of quantify the strength of semantic relations between context and versions of the distorted word correction; direct and inverse order of dictionary content filtering; an attempt to correct single and double errors; limiting scope of hypotheses search based on the structure of the dictionary graph are analyzed. The recommended option settings of algorithm of the errors correction, establishment of which provides effective work of the spelling corrector are defined.

Fig. 5. Ref.: 8 items.

УДК 004.934.2



ОБ ОДНОМ МЕТОДЕ АВТОМАТИЧЕСКОЙ СЕГМЕНТАЦИИ РЕЧЕВЫХ СИГНАЛОВ

Т.В. Шарий

Донецкий национальный университет, г. Донецк, Украина, tsphere@mail.ru

Исследованы изменения коэффициентов MFCC речевого сигнала на границах фонем. Введена мера кепстральной гладкости сигнала, учет которой, наряду с мерой спектрального перехода, позволил повысить точность фонемной сегментации.

СЕГМЕНТАЦИЯ, КЕПСТР, МЕРА СПЕКТРАЛЬНОГО ПЕРЕХОДА, МЕРА КЕПСТРАЛЬНОЙ ГЛАДКОСТИ, ФОНЕМА

Введение

Сегментация является важным этапом обработки речевого сигнала в научных исследованиях и работе автоматических систем распознавания речи (АСРР). От точности определения границ между фонемными сегментами зависит эффективность всей системы распознавания. На сегодняшний день наиболее точных результатов позволяет добиться ручная сегментация. В то же время, ручная сегментация является дорогостоящей операцией, требующей значительных затрат времени и участия эксперта. Большинство методов *автоматической* сегментации речевых сигналов [1, 2] основываются преимущественно на результатах распознавания фонем и фонетической транскрипции исходного речевого сигнала (контролируемая сегментация, supervised segmentation). Однако результаты распознавания часто ненадежны, а наличие транскрипции возможно только на этапе обучения лексических моделей.

Для определения лишь границ фонем, без необходимости распознавания («слепая» сегментация, «blind» segmentation), существуют простые методы, основанные на величине и скорости изменения определенных акустических характеристик. Две наиболее распространенные характеристики – это коэффициент перехода уровня сигнала через ноль (Zero Cross Rate, ZCR) [3] и мера спектрального перехода (Spectral Transition Measure, STM) [4]. Тем не менее, эксперименты показывают, что для надежной сегментации этих величин недостаточно. На основании результатов проведенных автором экспериментов в работе вводится новая характеристика речевого сигнала – мера кепстральной гладкости (Spectral Smoothness Measure, CSM). Учет этой величины позволяет лучше выделять границы фонем при их коартикуляции в непрерывной речи и, таким образом, повысить точность автоматической сегментации речевых сигналов.

В общем случае, если в качестве минимальной единицы речи выбрана фонема, сегментация речи может производиться на трех уровнях: фонемном, слоговом и словесном. В статье исследуется самый важный и сложный уровень – определение фонем-

ных границ. Для повышения точности фонемной сегментации в работе анализируются те параметры сигнала на фонемных переходах в непрерывной речи, которые могут служить индикаторами границ фонем.

1. Сегментация на основе энергии сигнала и коэффициента ZCR

В качестве одного из первых таких параметров использовался коэффициент перехода уровня сигнала через ноль [3]:

$$ZCR(f) = \frac{1}{L} \sum_{i=(f-1) \times L}^{f \times L - 1} \frac{|\text{sign}(s_{i+1}) - \text{sign}(s_i)|}{2}, \quad (1)$$

где f – номер фрейма; L – количество отсчетов сигнала в данном фрейме; s_i – значение i -го отсчета сигнала.

Главное преимущество коэффициента ZCR состоит в том, что он не зависит от амплитуды сигнала в данном фрейме. В статье [3] вычисление коэффициента ZCR лежит в основе классификации типа «гласный/согласный/пауза» («vowel/consonant/ pause», v/c/p). Классификация v/c/p необходима как базовый этап для выявления кандидатов пауз и границ предложений, а также извлечения просодических признаков речи. Для отделения пауз речи предложен адаптивный расчет уровня фонового шума. Использование конкретного порога энергии сигнала неприменимо, так как параметры окружающей обстановки меняются во времени, особенно в видео сценариях. Ниже подробно описан алгоритм сегментации v/c/p [3].

1. Входной сигнал квантуется во времени на неперекрывающиеся фреймы длиной 20мс, в которых вычисляется энергия сигнала, коэффициент ZCR и частота основного тона. Энергия рассчитывается путем усреднения значений модуля отсчетов сигнала в данном фрейме.

2. Огибающая частоты основного тона и энергии сигнала сглаживается.

3. Уровень шума (NoiseLevel) вычисляется на основе среднего (Mean_En) и дисперсии (Std_En) уровня энергии сигнала:

$$\text{NoiseLevel} = \text{Mean_En} - 0.75 \cdot \text{Std_En}. \quad (2)$$

Аналогичным образом, на основе математического ожидания ($Mean_ZCR$) и дисперсии (Std_ZCR) коэффициента перехода уровня сигнала через ноль, рассчитывается и ZCR -порог (ZCR_dyna):

$$ZCR_Dyna = Mean_ZCR + 0.5 \cdot Std_ZCR. \quad (3)$$

4. Фреймы классифицируются в соответствии со следующими правилами (тип сегмента Consonant – согласный звук, сегмент Vowel – гласный звук, сегмент Pause – пауза):

$$\begin{aligned} &IF\ ZCR > ZCR_Dyna\ THEN \\ &FrameType = Consonant \\ &ELSE\ IF\ Energy < NoiseLevel\ THEN \\ &FrameType = Pause \\ &ELSE\ FrameType = Vowel \end{aligned} \quad (4)$$

5. Величина уровня шума пересчитывается как взвешенная средняя энергия фреймов на каждой границе гласного и фоновых сегментов.

6. Фреймы повторно классифицируются, согласно с правилами (4), с учетом обновленной величины уровня шума. Паузы объединяются путем удаления изолированных коротких согласных. Гласный разделяется надвое в позиции падения уровня энергии, если его длительность слишком велика.

На практике расчет уровня шума и классификация $v/c/p$ выполняется каждые 4-5 секунд, чтобы учесть фоновые изменения.

Проведенные автором эксперименты показывают, что с помощью этого метода можно эффективно выделить шипящие и глухие согласные, однако точность сегментирования многих других звуков остается невысокой. На рис. 1 показаны результаты работы метода на примере фонемной последовательности «прочем». Видно, что хорошо отделены глухие согласные [п] и [ч], в то время как границы между остальными (звонкими) фонемами обнаружены не были. Тем не менее, идея вычисления уровня энергии сигнала для учета фонового шума на практике оказывается важной и полезной.

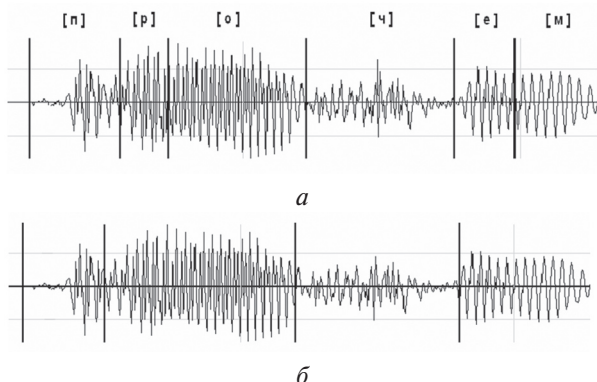


Рис. 1. Результаты сегментации фонемной последовательности «прочем»: ручная сегментация (а); сегментация методом $v/c/p$ (б)

2. Сегментация на основе меры спектрального перехода

Более удовлетворительные результаты, по сравнению с методом $v/c/p$ (ZCR), показывает метод автоматической сегментации, основанный на мере спектрального перехода (Spectral Transition Measure, STM), предложенной в работе [4] следующим образом:

$$STM(f) = \frac{\sum_{i=1}^D a_i^2(f)}{D}, \quad (5)$$

где D – размерность вектора признаков речевого сигнала, скорость изменения элементов которого рассчитывается в соответствии с формулой:

$$a_i(f) = \frac{\sum_{n=-I}^I c_i(f+n) \cdot n}{\sum_{n=-I}^I n^2}, \quad (6)$$

где f – номер текущего фрейма; $c_i(f)$ – i -ый коэффициент MFCC-кепстра речевого сигнала, вычисленного для фрейма f ; I – количество соседних фреймов, слева и справа от текущего, используемых для расчета скорости изменения кепстральных коэффициентов.

Параметр STM может быть интерпретирован как модуль скорости изменения спектра (в данном случае – MFCC-кепстра) сигнала. Согласно [4] положения максимумов величины спектрального перехода связаны с критическими точками, содержащими наиболее важную информацию для восприятия согласных звуков и слогов.

Алгоритм сегментации, основывающейся на параметре STM, включает следующие шаги [4]:

1. Речевой сигнал разбивается на фреймы длительностью 10 мс.

2. В каждом фрейме рассчитывается величина спектрального перехода (5). Параметр I в формуле (6) устанавливается равным 2.

3. Выделяются локальные пики кривой STM, построенной на шаге 2. Эти пики соответствуют фонемным границам.

Недостатком данного метода является то, что он редко определяет границы между гласными и согласными, в случаях выраженной коартикуляции или «плавного» звучания речи [4]. Пример на рис. 2 иллюстрирует данную проблему.

На рис. 2 видно, что коартикуляция фонем [а], [н'] и [о] в начале фразы «О нем существовали самые разные легенды» настолько сильна, что на границах данных фонем не наблюдается существенного изменения меры спектрального перехода.

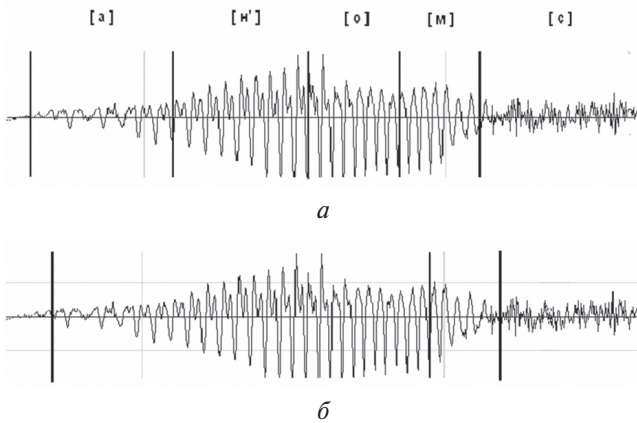


Рис. 2. Результаты сегментации фонемной последовательности «о нём с...»: ручная сегментация (а); сегментация на основе характеристики STM (б)

3. Сегментация с учетом меры гладкости кепстра

С другой стороны, автором экспериментально было установлено, что в подобных ситуациях кепстральная картина может приобретать специфическую форму – кепстр становится «гладким» (рис. 3).

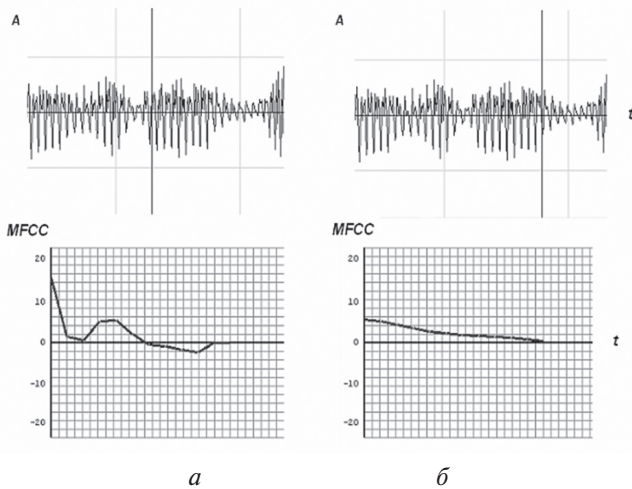


Рис. 3. Пример MFCC-кепстра: на границе между согласным и согласным звуком (а); на границе между согласным и гласным звуком (б)

Для отслеживания указанных ситуаций в работе вводится мера кепстральной гладкости (Spectral Smoothness Measure, CSM):

$$CSM(f) = \begin{cases} \frac{c_{\max}(f) - c_{\min}(f)}{\sum_{i=1}^{D-1} |c_{i+1}(f) - c_i(f)|} & , c_{\max}(f) \neq c_{\min}(f) \\ 1 & , c_{\max}(f) = c_{\min}(f), \end{cases} \quad (7)$$

где f – номер фрейма; $c_i(f)$ – i -ый коэффициент MFCC-кепстра речевого сигнала, вычисленного для фрейма f ; $c_{\max}(f)$ – значение максимального MFCC-коэффициента кепстра; $c_{\min}(f)$ – значение минимального MFCC-коэффициента кепстра; D – размерность кепстра.

Величина CSM стремится к 1 при увеличении степени гладкости кепстра (CSM равняется 1, в случае, если кепстр представляет собой прямую линию).

Предложен новый метод сегментации речевых сигналов, учитывающий меру гладкости кепстра. Метод «STM+CSM» является дополнением к методу STM, описанному в работе [4], а также учитывает уровень фонового шума, по аналогии с методом классификации v/c/p [3]. Алгоритм «STM+CSM» включает следующие действия:

1. Речевой сигнал разбивается на фреймы длиной 10мс.
2. Вычисляется уровень шума (NoiseLevel) на основе среднего (Mean_En) и дисперсии (Std_En) уровня энергии сигнала по формуле (2).
3. Энергия сигнала в каждом фрейме сравнивается с уровнем шума. Если оно не превосходит уровень шума, то переход к следующему фрейму.
4. В текущем фрейме рассчитывается величина спектрального перехода (5).
5. В текущем фрейме рассчитывается величина кепстральной гладкости (7).
6. Выделяются локальные пики кривой STM, построенной на шаге 2. Эти пики соответствуют фонемным границам.
7. Находятся точки, в которых выражение $(CSM(f) - Delta)$ меняет знак. Иначе говоря, находятся точки, в которых кепстр либо становится, либо перестает быть гладким. Найденные точки также считаются соответствующими фонемным границам. Значение порога $Delta$ выбирается равным около 0.7-0.8.

На рис. 4 показаны результаты работы предложенного метода для предыдущего примера фонемной последовательности «о нём с...». Видно, что была обнаружена граница между фонемами [а] и [н'], которая не фиксировалась методом STM. Тем не менее, остается некоторый процент случаев сложно разделяемых (даже вручную) фонем, при которых не происходит изменений ни меры спектрального перехода, ни меры кепстральной гладкости. В данном примере это граница между фонемами [н'] и [о].

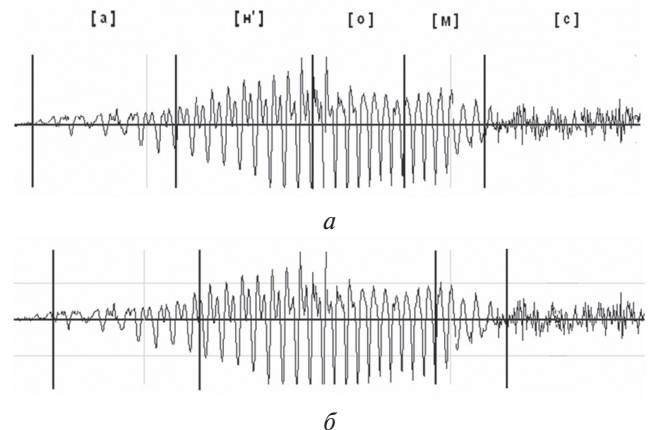


Рис. 4. Результаты сегментации фонемной последовательности «о нём с...»: ручная сегментация (а); сегментация на основе характеристик STM и CSM (б)

4. Сравнение методов автоматической сегментации речевых сигналов

Тестирование методов сегментации проводилось на выборке из 50 файлов 3 дикторов речевой базы VoxForge [5]. Для экспериментов использовалась инструментальная среда редактирования речевых файлов [6]. На первом этапе файлы были размечены вручную. На втором этапе тестировались методы V/C/P, STM и STM+CSM (табл. 1).

Таблица 1

Сравнение показателей точности трех методов автоматической сегментации речи

Метод	Обнаруженные границы (%)	Пропущенные границы (%)	Ложно обнаруженные границы (%)
V/C/P	70,41	29,59	16,76
STM	82,92	17,08	20,19
STM+CSM	90,87	9,13	22,25

Для оценки качества сегментации использовались традиционные показатели – процент верно обнаруженных маркеров границ сегментов, процент пропущенных (не обнаруженных) маркеров и процент ложно обнаруженных маркеров сегментации.

Следует отметить, что в работе маркеры сегментации, расставленные вручную и рассчитанные одним из методов, считаются совпадающими, если разница в их позициях не превышает размера фрейма. Это более мягкое условие по сравнению с условием полного совпадения позиций маркеров. Таким образом, во-первых, учитываются особенности алгоритмов автоматической сегментации речевых сигналов, «разрешающая способность» которых ограничена размером фрейма. Во-вторых, снижается влияние погрешности, с которой эксперт сегментирует сигнал вручную.

Как видно из табл. 1, метод STM+CSM позволяет добиться лучшей точности сегментации по сравнению с методами v/c/p и STM: процент верно обнаруженных границ между фонемами на тестовой выборке составил 90.87%. При этом наблюдается незначительный рост числа ложно обнаруженных маркеров, который можно считать не критичным по отношению к числу верно найденных маркеров. Появление «новых» меток сегментации можно объяснить тем фактом, что при определенных условиях звуки речи, представляемые в фонетической транскрипции одной фонемой, характеризуются несколькими стадиями звучания, в каждой из которых кепстральная картина может изменяться и иметь место пик кривой меры спектрального перехода.

5. Применение метода STM+CSM

Стоит подчеркнуть тот факт, что в табл. 1 приведены показатели эффективности сегментации с точки зрения именно фонемной сегментации, то есть разбиения речевого сигнала на абстрактные единицы речи (фонемы), используемые при транскрипции фразы. Вопрос о минимальных единицах описания речевых сигналов до сих пор остается открытым и спорным среди ученых, но можно утверждать однозначно, что такие единицы, как фонемы, не описываются целиком своими акустическими (спектральными и просодическими) параметрами. Поэтому достичь хороших результатов (98% обнаруживаемых границ и выше) методами «слепой» сегментации невозможно в принципе. С другой стороны, метод STM+CSM, предложенный в работе, полезен с точки зрения определения участков сигнала с медленно изменяющимся кепстром. Результаты экспериментов можно интерпретировать таким образом, что в 90,87% случаев указанные участки совпадают с «реальными» фонемами. В остальных ситуациях эти участки являются либо частью (одной из стадий) звучания фонемы, либо последовательностью двух-трех очень плавно сменяющихся фонем. В любом случае, сегментация представляет собой лишь предварительный этап обработки речевого сигнала, и интерпретация ее результатов во многом зависит от модуля постобработки сигнала (back-end). В контексте сказанного, метод STM+CSM позволяет повысить точность предварительной сегментации, что способствует также росту эффективности распознавания речи в целом.

Важной особенностью является также то, что разработанный подход теоретически можно совместить с методами контролируемой сегментации, например с методом максимального правдоподобия (Maximum Likelihood Segmentation, ML) [1,2]. Данный метод основывается на принципах динамического программирования и для его использования необходимо задать число фонем, на которые необходимо сегментировать определенный участок речевого сигнала. С помощью метода STM+CSM можно точнее оценить количество фонем на данном участке речи и далее с помощью метода ML более качественно определить непосредственно границы между фонемами.

Заключение

Автором исследованы параметры речевого сигнала на границах фонем в непрерывной речи, которые важны для автоматической фонемной сегментации речевых сигналов. Проанализированы такие методы «слепой» сегментации, как v/c/p классификация и сегментация на основе меры спектрального перехода (STM). Введена мера гладкости кепстра для нахождения фонемных переходов и предложен новый метод STM+CSM, учитывающий эту характеристику при сегментации наряду с мерой спек-

трального перехода. Эксперименты подтвердили повышение точности выявления границ фонем. Дальнейшая работа связана с усовершенствованием алгоритма с целью увеличения процента верно определяемых меток, а также уменьшения числа неверно обнаруживаемых лишних меток фонемных сегментов.

Список литературы: 1. Parse Structure and Segmentation for Improving Speech Recognition [text] / W.P. McNeill, J.G. Kahn, D.L. Hillard, M. Ostendorf // IEEE Spoken Language Technology Workshop. – 2006. – P.90–93. 2. On the Robust Automatic Segmentation of Spontaneous Speech [text] / B. Petek, O. Andersen, P. Dalsgaard // Proceedings of ICSLP'96. – 1996. – P.913–916. 3. Speech segmentation without speech recognition [text] / D.Wang, L.Lu, H.J.Zhang // International Conference on Multimedia and Expo (ICME '03). – 2003. – Vol.1. – P.405–408. 4. On the Relation Between Maximum Spectral Transition Positions and Phone Boundaries [text] / S. Dusan, L.R. Rabiner // Proceedings of ICSLP'06. – 2006. – P.17–21. 5. VoxForge home page [electronic resource] / URL: <http://www.voxforge.org/home/> / 30.10.2009. 6. Разработка инструментальной среды интеллектуального анализа аудиальных данных [текст] / А.А.Каргин, Т.В.Шарий // Труды VIII международной конференции «Интеллектуальный анализ информации ИАИ-2008», г.Киев. – 2008. – С. 558–564.

Поступила в редколлегию 26.10.2009

УДК 004.934.2

Про один метод автоматичної сегментації мовних сигналів / Т. В. Шарій // Біоніка інтелекту: наук.-техн. журнал. – 2009. – № 2 (71). – С. 61-65.

Стаття присвячена задачі автоматичної сегментації мовних сигналів. Досліджені акустичні параметри сигналу на межах фонем. Запропоновано новий метод «сліпої» сегментації STM+CSM, що ґрунтується на мірі спектрального переходу й мірі кепстральної гладкості. Експериментально підтверджено підвищення точності сегментації мовних сигналів з використанням нового методу.

Табл. 1. Іл. 4. Бібліогр.: 6 найм.

UDK 004.934.2

On one approach to automatic segmentation of speech signals / T.V. Shariy // Bionics of Intelligence: Sci. Mag. – 2009. – № 2 (71). – P. 61-65.

The paper is devoted to automatic speech segmentation task. The acoustic parameters at the phoneme boundaries are studied. The novel “blind” segmentation technique STM+CSM based on the Spectral Transition Measure and the Cepstral Smoothness Measure is developed. Experiments showed the increase of the segmentation accuracy of STM+CSM technique.

Tab. 1. Fig. 4. Ref.: 6 items.

УДК 004.81, 004.93



ІНТЕЛЕКТУАЛЬНА СИСТЕМА ПІДТРИМКИ ВЗАЄМОДІЇ З КОМП'ЮТЕРОМ ДЛЯ ЛЮДЕЙ З РУХОВИМИ ПОРУШЕННЯМИ

А.Л. Єрохін, О.В. Кольченко

Харківський національний університет внутрішніх справ,
м. Харків, Україна, ayerokhin@ukr.net
Харківський національний університет радіоелектроніки,
м. Харків, Україна, science.ukraine@gmail.com

Розглядається задача створення інтелектуальної системи взаємодії людей з обмеженими фізичними можливостями з комп'ютером, що забезпечує адаптацію користувачів з руховими порушеннями до управління комп'ютерними системами. Для розв'язання задачі розроблено модель та метод побудови магнітного інтерфейсу програмного додатку, що забезпечує суттєве скорочення вимушеної рухової активності для людей з руховими порушеннями.

АДАПТАЦІЯ КОРИСТУВАЧА, ІНТЕЛЕКТУАЛЬНА СИСТЕМА ВЗАЄМОДІЇ З КОМП'ЮТЕРОМ,
ЛЮДИНА З РУХОВИМИ ПОРУШЕННЯМИ, МАГНІТНИЙ ІНТЕРФЕЙС

1. Вступ та постановка задачі

На сучасному етапі розвитку систем та засобів штучного інтелекту стає можливою постановка задачі та розробка інтелектуальних методів, які можуть забезпечити адаптацію комп'ютера і людей з обмеженими фізичними можливостями

Велика кількість людей страждає хворобами, наслідком яких є порушення рухової активності. Рухові порушення істотно обмежують соціальну активність таких людей. Найчастіше одним із засобів комунікації з зовнішнім світом для таких людей є комп'ютер. Типовими засобами взаємодії з комп'ютером є клавіатура і маніпулятор «миша». Робота користувача з комп'ютерною мишею вимагає досить високого рівня контролю над рухами. Ось отут і виникає проблема використання комп'ютерної миші для людей з руховими порушеннями.

Ще однією категорією користувачів, що мають труднощі при роботі з комп'ютерною мишею, є люди похилого віку. Тремор рук істотно ускладнює позиціонування миші на елементах управління при роботі з комп'ютерними програмами, що негативно позначається на продуктивності і ступені задоволення від роботи.

Таким чином, актуальною є проблема розробки нових і удосконалення існуючих інтелектуальних методів та засобів взаємодії людини з комп'ютером для людей з руховими порушеннями.

У даний час існують два напрямки рішення значеної проблеми. Перший напрямок орієнтується на розробку нових апаратних засобів взаємодії людини з комп'ютером. До таких засобів відноситься спеціальний адаптер, що підключається між мишею і системним блоком комп'ютера. Пристрій *Assistive Mouse Adapter* розроблено компанією IBM у 2005 році [1]. Адаптер «відфільтровує» рухи людської руки, що трясеться, в результаті чого стабілізуються переміщення покажчика миші на екрані. Адаптер дозволяє налагоджувати рівень фільтрації.

Існує також можливість фільтрації випадкових натискань кнопок миші. Недоліками цього пристрою є його висока вартість і відсутність можливості автоматичної адаптації до особливостей кожного користувача.

Іншим напрямком, покликаним поліпшити якість взаємодії з комп'ютерами людей з обмеженими фізичними можливостями, є розробка методичних рекомендацій і стандартів для поліпшення доступності комп'ютерних інтерфейсів. Так, організацією *World Wide Web Consortium (W3C)* 11 грудня 2008 року опубліковано документ *Web Content Accessibility Guidelines (WCAG)* [2], що має статус стандарту для розробки веб-сторінок з урахуванням вимог доступності. У цьому документі є розділи, присвячені користувачам з руховими порушеннями. Незважаючи на наявність стандарту, під час розробки інтерфейсів додатків і веб-сторінок вимоги доступності часто просто відсутні у технічному завданні на розробку. Але навіть застосування зазначених стандартів, на жаль, не знімає проблеми взаємодії з комп'ютером користувачів з руховими порушеннями.

2. Вирішення задачі корекції мимовільних рухів користувача

Рухові порушення користувачів найчастіше виявляються у виді тремору. У роботі [3] приведена класифікація і характеристики різних видів тремору. Однією з характеристик тремору є частота. Амплітуда тремору визначається видом захворювання, його стадією і станом хворого.

Специфічною особливістю тремору, що виникає при рішенні задачі позиціонування покажчика на визначену користувачем мету на екрані, є посилення тремтіння у завершальній стадії руху [3].

Візуалізацію роботи з комп'ютерною мишею користувача без рухових порушень і користувача з руховими порушеннями представлено на рис. 1-4.

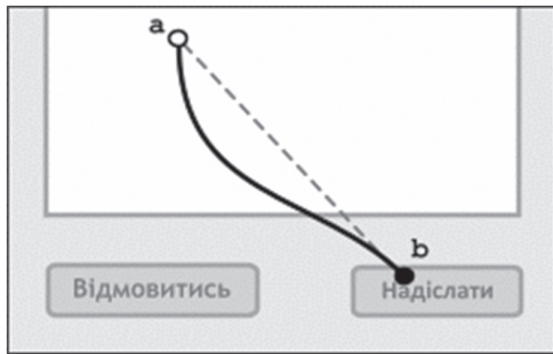


Рис. 1. Користувач без рухових порушень

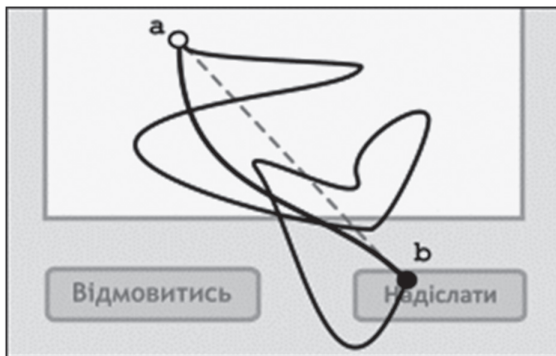
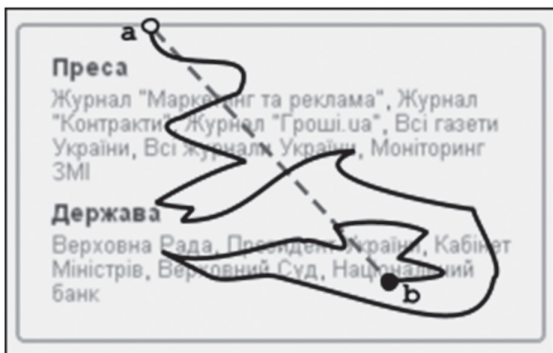


Рис. 2. Користувач з руховими порушеннями

Рис. 3. Користувач без рухових порушень
(каталог ukr.net)Рис. 4. Користувач з руховими порушеннями
(каталог ukr.net)

Аналіз роботи користувачів з покажчиком дозволяє зробити наступні висновки:

1) переміщення миші мають цільовий характер: покажчик переміщується з поточної позиції

до якого-небудь елемента управління (із точки *a* до точки *b*, рис. 1–4);

2) траєкторія переміщення миші користувачем без рухових порушень близька до прямої, яка з'єднує початкову і кінцеву точки шляху. Цим підтверджується відомий в інженерній психології принцип оптимальності рухів;

3) траєкторію переміщення миші користувачем з руховими порушеннями можна розглядати як результат накладання двох видів руху: цільового, для рішення задачі позиціонування покажчика в кінцеву точку, і випадкового, що є результатом прояву тремору.

Останній висновок нашою наукою на ідею про можливість застосування методів обробки часових рядів для корекції переміщення миші користувачами з руховими порушеннями. Це дозволить виділити тренд і відкинути випадкову складову в рухах таких користувачів. Крім того, аналіз випадкової складової дозволить сформулювати патерни для кожного виду тремору з метою створення профілів корекції рухів миші для різних видів рухових порушень.

Для користувачів з руховими порушеннями проблемними є не тільки задачі позиціонування покажчика, але і саме виконання великої кількості рухів. Для рішення задачі мінімізації рухів, пропонується увести функціональну залежність швидкості переміщення миші від відстані до цільового елемента управління. Таким чином, знаходячись на великій відстані до цільового елемента управління, користувач незначно переміщає покажчик, при цьому курсор переміщується на велику відстань. Наближаючи до цільового елемента управління, швидкість переміщення покажчика зменшується для виконання більш точного позиціонування. При цьому користувач виконує значно менший обсяг рухів, що запобігає швидкому настанню втоми. Виділення областей користувацького інтерфейсу, у межах яких швидкість переміщення миші сповільнюється для більш точного позиціонування, описано в [4].

Відмінність запропонованого в даній роботі способу полягає у тому, що:

а) метою є скорочення обсягу вимушеної рухової активності (рухів користувача з руховими порушеннями);

б) швидкість переміщення миші змінюється поступово, пропорційно відстані до цільового елемента управління.

Таким чином, для усунення ефекту тремору при роботі з мишею пропонується метод корекції переміщень покажчика, заснований на виділенні систематичної складової руху і економії рухів користувача.

Графічне представлення запропонованого методу подане на рис. 5.

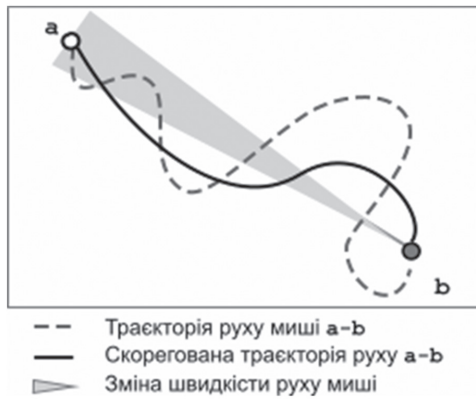


Рис. 5. Корекція рухів миші

3. Розробка моделі магнітного інтерфейсу

Взаємодія користувача з комп'ютерною програмою здійснюється у вигляді впливів на елементи управління, які саме і формують інтерфейс додатка. Таким чином, з погляду переміщень миші, кінцевими пунктами призначення покажчика є саме елементи управління.

Основна складність при роботі користувачів з руховими порушеннями полягає в нездатності побудови ефективної траєкторії переміщення і позиціонуванні покажчика на елементах управління додатка. Для вирішення зазначеної проблеми пропонується ідея «магнітного» інтерфейсу.

Для кращого розуміння принципу магнітного інтерфейсу розглянемо приклад з фізики. Нехай на деякій площині знаходяться декілька магнітів. У довільному місці на площині, не стикаючись з жодним із магнітів, знаходиться важка сталевая куля. Сила ваги не дає зрушити з місця кулю, незважаючи на дію на неї з боку магнітів сил магнітного притягання.

Однак, як тільки до кулі прикласти зовнішню силу, вона прийде в рух, причому на його траєкторію будуть впливати як вектор прикладеної зовнішньої сили, так і сили магнітного притягання, що діють з боку магнітів. Оскільки сила притягання магнітного поля зворотно пропорційна відстані, наближення кулі до одного з магнітів буде викликати усе більшу зміну його первісної прямої траєкторії руху, аж до «прилипання» до магніту.

У «магнітному» інтерфейсі роль кулі виконує миша, магнітів – елементи управління інтерфейсу, рис. 6. Ініціюючий зовнішній вплив на мишу виконує користувач.

Ідея «магнітного» інтерфейсу заснована на двох припущеннях:

- 1) кінцевою метою переміщення миші є який-небудь елемент управління;
- 2) намір користувача можна визначити за напрямком переміщення покажчика.

Між кожним елементом управління додатка і мишею встановлено зв'язок. Зв'язок характеризується силою, величина якої зворотно пропорційна відстані між елементом управління і покажчиком.

Найбільшу силу має зв'язок між мишею і найбільш близьким елементом управління. Зв'язок підсилюється при наближенні миші до елемента управління. Таким чином, «сильний» елемент управління начебто притягає мишу, при цьому пропорційно корегується траєкторія переміщення миші.

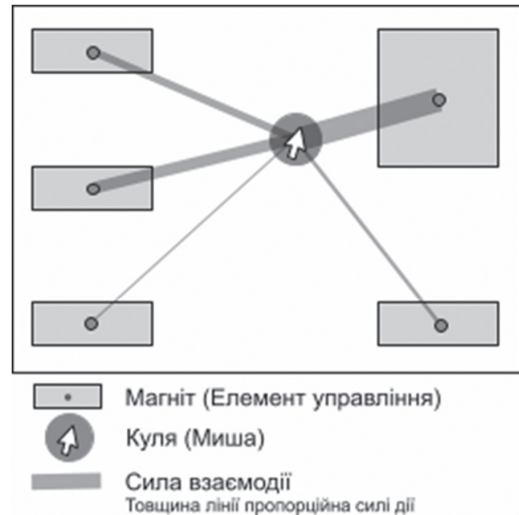


Рис. 6. «Магнітний» інтерфейс користувача

Новизна й інтелектуальність запропонованого підходу полягає в наступному:

- 1) елементи управління розглядаються як активні сутності, які здійснюють коригувальні впливи на траєкторію переміщення покажчика;
- 2) при виборі напрямку переміщення покажчика враховуються наміри користувача.

Формально «магнітний» інтерфейс зручно описувати, використовуючи теоретико-множинний підхід. Кожний елемент управління представляється точкою в двовимірному просторі з координатами x і y . Маніпулятор миша також представлений точкою з координатами на площині інтерфейсу. Для усіх елементів множини задана метрика: функція відстані. У найпростішому випадку – це Евклідова відстань. Таким чином, для зазначеної множини елементів управління і точкою, що відповідає покажчикові, можливе обчислення відстані:

$$r_{mc_i} = \sqrt{(x_m - x_{c_i})^2 + (y_m - y_{c_i})^2}, \quad (1)$$

де r_{mc_i} – відстань між покажчиком і елементом управління c_i ; x_m, y_m – відповідні координати покажчика; x_{c_i}, y_{c_i} – координати елемента управління c_i .

Крім відстані, для елементів множини необхідно визначити величину сили зв'язку між покажчиком і кожним елементом управління c_i . Силу взаємодії між кожним елементом управління і покажчиком пропонується розраховувати аналогічно силі взаємодії між полюсами постійного магніту:

$$\vec{F}_{mc_i} = k \frac{M_{c_i} m}{r_{mc_i}^n}, \quad (2)$$

де k – коефіцієнт пропорційності; M_{c_i} – деяка «магнітна» характеристика елемента управління. Наприклад, імовірність вибору елемента управління c_i , його важливість або інша властивість; $r_{mc_i}^n$ – відстань між покажчиком і елементом управління c_i , що задана відповідно до метрики; m – «магнітна» характеристика миші, $m \in [0..1]$; $i = 1..k$, k – кількість елементів управління поточної форми додатку.

Варіювання цієї характеристики дозволяє регулювати силу взаємодії елементів управління і миші. Так, при $m = 0$ «магнітний» інтерфейс відключається, оскільки $F_{mc_i} = 0$.

У формулі (2), що описує взаємодію полюсів магніта, $n = 2$; залежність сили впливу магнітного інтерфейсу від цього параметра представлена набором графіків для різних значень n на рис. 7;

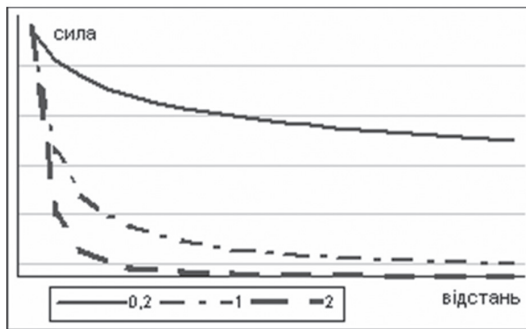


Рис. 7. Залежність сили впливу магнітного інтерфейсу від параметра n

Нехай \vec{F}_{user} – сила, прикладена користувачем до миші для ініціювання руху.

Тоді результуючий вектор сили, що впливає на покажчик у «магнітному» інтерфейсі, визначається за формулою:

$$\vec{F}_{mouse} = \vec{F}_{user} + \vec{F}_{mc_i}. \quad (3)$$

Приклади переміщення миші в «магнітному» інтерфейсі під дією сили \vec{F}_{mouse} показані на рис. 8.

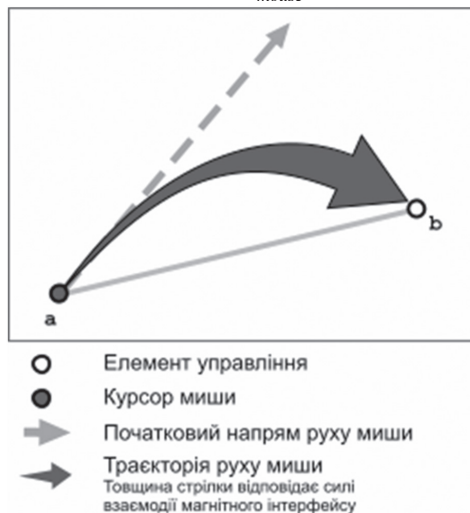


Рис. 8. Траєкторії покажчика в магнітному інтерфейсі

4. Висновки

Таким чином, у роботі запропоновано:

- метод корекції переміщень покажчика, заснований на виділенні систематичної складового руху і та економії рухів користувача з руховими порушеннями;

- модель і метод побудови спеціального «магнітного» інтерфейсу для користувачів з руховими порушеннями.

Подальші дослідження пов'язані з вивченням особливостей роботи користувачів з різними видами рухових порушень з метою формування відповідних патернів взаємодії. Додаткового вивчення вимагають параметри моделі магнітного інтерфейсу.

Також передбачається розробка алгоритмів для запропонованих методів та реалізація відповідного програмного забезпечення.

Список літератури: 1. *Mouse adapter gives computer access to millions of hand tremor sufferers* [Електронний ресурс] Режим доступу : [www/ URL : http://domino.research.ibm.com/comm/pr.nsf/pages/news.20050314_mouseadapter.html](http://domino.research.ibm.com/comm/pr.nsf/pages/news.20050314_mouseadapter.html) – 14.02.2009 р. – Загл. з екрану. 2. *Web Content Accessibility Guidelines (WCAG) 2.0* [Електронний ресурс] Режим доступу : [www/ URL : http://www.w3.org/tr/wcag20/](http://www.w3.org/tr/wcag20/) – 14.02.2009 р – Загл. з екрану. 3. *Peter G Bain. The Management of Tremor* [Електронний ресурс] / *Journal of Neurology, Neurosurgery, and Psychiatry.* – Режим доступу : [www/ URL : http://jnnp.bmj.com/cgi/content/full/72/suppl_1/i3](http://jnnp.bmj.com/cgi/content/full/72/suppl_1/i3) – 14.02.2009 р – Загл. з екрану. 4. *Automatically adjusting a speed of a graphical pointer.* Patent [Електронний ресурс] Режим доступу : [www/ URL : http://www.wipo.int/pctdb/en/wo.jsp?IA=wo2004086212&display](http://www.wipo.int/pctdb/en/wo.jsp?IA=wo2004086212&display) – 14.02.2009 р – Загл. з екрану.

Надійшла до редакції 23.10.2009

УДК 004.81, 004.93

Интеллектуальная система поддержки взаимодействия с компьютером для людей с двигательными нарушениями / А.Л.Ерохин, А.В.Кольченко // Бионика интеллекта: науч.-техн. журнал. – 2009. – № 2 (71). – С. 66-69.

Взаимодействие пользователей с двигательными нарушениями с компьютером затруднено. Для облегчения работы пользователей с мышью предлагается метод коррекции произвольных движений. При этом решаются задачи позиционирования указателя и экономии движений пользователя. Также описана оригинальная концепция магнитного графического пользовательского интерфейса. Магнитный интерфейс корректирует траекторию перемещения мыши с учетом намерений пользователя.

Рис. 8. Библиогр.: 4 назв.

UDC 004.81, 004.93

Intelligence system for interaction of users with motor disorders with a computer / A.L.Yerokhin, O.V.Kolchenko, // *Bionics of Intelligence: Sci. Mag.* – 2009. – № 2 (71). – P. 66-69.

The interaction of users with motor disorders with a computer is difficult. To facilitate mouse related tasks the method of correction of involuntary movements is proposed. The method solves the problem of positioning the mouse pointer and minimizes the amount of user movements. The original concept of the magnetic graphical user interface is considered. Magnetic interface adjusts the trajectory of the mouse movements, taking into account the intentions of the user.

Fig. 8. Ref.: 4 items.

УДК 519.686.4



ИССЛЕДОВАНИЕ И ПОСТРОЕНИЕ ИГРОВОЙ МОДЕЛИ ОБУЧЕНИЯ НА ОСНОВЕ ТЕХНОЛОГИЙ СОЦИАЛЬНЫХ СЕТЕЙ

Е.Д. Емцев¹, Н.С. Лесная², И.Н. Келеберда³

¹ХНУРЕ, г. Харьков, Украина, doctor_e@bk.ru

²ХНУРЕ, г. Харьков, Украина

³ХНУРЕ, г. Харьков, Украина, i.keleberda@dmil.com

Проведен анализ технологий социальных сетей и игровой модели обучения. Разработанная игровая модель обучения на основе имитационных игр была организована на основе технологий социальной сети Ning.com.

СОЦИАЛЬНЫЕ СЕТИ, ИГРОВАЯ МОДЕЛЬ, ИГРОВЫЕ ТЕХНОЛОГИИ ОБУЧЕНИЯ, ИМИТАЦИОННАЯ ИГРА

Введение

В настоящее время социальные сети активно используются в качестве средства взаимодействия в глобальной сети Интернет. В связи с этим социальные сети являются перспективной платформой для организации процесса обучения не только в учебных заведениях, но и в сети Интернет. Для использования социальных сетей в данных целях необходимо исследовать новые формы дистанционного обучения.

Использование социальных сетей как базы для построения процесса обучения позволяет решить такие проблемы, как связь между учеником и учителем, объединение учащихся в группы по интересам для их взаимодействия и обмена информацией между собой, поиска и консультации у других экспертов. С помощью социальной сети возможно также организовать: семинары, лекции, самостоятельное обучение или обучение с учителем и так далее. Использование социальных сетей дает огромные возможности для перехода процесса обучения на новый уровень.

Доклады об использовании социальных сетей в дистанционном образовании были представлены на 50-ой научной конференции МФТИ [1] и на 51-ой научной конференции МФТИ на ФИВТ [2], на которых рассматривались возможности использования социальных сетей в образовании.

Объектом исследования являются технологии социальных сетей с целью организации процесса обучения. Предметом исследования является игровая модель обучения и ее внедрение в социальную сеть.

Целью данной работы является организация процесса обучения на базе игровой модели, что обеспечит заинтересованность участников данного процесса.

1. Анализ технологий социальных сетей

Социальная сеть — это интерактивный многопользовательский веб-сайт, контент которого наполняется самими участниками сети.

На самом примитивном уровне понимания социальная сеть — это определенное количество людей, которых вы знаете и с которыми поддерживаете связь. Если брать как пример школу, то в вашу социальную сеть входили бы все ваши одноклассники. На работе в нее попадают все, с кем вам приходится общаться на работе.

Одна из причин, по которой социальные сети являются успешными — это то, что они предоставляют пользователям шанс открыться и активно участвовать в работе сообщества. Любая социальная сеть (форум, сообщество вокруг блога, сообщество наподобие MySpace или Orkut) будет успешной только в том случае, если ее члены имеют достаточный запас энтузиазма для того, чтобы принимать активное участие в общении и обмене идеями.

К современным средствам общения в сети Интернет относят: гостевые книги, форумы, блоги, чаты и электронную почту. Именно в связи с их развитием и стали образовываться социальные сети, — то есть совокупности участников, объединенных не только средой общения, но и с явно установленными связями между собой. В целом все современные системы общения для сетевых сообществ обладают несколькими общими чертами:

- необходимость регистрации пользователя для получения доступа к контенту сайта;
- сеансовый тип работы;
- необходимость настройки пользовательского окружения (внешний вид, дополнительные данные о себе, свои интересы и так далее).

Социальные сети и поддерживающие их сервисы оказались очень эффективным методом обеспечения посещаемости сайтов, обратной связи и постепенно стали одним из средств генерации контента (содержимого, имеющего ценность). На основе такого подхода появилось и быстро набрало популярности довольно большое количество социальных web-сервисов, объединенных общим названием сервисы Web 2.0.

2. Исследование игры в процессе обучения

Игра, являясь простым и близким человеку способом познания окружающей действительности, является наиболее естественным и доступным путем к овладению теми или иными знаниями, умениями, навыками. В структуру игры в качестве деятельности входит целеполагание, планирование, реализация цели, а также анализ результатов, в которых участник должен полностью себя реализовать. Мотивация игровой деятельности обеспечивается ее добровольностью, возможностями выбора и элементами соревновательности, удовлетворения потребности в самоутверждении, самореализации[3].

В основе игры лежит способность оперировать образами действительности, развитие данной способности ведет к дальнейшему переходу к более сложным формам творческой деятельности.

Большое влияние оказывает игра и на развитие коммуникационных способностей учащихся во время обучения. Кроме того, что учащийся участвует в игре, осваивает правила, способы взаимодействия в совместной игре с другими участниками, он приобретает опыт взаимопонимания, учится пояснять свои действия и намерения, согласовывать их с другими участниками.

Игра способствует раскрепощению личности, развитию коммуникативных способностей, преодолению боязни общения с незнакомыми людьми, развитию культуры общения. В процессе игры формируется способность принятия самостоятельных решений, способность оценивать свои действия и действия других.

Влияние игры проявляется в воспитании чувства коллективизма и товарищества, а также чувства высокой личной ответственности перед коллективом (командой) за результаты игры.

3. Игровые технологии обучения

Любая технология обладает средствами, активизирующими и интенсифицирующими деятельность учащихся. К таким технологиям можно отнести игровые технологии, проблемное обучение, коммуникативные технологии.

В настоящее время в процессе обучения наиболее широко используются такие виды игр как деловые, имитационные и операционные.

Деловые игры отличаются от других, во-первых, имитацией деятельности реальных социально-экономических систем, во-вторых, участники игры выступают в тех или иных ролях лишь для приобретения опыта преодоления конфликтов и принятия деловых решений; в-третьих, деловая игра всегда метод коллективного обучения; в-четвертых, в деловых играх создается эмоциональный настрой для активизации процесса обучения [4].

Деловая игра используется для решения комплексных задач усвоения нового, закрепления материала, развития творческих способностей, формирования общеучебных умений, дает возможность обучаемым понять и изучить учебный материал с различных позиций.

Суть имитационной игры – на занятиях имитируется деятельность предприятия или его подразделения: команды разработчиков, отдела тестирования и так далее. Имитироваться могут события, конкретная деятельность людей (деловое совещание, обсуждение плана, проведение беседы и так далее), обстановка и условия, в которых происходит событие или осуществляется деятельность (кабинет, зал заседаний). Сценарий имитационной игры, кроме сюжета события, содержит описание структуры и назначения имитирующих процессов и объектов.

Операционные игры помогают отработать выполнение конкретных специфических операций работы с клиентами, обработку заявлений, прием документов и так далее. В операционных играх моделируется соответствующий рабочий процесс. Игры этого типа проводятся в условиях, имитирующих реальные.

4. Алгоритм обучения информационным технологиям на основе имитационной игры

В основу разрабатываемой игровой модели обучения легли имитационные игры. В качестве примера для построения игровой модели обучения используем процесс обучения тестированию программного продукта.

Разработанная модель состоит из следующих этапов:

- подготовительный этап – на данном этапе участники проходят процесс регистрации в обучающей программе, разбиения на учебные группы (команды) и распределения ролей рамках команды;

- этап получения и выполнения заданий – на данном этапе каждая команда имитирует процесс работы команды специалистов по тестированию, они выполняют ряд поставленных перед ним задач по тестированию, в конце данного этапа каждая команда по результатам тестирования оформляет документацию;

- этап защиты выполненных заданий – на данном этапе каждая команда имитирует процесс сдачи программного продукта после этапа тестирования, она должна показать, что тестирование было проведено в должном объеме и покрывает весь программный код, свидетельством этому является документация по результатам тестирования;

- этап приема выполненных заданий у другой группы – на данном этапе команда имитирует процесс приема программного продукта от команды

специалистов по тестированию, на данном этапе команда должна убедиться, что процесс тестирования был проведен должным образом;

– этап участия в учебном квесте – на данном этапе все команды участвуют в квесте, который проверяет уровень знаний учащихся в области тестирования. На данном этапе проверяется не только уровень теоретических знаний, но и развивается умение проводить быстрое тестирование;

– этап выхода – это завершающий этап, на котором происходит подведение итогов и объявление результатов учебной деятельности каждой команды.

Последовательность этапов игровой модели обучения представлена на рис. 1.

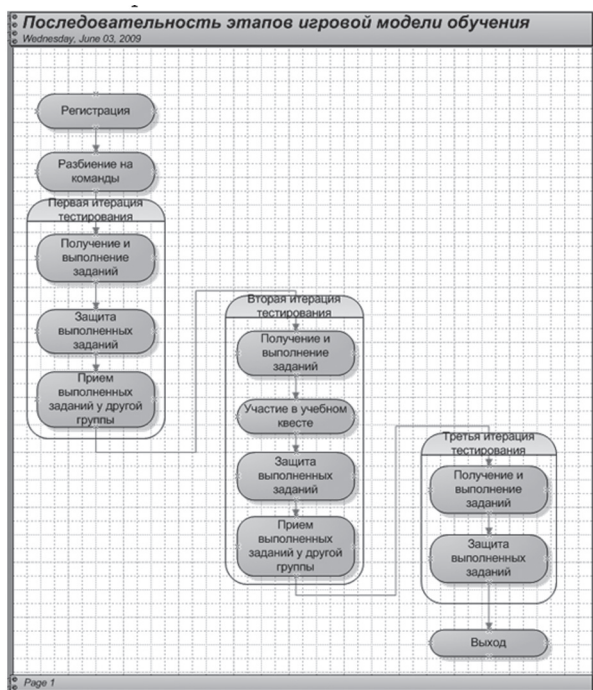


Рис. 1. Последовательность этапов игровой модели обучения

Как видно из рисунка, в процессе обучения происходят три итерации тестирования, на каждой итерации объем тестирования увеличивается. Это происходит за счет расширения проводимых видов тестирования на каждой итерации.

В результате прохождения всего процесса обучения каждая команда должна провести полное тестирование данного им программного продукта.

5. Построение игровой модели обучения на основе технологий социальных сетей

В качестве web-сервиса для организации обучения был выбран сервис www.ning.com. Данный сервис дает возможность создать и сконфигурировать свою собственную социальную сеть; имеет широкий набор функций для организации общения участников между собой и с руководителями. При рассмотрении данного сервиса было определено, что он имеет необходимый набор функций

для организации всех этапов обучения игровой модели.

На рис. 2 представлена диаграмма вариантов использования (Use Case Diagram) подготовительного этапа игровой модели. Данная диаграмма показывает, как организован подготовительный этап игровой модели в обучающей социальной сети «E-learning».



Рис. 2. Диаграмма вариантов использования подготовительного этапа

На подготовительном этапе участник проходит процесс регистрации, после чего ему приходит приглашение на вступление в учебную группу от его руководителя. За создание учебной группы и приглашение туда участника отвечает руководитель. После того как участник зарегистрировался в учебной группе, он может начать обучение.

Первая итерация тестирования, также и вторая, и третья имеют практически идентичную структуру. Каждая из них состоит из следующих этапов:

- этап получения и выполнения заданий;
- этап приема выполненных заданий у другой группы;
- этап защиты выполненных заданий;
- этап участия в учебном квесте.

Диаграмма вариантов использования (Use Case Diagram) для этапа получения и выполнения заданий показывает (рис. 3), что могут использовать на данном этапе участники для организации взаимодействия с целью выполнения поставленных перед ними задач.

В начале этапа получения и выполнения заданий руководитель в каждой учебной группе добавляет список заданий для данного этапа обучения, после чего и начинается выполнение этих заданий каждой командой. Для выполнения поставленных перед ними задач им необходимо организовать процесс общения между собой и, если необходимо, с их руководителем. Для этого можно использовать: форум, блоги, заметки, чат, доску комментариев. Для обмена информацией о том или ином виде тестирования есть возможность добавлять видео презентации или видео лекции.

На этапе приема выполненных заданий у другой группы и на этапе защиты выполненных заданий используются одни и те же функции, отличие

только в выполняемых каждой командой ролях. Вследствие этого для указанных двух этапов присуща одна диаграмма использования (Use Case Diagram), которая представлена на рис. 4.

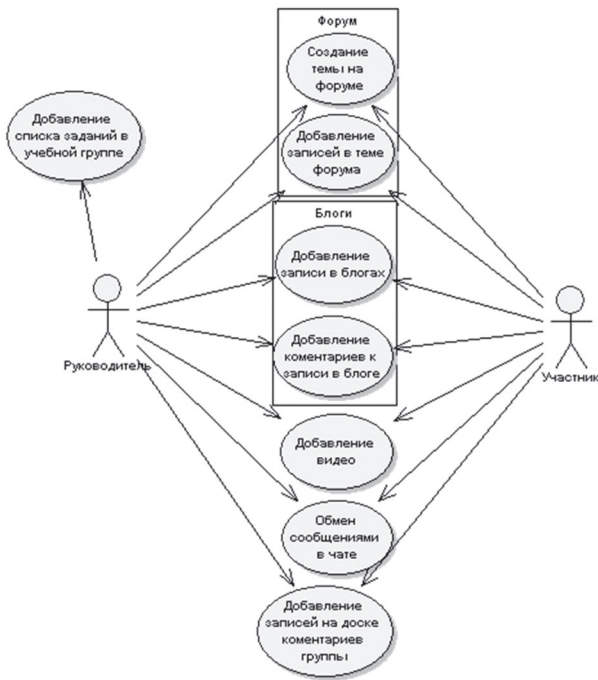


Рис. 3. Диаграмма вариантов использования этапа получения и выполнения заданий

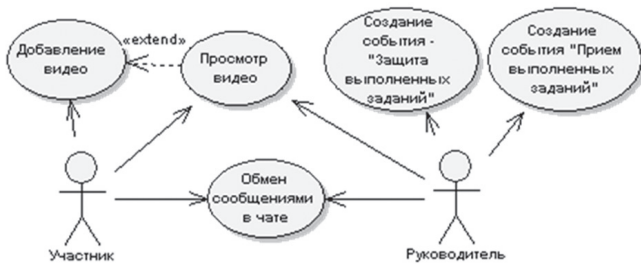


Рис. 4. Диаграмма вариантов использования для этапа приема выполненных заданий у другой группы и этапа защиты выполненных заданий

Как видно из диаграммы, руководитель создает для одной команды событие «Защита выполненных заданий», а для другой команды событие «Прием выполненных заданий». Затем перед наступлением события «Защита выполненных заданий» команда, которая защищает результаты своей работы, загружает видео, в котором ее участники рассказывают, как они выполняли поставленные перед ними задачи и какие у них получились результаты. События «Защита выполненных заданий» и «Прием выполненных заданий» назначены на одно и то же время. В назначенное время обе команды должны присутствовать в полном составе. При просмотре видео презентации могут возникнуть некоторые вопросы как и у руководителя, так и у участников другой группы. Для обсуждения всех появившихся вопросов используется чат социаль-

ной сети, который позволяет организовать живой диалог между участниками данных событий.

Этап участия в учебном квесте заключается в прохождении каждой командой квеста. Диаграмма использования данного этапа представлена на рис. 5.

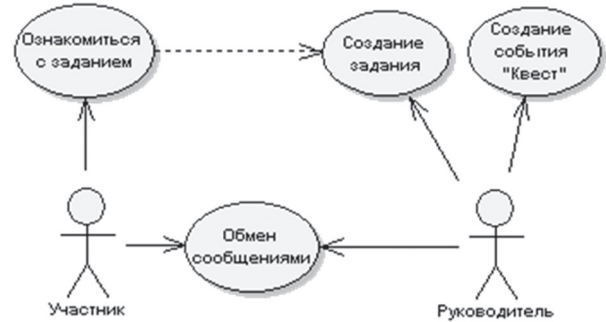


Рис. 5. Диаграмма вариантов использования этапа участия в учебном квесте

Как видно из этой диаграммы, на данном этапе руководитель создает событие «Квест» для всех участников. Затем при наступлении этого события руководитель создает страницу с первыми заданиями, с которыми знакомятся все участники и которые они должны выполнить. При выполнении первого задания квеста участник отправляет результат (ответ) сообщением руководителю, на что получает ответ от руководителя о том, был ли ответ правильным или нет. Если ответ правильный, руководитель создает следующую страницу с заданием, и так этот процесс продолжается дальше до тех пор, пока участники не закончат отвечать на все вопросы квеста.

Диаграмма вариантов использования для этапа выхода представлена на рис. 6.

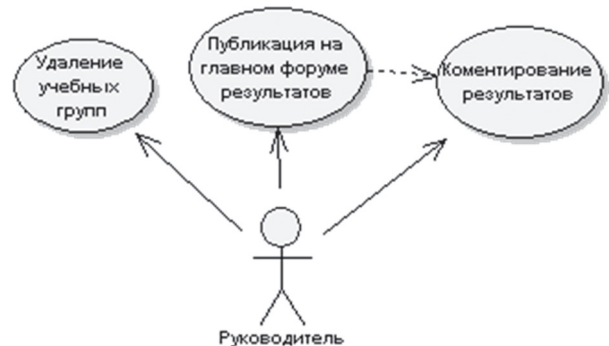


Рис. 6. Диаграмма использования этапа выхода

После прохождения всех этапов обучения участники переходят на этап выхода из игры, на котором руководитель удаляет учебные группы и публикует финальные результаты на главном форуме и комментирует результат каждой команды.

Выводы

В рамках статьи были проанализированы технологии социальных сетей, рассмотрена игровая модель и игровые технологии обучения. На базе

данной модели был предложен алгоритм обучения информационным технологиям, который состоит из подготовительного этапа, этапа получения и выполнения заданий, этапа защиты выполненных заданий и этапа приема выполненных заданий другой группы. Выбранный имитационный подход для игровой модели обучения был применен в области изучения тестирования программных продуктов. Также рассмотрен процесс построения игровой модели обучения на основе технологий социальных сетей с помощью диаграмм вариантов использования (Use case diagram).

Список литературы: 1. Пустовой, Т.В. Дистанционное образование и социальные сети. Синергетический подход в построение систем удаленного образования федерального уровня [Текст] / Т.В. Пустовой: тез. докл. науч.-техн. конф. МФТИ «Современные проблемы фундаментальных и прикладных наук». – 2007. – Т. 10. 2. Пустовой, Т.В. Разработка архитектуры гетерогенного образовательного пространства [Текст] / Т.В. Пустовой: тез. докл. науч.-техн. конф. МФТИ. 3. Пидкасистый, П.И. Технология игры в обучении и развитии [Текст] / П.И. Пидкасистый, Ж.С. Хайдаров: учеб. пособие. – М.: МПУ, 1996. – 132 с. 4. Пидкасистый, П.И. Технология игры в обучении

[Текст] / П.И. Пидкасистый: учеб. пособие. – М.: Просвещение, 1992. – 65с.

Поступила в редакцию 26.10.2009

УДК 519.686.4

Дослідження й побудова ігрової моделі навчання на основі технологій соціальних мереж / Е.Д. Емцев, Н.С. Лесна, І.М. Келеберда // Біоніка інтелекту: наук.-техн. журнал. – 2009. – № 2 (71). – С. 70-74.

У статті розглядається один з перспективних напрямків розвитку технологій навчальних соціальних мереж. Організація процесу навчання на основі ігрової моделі в рамках навчальних соціальних мереж може бути гарним прикладом для використання ігрової моделі в дистанційному навчанні.

Іл. 6. Бібліогр.: 4 найм.

UDK 519.686.4

Investigation and building game model process on base of social network technology / E.D. Emtsev, N.S. Lesnaya, I.N. Keleberda // Bionics of Intelligence: Sci. Mag. – 2009. – № 2 (71). – P. 70-74.

In article consider one of the promising directions of progressing study social network technology. Organization study process on the base of game model in scope of study social network could be a good example for the using game model in e-learning.

Fig. 6. Ref.: 5 items.

УДК 004.048



РЕПОЗИТОРИЙ ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА ДАННЫХ

Т.Б. Шатовская¹, В.Б. Репка², И.В. Каменева³, М.М. Марченко⁴

1ХНУРЭ, г. Харьков, Украина, shatovska@gmail.com,

2ХНУРЭ, г. Харьков, Украина, vic-repka@yandex.ru,

3ХНУРЭ, г. Харьков, Украина, irina.kamenieva@gmail.com,

4ХНУРЭ, г. Харьков, Украина, shami13@gmail.com;

В данной статье представлен анализ известных репозиторий статистических наборов данных, а также на основе проведенного анализа была разработана модель интеллектуального научного репозитория статистических наборов данных.

РЕПОЗИТОРИЙ, ИНТЕЛЛЕКТУАЛЬНЫЕ АГЕНТЫ, JADEX, ВЕБ-СЕРВИС, ORACLE, RDF, FTP

Вступление

Для проведения исследований в области интеллектуального анализа данных и машинного обучения необходимым условием является наличие различных входных наборов данных. В настоящее время исследователи создают свои базы данных таких наборов. Примером такой системы может служить: The UCI Machine Learning Repository [1], который является наиболее популярным среди исследователей благодаря своей классификации наборов данных, а также содержанием множества наиболее часто встречающихся выборок, Data Envelopment Analysis Dataset Repository [2], XMLData Repository [3], Frequent Itemset Mining Dataset Repository [4]. Наряду с вышеуказанными статистическими репозиториями целая плеяда от простых хранилищ файлов до специализированных хранилищ данных может быть использована исследователями при решении прикладных задач, проведении исследований собственных алгоритмов и научных проблем. Казалось бы, единственной сложностью для пользователя будет являться поиск и непосредственное понимание структуры столь разрозненных хранилищ информации. Однако детальное исследование таких хранилищ данных приводит нас к осознанию наличия более глубоких проблем в использовании данных. В частности наблюдается полное несогласование и жесткость структуры файлов с данными с SDMX – Statistical Data and Metadata Exchange – стандартом и структурой, используемой многими европейскими организациями, невозможность предварительной подготовки данных к конкретной прикладной задаче, отсутствие истории использования данных для тех или иных научных и прикладных задач.

1. Постановка задачи

Целью данной работы является концептуально новый подход к созданию интеллектуального хранилища статистических данных, основанного на взаимодействии онтологических моделей методов Data mining, пользователей системы и агентном подходе хранения и обработки файлов репозитория данных.

2. Анализ статистических репозиторий

The UCI Machine Learning Repository. Все данные о наборах данных хранятся в виде файлов как ftp-хранилище. Описание каждого набора данных также хранится в виде файла, что замедляет скорость поиска информации.

Преимущества репозитория: хорошо отсортированные данные, полнотекстовый поиск.

Недостатки репозитория: только текстовый формат данных, неудобен в использовании и для преобразований. Отсутствует система персонализации, пользователь может добавить выборку только по жесткому шаблону, что требует дополнительных затрат на администрирование системы.

Data Envelopment Analysis Dataset Repository (DEA Dataset Repository) позволяет пользователям загружать и искать наборы данных.

Преимущества репозитория: поиск по любому из критериев.

Недостатки репозитория: формат файлов, интерфейс. Поиск осуществляется только по одному из множества критериев, то есть нельзя совместить поиск по нескольким условиям. Неопытным пользователям сложно осуществлять поиск по данному репозиторию, нет просмотра всех наборов данных репозитория.

XML Data Repository. В XML Data Repository собраны данные только в виде XML, а также статистические данные для использования в научно-исследовательских экспериментах.

Недостатки данной системы: неудобство поиска, нет понимания, для каких задач можно использовать данную выборку, недостаточный объем информации по данным. Нет дополнительной информации о наборах данных, некоторые наборы данных очень большие по объему.

Преимущества: универсальный формат данных XML, прост для преобразования в любой другой формат и для программного использования. Отсутствие регистрации и получение данных по протоколу http позволяет использовать репозиторий агентами для поиска данных.

Frequent Itemset Mining Dataset Repository содержит небольшое количество наборов данных, которые не содержат классификации.

Недостатки данной системы: неудобство поиска, нет отображения дополнительной информации о наборах данных, некоторые наборы данных недоступны для скачивания.

Преимущества: для каждого набора данных существует описание экспериментального использования.

В конечном итоге были выявлены следующие недостатки существующих репозиториях: наличие только текстового формата данных, отсутствие систем персонализации, неудобство пользовательского интерфейса, неудобство поиска, недостаточный объем информации о данных.

3. Репозиторий интеллектуального анализа научных наборов данных

Система репозитория интеллектуального анализа научных наборов данных предназначена для использования исследователями, которые проводят эксперименты в области интеллектуального анализа данных и машинного обучения. Основной целью данной системы является хранение наборов данных, их автоматическая классификация, поиск, предобработка, персонализация работы с системой. Система представляет собой веб-приложение, которое использует агентную платформу Jadex[5], [6]. Общая схема работы системы изображена на рис. 1.

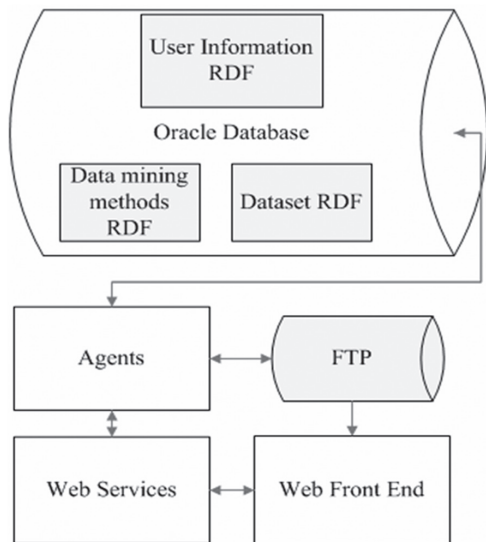


Рис. 1. Общая схема работы системы

Система состоит из следующих блоков: сервер баз данных, FTP сервер, веб-сервисы, агентная платформа Jadex, веб-интерфейс.

Сервер баз данных Oracle используется для хранения информации о пользователях системы, о методах исследований и хранит информацию о мета данных для наборов данных [7], [8].

FTP сервер используется для хранения файлов наборов данных.

Агентная платформа Jadex используется для проведения бизнес-логики системы [9].

Для реализации поиска в репозитории был разработан поисковый агент, основанный на цели, который может учитывать текущее состояние системы и различные критерии поиска, для того чтобы принять ту или иную стратегию поиска. При этом интеллектуальный агент действует не просто рефлексивно, но и принимает решение: какие именно действия необходимо выполнить, для того чтобы достичь поставленной цели в условиях текущего состояния среды.

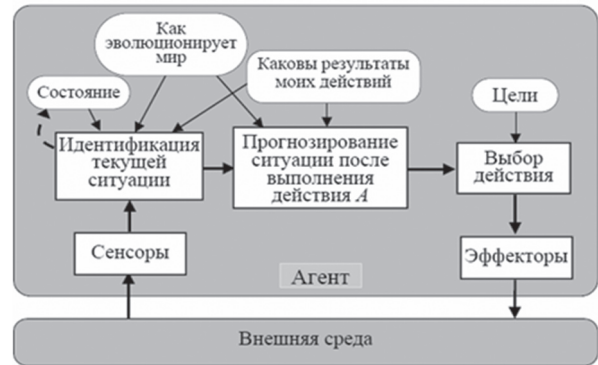


Рис. 2. Архитектура агента основанного на целях

Для реализации поискового агента в Jadex используются такие понятия как знания (beliefs), факты (facts), планы (plans), события (events), а также цели (goals). На основании имеющихся знаний и событий окружающей среды поисковый агент выбирает тот или иной план для достижения поставленной цели – поиска наиболее релевантных данных по запросу [9], [10].

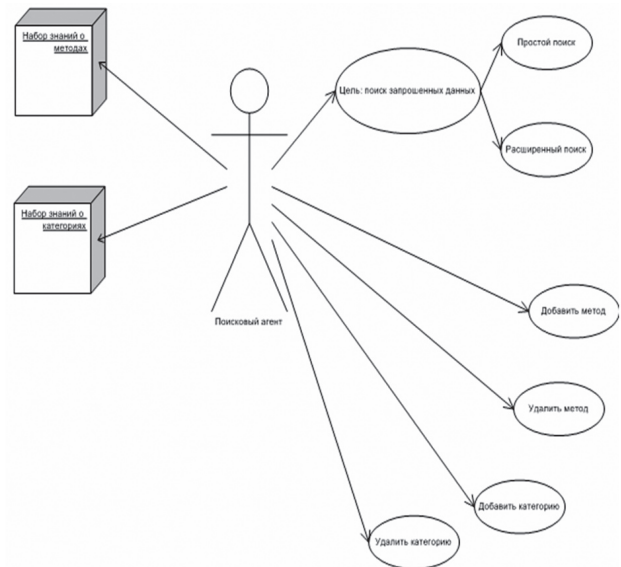


Рис. 3. Структурная схема поискового агента

Структурно-поисковый агент состоит из двух основных планов (SimpleSearchPlan и ExtendedSearchPlan) и нескольких вспомогательных (AddMethodPlan, RemoveMethodPlan, AddCategoryPlan, RemoveCategoryPlan), также двух наборов знаний.

В поисковом агенте все планы имеют пассивную стратегию работы. То есть это подразумевает, что для обработки каждого сообщения создается новый объект плана, и после обработки сообщения этот объект удаляется. Такая стратегия позволяет обрабатывать несколько сообщений одновременно.

В качестве входных данных для агента применяются сообщения с различным содержанием: строка с запросом поиска; набор пар: ключ и значение, а также события добавления или удаления сведений об имеющихся в хранилище методах и категориях. Результатом работы агента является набор ссылок на найденные данные.

Главной целью агента является поиск методов в хранилище. Для достижения этой цели агент использует SimpleSearchPlan и ExtendedSearchPlan планы. Для SimpleSearchPlan плана входными данными является строка, содержащая запрос поиска. При этом в строке запроса могут использоваться булевы выражения, например, “и”, “или”, “нет” и их комбинации. А ExtendedSearchPlan план используется для расширенного поиска. Входными данными является набор, содержащий пары: ключ и значение. На основании полученных данных поисковый агент строит запрос к репозиторию данных [11], [12].

Для того чтобы интегрировать Jadex платформу с веб, был разработан внешний интерфейс – Jadex Web Service eXtension (JWSX). Суть этого интерфейса заключается в создании внешнего доступа к функциям агента через веб сервисы [13], [14].

Веб-сервисы используются для взаимодействия пользователя веб-интерфейса с агентной платформой.

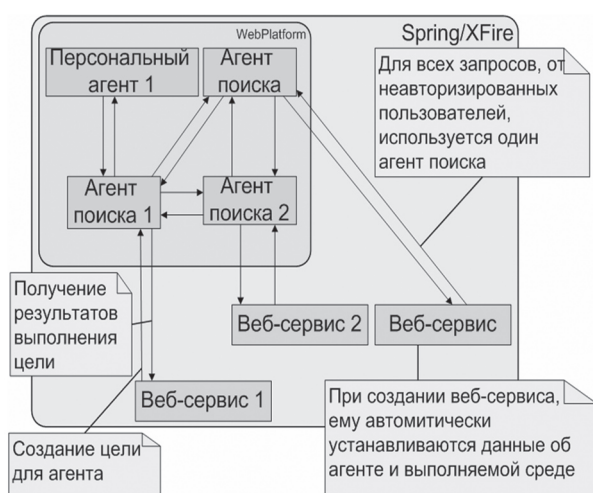


Рис. 4. Интеграция агента поиска с веб-сервисами

Варианты использования данной системы изображены на рис. 5.

В свою очередь каждый из представленных вариантов использования можно уточнить, используя несколько вариантов использования.



Рис. 5. Варианты использования системы

Вариант использования «Регистрация» включает в себя следующие варианты использования:

- регистрация начинающего пользователя;
- регистрация продвинутого пользователя.

Вариант использования «Просмотр наборов данных» включает в себя следующие варианты использования:

- просмотр всех наборов данных, находящихся в системе;
- просмотр конкретного набора данных в расширенном виде: все метаданные, комментарии, оценки.

Вариант использования «Фильтрация наборов данных» включает в себя следующие варианты использования:

- фильтрация наборов данных по методу исследования;
- фильтрация наборов данных по параметрам метаданных.

Вариант использования «Менеджмент комментариев» включает в себя следующие варианты использования:

- редактирование комментария;
- удаление комментария.

Вариант использования «Менеджмент пользователей» включает в себя следующие варианты использования:

- редактирование данных о пользователе;
- удаление пользователей;
- изменение статуса пользователя;
- назначение пользователя администратором.

Вариант использования «Менеджмент наборов данных» включает в себя следующие варианты использования:

- редактирование метаданных о наборе данных;
- редактирование местоположения файла набора данных;
- удаление набора данных;
- добавление методов исследований для использования набором данных.

Вариант использования «Менеджмент методов исследований» включает в себя следующие варианты использования:

- добавление метода исследования;
- редактирование данных о методе исследования;
- удаление метода исследования;
- изменение в иерархической структуре методов исследования.

Все варианты использования системы распределяются между агентами.

Данный репозиторий будет иметь следующие функциональные возможности:

- а) регистрация пользователей;
- б) аутентификация пользователей;
- в) менеджмент научных наборов данных: добавление новых выборок в систему, редактирование выборок, удаление выборок;
- г) менеджмент методов исследования: добавление новых методов, редактирование методов, удаление методов;
- д) поиск по онтологии наборов данных по таким критериям: тип задачи; предметная область; имя набора данных; ключевые слова; вид (шкала) входной/выходной характеристики; количество элементов (объем); дата (помещения в репозиторий, модификации, создания); автор задачи/набора данных/экспериментов; методы, какими решалась задача; эффективность работы набора данных.
- е) добавление результатов работы с выборками;
- ж) определение степени валидности набора данных.

Выводы

Проведен сравнительный анализ современных поисковых алгоритмов в репозиториях научных наборов данных и поисковых сервисов, выявлены их преимущества и недостатки; разработан поисковый агент, исследована и реализована возможность интеграции интеллектуальных агентов и веб-сервисов.

Данная система позволит исследователям иметь общее рабочее пространство для подготовки к экспериментам с алгоритмами интеллектуального анализа данных и машинного обучения, используя основные функции системы, такие как: загрузка наборов данных, поиск наборов данных, поиск исследователей, интересующихся теми же методами анализа данных.

В дальнейшем данную систему можно улучшить развитием входящих в нее онтологий и увеличением числа агентов. Также можно добавить подсистему работы со статьями и результатами научных экспериментов исследователей, которые проводились с использованием наборов данных из репозитория.

- Список литературы:** 1. The UCI Machine Learning Repository [Электронный ресурс]. – Режим доступа: [www/ URL: http://archive.ics.uci.edu/ml/](http://archive.ics.uci.edu/ml/) – 09.09.2009 г. – Назв. с экрана. 2. Data Environment Analysis Dataset Repository [Электронный ресурс]. – Режим доступа: [www/ URL: http://www.etm.pdx.edu/DEA/Dataset/default.htm/](http://www.etm.pdx.edu/DEA/Dataset/default.htm/) – 09.09.2009 г. – Назв. з экрана. 3. XMLData Repository [Электронный ресурс]. – Режим доступа: [www /URL: http://www.cs.washington.edu/research/xmldatasets/](http://www.cs.washington.edu/research/xmldatasets/) – 09.09.2009 г. – Назв. с экрана. 4. Frequent Itemset Mining Dataset Repository [Электронный ресурс]. – Режим доступа: [www/URL: http://fimi.cs.helsinki.fi/data/](http://fimi.cs.helsinki.fi/data/) – 09.09.2009 г. – Назв. с экрана. 5. Jadex BDI Agent System [Электронный ресурс]. – Режим доступа: [www/URL: http://jadex.informatik.uni-hamburg.de/bin/view/About/Overview/](http://jadex.informatik.uni-hamburg.de/bin/view/About/Overview/) – 09.09.2009 г. – Назв. с экрана. 6. Jadex User Guide [Электронный ресурс]. – Режим доступа: [www /URL: http://jadex.informatik.uni-hamburg.de/docs/jadex-0.96x/userguide/index.single.html/](http://jadex.informatik.uni-hamburg.de/docs/jadex-0.96x/userguide/index.single.html/) – 09.09.2009 г. – Назв. с экрана. 7. Oracle Database 10g [Электронный ресурс]. – Режим доступа: [www/URL: http://www.oracle.com/technology/products/database/oracle10g/index.html/](http://www.oracle.com/technology/products/database/oracle10g/index.html/) – 09.09.2009 г. – Назв. с экрана. 8. Oracle Database 10g Release 2 Spatial [Электронный ресурс]. – Режим доступа: [www /URL: http://www.oracle.com/technology/products/spatial/10gr2_tech_info.html/](http://www.oracle.com/technology/products/spatial/10gr2_tech_info.html/) – 09.09.2009 г. – Назв. с экрана. 9. Jadex Tutorial [Электронный ресурс]. – Режим доступа: [www/URL: http://jadex.informatik.uni-hamburg.de/docs/jadex0.96x/tutorial/index.single.html](http://jadex.informatik.uni-hamburg.de/docs/jadex0.96x/tutorial/index.single.html/) – 09.09.2009 г. – Назв. с экрана. 10. The Foundation of Physical Intelligent Agents [Электронный ресурс]. – Режим доступа: [www/ URL: http://www.fipa.org/](http://www.fipa.org/) – 09.09.2009 г. – Назв. с экрана. 11. Агентно-ориентированные системы: от формальных моделей к промышленным приложениям [Электронный ресурс]. – Режим доступа: [www/ URL: http://window.edu.ru/window_catalog/pdf2txt?p_id=27142&p_page=7/](http://window.edu.ru/window_catalog/pdf2txt?p_id=27142&p_page=7/) – 09.09.2009 г. – Назв. с экрана. 12. Гаврилова, Т. А. Базы знаний интеллектуальных систем [Текст]: учеб. / Т.А. Гаврилова, В.Ф. Хорошевский. – СПб.: Питер, 2000. – 384 с. 13. Hendler J. Agents and the Semantic Web, IEEE Intelligent Systems [Текст] / Hendler J. – John Wiley and Sons. – Apr. 2001, pp. 30–37. 14. Codehaus XFire – Home [Электронный ресурс]. – Режим доступа: [www/URL: http://xfire.codehaus.org/](http://xfire.codehaus.org/) – 09.09.2009 г. – Назв. с экрана.

Поступила в редколлегию 28.10.2009

УДК 004.048

Репозитарій інтелектуального аналізу даних / Т.Б. Шатовська, В.Б. Репка, І.В. Каменева, М.М. Марченко // Бюлетень інтелекту: наук.-техн. журнал. – 2009. – № 2 (71). – С. 75-78.

Проводиться порівняльний аналіз сучасних статистичних репозитаріїв. Представлена загальна схема роботи репозитарія інтелектуального аналізу даних, а також варіанти використання репозитарія інтелектуального набору даних.

Л. 2. Бібліогр.: 14 назв.

УДК 004.048

Data Mining Repository / Т.В. Shatovska, V.B. Repka, I.V. Kamenieva, M.M. Marchenko // Bionics of Intelligence: Sci. Mag. – 2009. – № 2 (71). – P. 75-78.

In the article dissects modern statistical repositories. A general scheme of data mining repository is present and variants of usage of Data mining repository also.

Fig. 5. Ref.: 14 items.

УДК 519.686.4



ИСПОЛЬЗОВАНИЕ МЕТОДА УЧЕБНЫХ ПРОЕКТОВ НА БАЗЕ ТЕХНОЛОГИИ СОЦИАЛЬНЫХ СЕТЕЙ

О.Н. Купин¹, Н.С. Лесная², И.Н. Келеберда³

¹ХНУРЭ, г. Харьков, Украина, devilforce@yandex.ru

²ХНУРЭ, г. Харьков, Украина

³ХНУРЭ, г. Харьков, Украина, i.keleberda@gmail.com

В ходе написания данной статьи был произведен анализ текущей ситуации в области учебных и научных социальных сетей, исследована структура учебного проекта, возможность адаптации его в области обучения информационным технологиям. Произведено описание архитектуры учебной социальной сети и определены требования для использования в данной сети метода учебных проектов. Построена модель учебного проекта в рамках концепции учебных социальных сетей.

PROBE, УЧЕБНЫЙ ПРОЕКТ, МЕТРИКИ, ЧАСТЬ, ОБЪЕМ ПРОГРАММНОГО ПРОДУКТА

Введение

Использование в системе образования дистанционных курсов привело педагогов к необходимости использования сетевых технологий в учебном процессе. Доступность Интернета обусловила целесообразность его использования в учебных и научных целях, а развитие социальных сетей и их большая популярность в последнее время приводит к необходимости спроецировать их в научно-педагогическое пространство. Одним из направлений использования социальных сетей является построение модели учебных и научных проектов в системе социальных сетей. Это очень перспективное направление в свете реалий современного обучения.

Анализ психолого-педагогической литературы показывает, что проектная деятельность и педагогов, и учащихся находится в процессе становления, обобщения эмпирических фактов и результатов исследования. Отсутствие необходимых знаний, умений, психологических обоснований проектирования — одна из серьезнейших проблем современного образования как учащихся, так и педагогов. Известно, что знание, оторванное от понимания, временами не имеющее ничего общего с опытом учащегося, практически не используется. В такой ситуации метод проектов, как один из способов, позволяет педагогу решить сложную задачу смены традиционного образования с безликой формы в личностную культуру учащегося [2].

Таким образом, использование метода проектов в учебном процессе высшей школы является целесообразным. Несмотря на то, что технология социальных сетей появилась относительно недавно, она имеет сейчас очень широкую аудиторию и представляет собой альтернативную модель общения, обмена информацией, а главное — систематизацию необходимой информации и легкого доступа к ней. Анализируя эти факторы, можно сделать вывод, что социальная сеть является наилучшим решением, которое необходимо взять как базовое для реализации инструментов, и разработать модель метода учебных проектов в интерактивном пространстве.

1. Анализ предметной области и постановка задачи

В данной статье исследуется метод учебных проектов в педагогической деятельности на примере построения модели создания программных продуктов в сфере обучения информационным технологиям.

Учебный проект с точки зрения студента — это деятельность, позволяющая проявить себя, попробовать свои силы, приложить свои знания, принести пользу и показать публично достигнутый результат; это деятельность, направленная на решение интересной проблемы, сформулированной самими учащимися в виде цели и задачи, когда результат этой деятельности — найденный способ решения проблемы — носит практический характер, имеет важное прикладное значение и, что весьма важно, интересен и значим для самих разработчиков.

2. Построение модели учебного проекта создания программного продукта

Введем пять этапов разработки учебного проекта, смысл каждого определяется его названием и комплексом решаемых задач: предварительный этап, этап начала взаимодействия студентов, этап анализа предметной области и написания документации, этап кодирования и тестирования программного продукта, этап выпуска и презентации.

2.1 Предварительный этап

На предварительном этапе преподаватель создает или подбирает из существующих в системе пользователей (студентов) группу для осуществления проекта. Создает сущность «проект» в системе и регистрирует пользователей в этой сущности. После чего настраивает возможности (индивидуально или всей группе) работы пользователей в проекте. Важным на этом этапе также является определение преподавателем сроков начала и окончания проекта. Дата и время начала проекта открывает для студентов, определенных на конкретный проект, доступ ко всем ресурсам, хранилищу данных, чатам, календарю, а дата окончания проекта — закрывает

этот доступ. Преподаватель также назначает время вводной дискуссии в проект и финальной дискуссии по окончанию проекта.

2.2 Этап начала взаимодействия студентов

Данный этап начинается с момента старта проекта, дату которого определил преподаватель на предыдущем этапе. После старта проекта за короткий промежуток времени преподаватель проводит вводную дискуссию со студентами, во время которой определяет цели проекта, результат для конечного пользователя программного продукта, дает советы по организации работы в команде, специфике предметной области, предполагаемым источникам информации. Итогом данной дискуссии является документ с общими характеристиками программного продукта, который подготавливает преподаватель заранее и который представляет собой техническое задание.

Этот документ представляет собой обзор того продукта, который заказчик (в случае учебного проекта – это преподаватель) хочет получить от группы разработчиков. В рамках этого документа описываются цели, поставленные заказчиком, определяется список терминов (гlossарий), определяются основные проблемы, которые необходимо решить в ходе выполнения проекта, описываются пользователи, которые будут работать с этой системой после введения ее в эксплуатацию и их потребности. Определяются также возможности и ограничения изготавливаемого продукта, показатели качества, требования к документации и другие требования к системе.

После данной дискуссии студентами подготавливается график дискуссий, в рамках которых должны решиться следующие задачи: разбивка проекта на части, количество которых равно количеству человек в команде, назначение каждой части на участника команды, также определяются временные рамки для каждого из последующих этапов, согласовываются варианты коммуникации и других организационных вопросов.

2.3 Этап анализа предметной области и написания документации

После того, как система разбита на отдельные логические части и каждая часть закреплена за каждым отдельным студентом, начинается этап уже непосредственно разработки программного продукта. Первым этапом разработки является написание документации по программному продукту. В связи с тем, что за каждым закреплена отдельная часть с отдельным функционалом, после написания и согласования общей части документации происходит независимое написание документации по каждой из частей проекта. С периодичностью, определенной на предыдущем этапе, проходят дискуссии между членами команды с текущими отчетами о проделанной работе с момента послед-

ней дискуссии и с планами о проделываемой работе с момента окончания дискуссии и до начала следующей. Во время данных дискуссий происходит обмен опытом, полученной новой информацией, обсуждаются трудности в том или ином этапе разработки. Данные дискуссии начинают проводиться с данного этапа разработки и заканчиваются уже на последнем этапе, во время сдачи проекта.

Общим для всех частей системы и для дальнейшей работы с ней необходимо написать общие части документации для всей системы: необходимо определить список акторов в системе и их взаимодействие. Актор – это внешний фактор и его действия несут недетерминированный характер. В роли актора может выступать и программная система, если она инициирует выполнение некоторых работ, удовлетворяющих поставленной цели системы. Актор, как абстракция внешнего объекта, может быть человеком или внешней системой. В модели системы актор может быть представлен классом, а пользователь – экземпляром класса. Если актором является другая программная система, то он будет представлять ее интересы. При этом одно лицо может быть экземпляром нескольких акторов.

Если актор находится вне системы, то он взаимодействует с ней через сценарий, который инициализирует последовательность операций для выполнения системы. Когда пользователь, как экземпляр актора, вводит определенный символ для старта экземпляра соответствующего сценария, это приводит к выполнению ряда действий в системе, которые заканчиваются тогда, когда экземпляр сценария находится или в состоянии ожидания очередного входного символа или завершения сценария. Данная информация базируется на документе технического задания, предоставляемом преподавателем.

Затем для каждой из частей определяется реестр вариантов использования.

Варианты использования это – описание последовательности действий, которые может осуществлять система в ответ на внешние воздействия пользователей или других программных систем. Варианты использования отражают функциональность системы с точки зрения получения значимого результата для пользователя, поэтому они точнее позволяют ранжировать функции по значимости получаемого результата.

Далее каждый определяет функциональные требования и бизнес-требования для своей части в целом. Функциональные требования к системе определяют действия системы, которые она должна выполнять. Функциональные требования реализуются через функции системы. Под функцией автоматизированной системы подразумевается совокупность действий автоматизированной системы, направленная на достижение определенной цели

или аспект определенного поведения системы, а под задачей — функция или часть функции системы, представляющая собой формализованную совокупность автоматических действий, выполнение которых приводит к результату заданного вида. Бизнес-требования описывают, каким образом разрабатываемая система связана с достижением бизнес-целей организации и что она должна для этого делать. Можно сказать, что бизнес-требования являются своего рода задачами, которые должна решать автоматизированная система для достижения целей своего создания.

Последним на данном этапе разработки документации является определение интерфейса пользователя. Здесь описывается, как должен взаимодействовать конечный пользователь с модулями системы.

После определения требований к системе разработка проекта переходит в стадию планирования. Известно, что чем точнее будут определены требования, тем более точным будет произведено планирование. Основной единицей планирования являются следующие оценки: времени, необходимого количества рабочих часов разработчиков, инженеров по качеству, аналитиков и менеджеров, необходимых для успешного завершения проекта. Изначально оценка затрачиваемого времени является неопределенной единицей, так как неясно, насколько будет велик конечный продукт. Чем раньше будет произведена оценка затрачиваемого времени, тем менее точной она будет. Преимущество в использовании определенных методов оценки затрачиваемого времени в том, что команда имеет механизм, который посредством приобретения опыта может улучшать процесс реализации проектов: имеется основа для сбора информации по данному методу, возможность использовать статистику оценки времени для более точного планирования в последующих проектах. В модели учебного проекта было решено использовать метод PROxy Based Estimating (метод оценивания времени, основанный на частичных объектах), сокращенно PROBE.

Метод оценивания времени, основанный на частичных объектах, базируется на том, что каждый студент создает компонент, опираясь при этом на те, которые он создавал ранее. Такие компоненты называются частями. Части — это единица программной реализации, представляющая файл, метод, объект, скрипт, страницу.

Критерии для точного определения части:

- размер части должен зависеть от необходимых усилий для завершения проекта;
- содержание части должно быть конечным и иметь определенную величину;
- часть должна быть легко отображаемой визуально и интуитивно понятна;

- часть должна быть легко изменяема;
- часть должна быть чувствительна ко всем изменениям в процессе реализации.

Для того чтобы часть была полезной, она должна быть тесно связана с требованиями к разработке программного продукта. Объективно оценив размер части, можно оценить размер необходимой работы по компоненту, который эта часть отображает. Определить эффективность части можно на основе истории создания разработанных ранее подобных частей. Также очень важно, чтобы язык программирования, стили и категория разрабатываемого приложения, которые будут использоваться в проекте, были определены до этапа расчета оценки затрачиваемого времени.

Многие потенциальные типы частей могут удовлетворять поставленным выше ограничениям. Программный метод является очевидным примером для использования его как части. Действительно, очень удобно, имея детальный дизайн, принять метод как часть и определять затрачиваемое время на написание метода (части). Также возможно использовать в виде части программные окна, веб-страницы, объекты, скрипты, разделы документации и так далее.

Принципы объектно-ориентированного проектирования позволяют предположить, что объекты будут как нельзя лучше подходить на роль частей. Объекты — это физические сущности, размер которых может быть определен программно. Удобство использования объектов, как частей, связано с тем, что количество строк кода (которые являются важной метрикой) соотносится с часами, затраченными разработчиком, так же как и объекты, которые в свою очередь состоят из строк кода. Таким образом, объекты, являющиеся в конечном итоге физическими величинами, очень удобно использовать в качестве частей. При таком выборе необходимо учитывать наличие опыта написания кода именно на том языке, который является требованием планируемого объекта.

По результатам оценки времени, затрачиваемого на проект, необходимо создать концептуальный дизайн системы. Этот дизайн предусматривает разбиение системы на объекты вплоть до названия методов данной системы.

Следующей не менее важной частью разработки документации является написание детального дизайна системы, декомпозированного вплоть до самых нижних объектов системы. Этот документ является завершающим в цепочке спецификаций и позволяет полностью сосредоточиться на стадии кодирования системы. Спецификация описывает низкоуровневую организацию продукта. Здесь для каждого модуля разработчик должен определить все требования, включая передаваемые параметры, глобальные структуры и переменные, вызыва-

емые подпрограммы и так далее. Эта информация важна для программистов, которые параллельно реализуют различные модули, взаимодействующие друг с другом. Хорошо выполненная техническая спецификация снимает все проблемы, возникающие при объединении отдельных модулей в единое целое. Вторая часть низкоуровневого проектирования – создание псевдокода – является трудной, но очень важной частью процесса. Программисты не любят писать псевдокод, поскольку им кажется, что непосредственное кодирование осуществить быстрее. Часто это действительно так. Для маленьких процедур и модулей код размером в несколько строк написать проще, при этом не надо тратить время на представление одних и тех же действий дважды. Однако для крупных проектов с большой командой усилия, затраченные на эту работу, с лихвой окупятся на этапе сопровождения продукта, делая его более понятным и модифицируемым с меньшими затратами.

Имея концептуальный дизайн, название каждого объекта и категорию, далее необходимо найти объекты в базе данных предыдущих проектов, которые являются аналогичными объектам создаваемой системы.

В каждой из частей рекомендуется использовать код, написанный ранее. Тогда при оценке времени данной части можно ссылаться на количество затраченного времени на подобную часть из истории.

Если студент находит созданные ранее объекты или процедуры, которые будут отвечать концептуальному дизайну, он может использовать их повторно. Для любых объектов, при их использовании повторно, необходимо внести их имена и размер в строку повторно использованных объектов. Данный метод оценки времени подразумевает два типа повторно используемых объектов: повторно используемые объекты из библиотеки объектов и вновь создаваемые вспомогательные объекты, которые пересекаются с данной частью своим функционалом. Данные объекты определяются как повторно используемые [1].

Оцененные новые и измененные строки кода рассчитываются – $N = B0 + B1 * E$. Метод линейной регрессии производит расчет статистических параметров. Согласно этому методу определяется 2 параметра β_0 и β_1 , используемые в расчетах размера программного продукта.

Формула линейной регрессии, используемая для расчетов размера программного обеспечения, показана ниже. Параметры β_0 и β_1 рассчитываются с учетом статистических данных по предыдущим проектам:

$$\beta_1 = \frac{\left(\sum_{i=1}^n x_i y_i \right) - (n x_{avg} y_{avg})}{\left(\sum_{i=1}^n x_i^2 \right) - (n x_{avg}^2)}, \quad (1)$$

$$\beta_0 = y_{avg} - \beta_1 x_{avg}. \quad (2)$$

Таким образом, размер программного обеспечения рассчитывается следующим образом:

$$PS = \beta_{0size} + \beta_{1size} * (E). \quad (3)$$

Затраченное время:

$$DT = \beta_{0time} + \beta_{1time} * (E). \quad (4)$$

Вспомогательные и регрессионные параметры определяются для того, чтобы в конечном итоге при подсчете объема программы учесть опыт предыдущих разработок.

Прогнозируемый интервал представляет собой допустимые рамки оцененного объема или времени. Он не является обязательным для попадания, а всего лишь предположением. Широкий диапазон статистических данных будет иметь более широкий прогнозируемый интервал. Формула расчета прогнозируемого интервала попадания показана ниже, где n – количество данных точек; стандартное отклонение от регрессионной линии; t – распределенное значение вероятности p (70%); k – это оцененный размер, avg – среднее значение данных.

$$Range = t(p, n) \sigma \sqrt{1 + \frac{1}{n} + \frac{(x_k - x_{avg})^2}{\sum_{i=1}^n (x_i - x_{avg})^2}}. \quad (5)$$

Стандартное отклонение измеряется разницей данных вокруг регрессионной линии. Широко разбросанные данные будут иметь более высокое стандартное отклонение, чем тесно сгруппированные. Ниже приведена формула расчета стандартного отклонения, которое равняется квадратному корню из дисперсии:

$$Variance = \sigma^2 = \frac{1}{n-2} \sum_{i=1}^n (y_i - \beta_0 - \beta_1 x_i)^2. \quad (6)$$

Значение прогнозируемого интервала предназначается для определения качества оценки с целью последующего ее использования в других частях.

Оцененное количество строк кода рассчитывается таким образом – $E = BA + NO + M$, где BA – суммарные базовые добавления частей, NO – суммарные новые объекты, M – количество строк, которые необходимо изменить для использования метода PROBE, что напрямую зависит от качества данных, которые используются. Если статистические данные имеют большой разброс в значениях, то прогнозируемый интервал окажется достаточно большим.

Рассчитывается верхняя и нижняя граница прогнозируемого интервала следующим образом: верхняя – $UPI = P + Range$ (где P – это оцененный размер части, а $Range$ – прогнозируемый интервал), нижняя – $LPI = P - Range$ (или 0).

Задача разработки спецификаций проекта считается выполненной, если пользователь четко сформировал свои ожидания к результирующему продукту, а программист способен однозначно реализовать эти ожидания в продукте без любых других знаний о проекте, руководствуясь только спецификациями.

2.4 Этап кодирования и тестирования программного продукта

На этом этапе работа над документацией считается завершенной, и группа студентов приступает к реализации задокументированной функциональности. Отведенное время на этот этап разбивается на более мелкие итерации, во время которых производится планирование поэтапной реализации каждой части каждым студентом. После чего наступает сама реализация всех частей студентами.

По окончании реализации части кодирования и выпуска первой версии программы для бета-тестирования студенты начинают тестировать части друг друга. Причем очень важно, чтобы каждый студент тестировал не свою часть, а часть коллеги по команде. Во время тестирования и нахождения дефектов каждый дефект заносится в систему фиксации дефектов. После занесения дефекта студент, в чьей части он был найден, получает уведомление о том, что в его части был найден дефект. После чего происходит процедура исправления и последующей перепроверки неисправности. Таким же образом перед проведением тестирования студент пишет тест кейсы по своей части.

В период тестирования программного продукта очень важным аспектом является понятие качества и производительности кода. Качеством программного обеспечения называется характеристика программного обеспечения как степени его соответствия требованиям. Конечным определением же является удовлетворение потребностей пользователей. Причем потребностей, а не предпочтений, что является уже второстепенным фактором. Существует следующая иерархия потребностей пользователей: выполнение обязательных заданий, удовлетворение требований производительности, удобность в использовании, экономичность, современность и надежность.

Для того чтобы быть полезным, программное обеспечение должно быстро устанавливаться и легко запускаться, последовательно и правильно обрабатывая допустимые и недопустимые действия пользователя и не содержать критических ошибок. Ошибки, критически не важные для пользователя, являются таковыми, пока они не влияют на ключевые операции, не вызывают неудобства, не вызывают затраты времени и денег, не приводят к потере достоверности полученных результатов. Программное обеспечение является единственной современной технологией, качество которой зависит от тестирования.

Основной способ нахождения и исправления дефектов — блочное тестирование (unit testing), интеграционное и системное тестирование, командный анализ и анализ разработчика по написанному коду.

Во время персонального анализа написанного кода разработчик должен найти и устранить дефекты до начала тестирования. Персональный анализ написанного кода наиболее эффективен, когда он структурирован и подлежит внешней оценке. Даже на уровне анализа написанного кода нахождение ошибок является более эффективным, чем при проведении различных тестов инженерами по качеству.

Эффективность блочного тестирования составляет в среднем нахождение от двух до четырех дефектов в час и покрывает около 50% всех дефектов системы. Анализ кода (code review) находит от шести до десяти дефектов в час и покрывает около 70% дефектов системы.

Во время проведения анализа кода разработчик должен: обнаружить неправильное или странное поведение системы, выяснить какая последовательность действий приводит к неправильному поведению системы, найти, где в коде программы происходит инициация неправильного поведения системы, выяснить, к каким ошибкам может привести такое поведение. Такой способ обнаружения дефектов занимает обычно много времени. С помощью обзора и проверки кода программист может проследить за программной логикой отдельных модулей, знать, в какой части тестируемого модуля он находится во время выявления дефекта, что программа должна делать.

Основной целью анализа кода является найти как можно больше дефектов до тестирования и выпуска версии. Для решения данной задачи программист должен использовать стандарты кодирования (coding standard), следовать полноте требований к дизайну системы, оценить и по возможности улучшить процесс анализа кода, использовать список для контроля (checklist) проведенного анализа кода.

Анализ кода будет более эффективным только в том случае, если список для контроля проведенного анализа будет построен с учетом опыта разработчика. Необходимо использовать статистические данные для выбора столбцов списка, собирать и анализировать данные во время анализа кода, сопоставить список для контроля проведенного анализа кода с существующим опытом разработчика.

В дополнение к анализу кода разработчику необходимо провести анализ дизайна системы (design review). Требованиями к данному анализу:

- созданный дизайн должен легко поддаваться анализу;
- анализ дизайна системы должен быть последовательным;

– реализованная логика должна полностью отвечать требованиям, описанным в дизайне системы.

Стадии проверки дизайна системы:

– анализ на соответствие реализованных функциональных требований к требованиям, описанным в дизайне системы;

– проверка общей структуры и потока выполнения программной логики;

– проверка логических конструкций на корректность;

– проверка на отказоустойчивость и безопасность;

– проверка правильности использования вызова функций, методов и процедур;

– проверка на корректность использования переменных, параметров методов, типов и файлов.

Для того чтобы сделать процесс анализа системы более эффективным, необходимо определить размерность оценки качества.

Размерность выхода (yield) зависит от доли дефектов в продукте, которые были найдены на данном этапе, и эффективности исправления. Размерность выхода измеряет эффективность анализа дизайна и программного кода, ревизий, компиляций версии и тестирования и рассчитывается следующим образом: размерность выхода (для фазы) = $100 * (\text{количество найденных дефектов}) / (\text{количество найденных дефектов} + \text{количество ненайденных дефектов})$. Размерность выхода – это процент найденных дефектов до определенной фазы к общему количеству дефектов, найденных за фазу. Размерность выхода наиболее эффективна в случае, если разработчики и инженеры по качеству документируют все дефекты, то есть дефекты, найденные во время анализа дизайна и кода системы, тестирования и компиляции версии.

Для повышения продуктивности использования полученных значений контроль измерений должен быть доступен во время процесса измерений, например: количества частей, проанализированных за час, количества дефектов, найденных за час, количества дефектов, найденных за определенную часть фазы.

2.5 Этап выпуска и презентации

На этом этапе у студентов остается небольшое количество времени для того, чтобы подготовить презентацию и папку с документами. После подготовки этих документов подходит стадия окончательной сдачи проекта, когда все работы закончены и остается лишь презентовать и оценить проект.

Прием проекта производится преподавателем или группой преподавателей. Во время презентации проекта должны присутствовать все студенты, работающие над одним проектом. С докладом о своей части должен выступить каждый участник проекта, так как ориентироваться в системе в целом обязан каждый из них. По окончании презентации

проекта ставится одна оценка всей группе, независимо от того, кто успешнее выступил. Оценивается объем выполненной работы, качество выполнения частей в целом, качество работы в команде, ответственность к поставленной задаче.

Выводы

В ходе выполнения данной работы:

– был произведен анализ текущей ситуации в области социальных сетей;

– рассмотрены существующие социальные сети, их достоинства и недостатки;

– исследована структура учебного проекта;

– процесс выполнения учебного проекта разбит на этапы;

– были распределены роли пользователей в системе и их значение;

– произведено описание архитектуры учебной социальной сети и определены необходимые требования для использования данной сети, к методу учебного проекта;

– построена модель учебного проекта в рамках концепции учебных социальной сети.

Список литературы: 1. Вингер, К. М. Разработка требований к программному обеспечению [Текст] / К. М. Вингер. – М. : Русская редакция, 2004. – 554 с. 2. Краля, Н. А. Метод учебных проектов как средство активизации учебной деятельности учащихся [Текст] : учеб. пособие / под ред. Ю. П. Дубенского. – Омск : Издательство ОмГУ, 2005. – 59 с. 3. Кармайкл, Э. Быстрая и качественная разработка программного обеспечения [Текст] / Э. Кармайкл. – М. : Компьютерная литература, 2003. – 404 с. 4. Халл, Э. Разработка и управление требованиями [Текст] / Э. Халл, К. Джексон, Д. Дик. – М. : Телелоджик, 2005. – 240 с. 5. Лешек, А. М. Анализ и проектирование информационных систем с помощью UML 2.0 [Текст] / А. М. Лешек. – К. : Вильямс, 2008. – 816 с.

Поступила к редколлегии 28.10.2009

УДК 519.686.4

Використання метода навчальних проектів на базі технології соціальних мереж / О. М. Купин, Н.С. Лесна, І.М. Келеберда // Біоніка інтелекту: наук.-техн. журнал. – 2009. – № . – С. 79-84.

Темою даної статті є дослідження і побудова моделі навчального проекту на базі технології соціальних мереж у навчальних та наукових закладів України задля інтеграції у європейський науково-технічний простір.

Метою роботи є підвищення якості створення програмного продукту під час виконання навчального проекту.

Бібліогр.: 5 найм.

UDK 519.686.4

Using of the education project method based on the social network technology. / O. N. Kupin, N. S. Lesnaya, I. N. Keleberda // Bionics of Intelligence: Sci. Mag. – 2009. – № 2 (71). – P. 79-84.

One of the most important problem of the modern education is practice implementation of theoretical knowledge. This problem solved with education project method improve with social network conception.

Ref.: 5 items.

УДК 004.93



ГИБРИДНЫЙ МУЛЬТИАГЕНТНЫЙ ПОДХОД ДЛЯ ПОСТРОЕНИЯ НЕЙРО-ФАЗЗИ МОДЕЛЕЙ В ЗАДАЧАХ ДИАГНОСТИКИ

А.А. Олейник¹, С.А. Субботин²

¹ Запорожский национальный технический университет, г. Запорожье, Украина, olejnik@zntu.edu.ua;

² Запорожский национальный технический университет, г. Запорожье, Украина, subbotin@zntu.edu.ua

Исследуется мультиагентный подход для моделирования сложных объектов и систем. Разработан гибридный мультиагентный метод структурно-параметрического синтеза нейро-фаззи сетей. Решена практическая задача моделирования основных параметров авиадвигателей.

ОПТИМИЗАЦИЯ, КОЛЛЕКТИВНЫЙ ИНТЕЛЛЕКТ, АГЕНТ, ВЫБОРКА, НЕЧЁТКАЯ ЛОГИКА, НЕЙРО-ФАЗЗИ СЕТИ

Введение

В настоящее время диагностики и прогнозирования работы сложных технических объектов и процессов являются необходимым элементом производства, поскольку позволяют сократить издержки от выпуска дефектной или ненадёжной продукции. Одним из весьма эффективных средств автоматизации принятия диагностических решений являются нейро-фаззи сети, преимуществами которых являются логическая прозрачность и незначительные затраты материальных ресурсов на программную и аппаратную реализацию [1, 2]. Благодаря этому нейро-фаззи сети нашли широкое применение на практике [3, 4].

Однако нерешённой остаётся задача оптимального синтеза нейро-фаззи сети, обеспечивающего наилучшее соотношение структуры сети и настраиваемых параметров. Существующие методы преимущественно направлены на решение задачи параметрической идентификации, в то время, как структура задаётся экспертным путём. Помимо того, традиционные методы настройки нейро-фаззи сетей, основанные на использовании техники обратного распространения ошибки, выдвигают требование дифференцируемости функций принадлежности, что, в свою очередь, ограничивает выбор структуры сети. В связи с этим актуальной является задача разработки новых методов структурно-параметрического синтеза нейро-фаззи сети, которые бы обеспечивали одновременно и подбор структуры сети, и её настройку без ограничений на функции принадлежности и другие параметры сети.

Для решения задачи структурно-параметрического синтеза нейро-фаззи сетей предлагается использовать методы коллективного интеллекта [5, 6]. Среди наиболее известных методов коллективного интеллекта можно выделить метод муравьиных колоний (Ant Colony Optimization, ACO) [7, 8], метод пчелиной колонии (Bee Colony Optimization, BCO) [9, 10], метод оптимизации с использованием роя частиц (Particle Swarm Optimization, PSO) [11] и другие [12, 13]. Основными преимущества-

ми методов, основанных на моделировании коллективного интеллекта, являются: мультиагентная реализация, отсутствие требований к оптимизируемому параметрам, способность решать задачи как дискретной, так и непрерывной оптимизации.

Целью данной работы является исследование методов коллективного интеллекта, а также разработка мультиагентного метода структурно-параметрического синтеза нейро-фаззи сетей, основанного на моделировании коллективного интеллекта с прямой и не прямой связью между агентами. В работе исследуются два наиболее известных метода роевого интеллекта, использующие различные подходы к реализации связей между агентами: метод АСО и метод ВСО, основанные на моделировании не прямой и прямой связи между агентами соответственно.

1. Постановка задачи

Пусть задано множество нечётких продукционных правил Ω вида “ЕСЛИ $\langle A \rangle$, ТО $\langle B \rangle$ ”, а также задана архитектура нейро-фаззи сети, на основе которой должна моделироваться аппроксимируемая зависимость, описанная при помощи заданного множества нечётких правил Ω . Тогда задача структурно-параметрического синтеза нейро-фаззи сети заключается в том, что для заданной архитектуры сети (Сугено или Мамдани) необходимо определить оптимальную структуру путём получения редуцированной базы нечётких правил Ω' и настроить коэффициенты функций принадлежности Θ , при которых обеспечивалась бы допустимая ошибка прогнозирования $\varepsilon_{\text{доп.}}$: $\varepsilon < \varepsilon_{\text{доп.}}$.

2. Сравнительный анализ методов АСО и ВСО

Метод АСО [6–8] представляет собой эвристический, итеративный, мультиагентный метод интеллектуальной оптимизации с не прямой связью между агентами, основанный на случайном поиске. Основной идеей метода АСО является моделирование поведения в процессе фуражировки муравьёв, реализующиеся в методе с помощью программных агентов. Особенностью такой фуражировки является выделение феромонов, которые

агенты оставляют на пути при своём перемещении. Феромоны также в процессе работы метода испаряются с течением времени. Таким образом достигается то, что на наилучшем пути остаётся большее количество феромонов, поскольку добавление феромонов происходит чаще, чем испарение. А так как выбор пути для перемещения агентов основывается на информации о количестве феромонов, то агенты выбирают лучший путь.

Метод ВСО также является эвристическим, итеративным, мультиагентным методом интеллектуальной оптимизации, основанным на случайном поиске, но с прямой связью между агентами. Данный метод моделирует процесс фуражировки пчёл. Главной особенностью данного метода является дифференциация задач агентов. В связи с этим существуют агенты трёх типов: разведчики, которые ищут новые решения, занятые фуражиры, разрабатывающие уже найденные решения (что обеспечивает локальную оптимизацию) и незанятые фуражиры, которые ничем не заняты, а ожидают момента, когда смогут начать участвовать в поисках лучшего источника нектара. За счёт такого разделения функций агентов достигается углублённое изучение уже найденных решений (с помощью занятых фуражиров) и изучение пространства поиска в целом (с помощью агентов-скаутов). Связь между агентами в методе пчелиной колонии обеспечивается за счёт моделирования виляющего танца агентами, что обеспечивает прямую связь между агентами.

Можно видеть, что метод АСО и метод ВСО обладают рядом сходных свойств:

- стохастичность: оба метода представляют собой методы случайного поиска;
- мультиагентность: методы реализуются с помощью агентно-ориентированного подхода;
- итеративность: для нахождения оптимума необходимо выполнить достаточно большое количество итераций;
- гарантированность оптимизации: сходимость к оптимальному решению гарантируется, но время сходимости заранее не определено;

- скорость нахождения оптимального значения выше, чем у традиционных методов оптимизации;
- трудность теоретического анализа процесса нахождения оптимума;
- зависимость от начальных параметров.

Между методом АСО и методом ВСО присутствуют также определённые различия. Данные различия приведены в табл. 1.

Исходя из представленных различий между методом АСО и методом ВСО, можно сделать вывод, что метод ВСО лучше применять для задач непрерывной оптимизации, поскольку в нём выполняется дифференциация агентов, за счёт которого достигается улучшенное изучение локальных уже найденных областей с источниками нектарами, а метод АСО лучше применять для задач дискретной оптимизации, поскольку в нём выполняется динамическое изменение состояния среды посредством добавления и испарения феромона, что позволяет исключить худшие и выделить лучшие дискретные значения параметров.

3. Метод структурно-параметрического синтеза, основанный на мультиагентном подходе с прямой и непрямой связью между агентами

Структурно-параметрический синтез нейрофаззи сетей включает в себя два основных этапа: структурную идентификацию и параметрическую идентификацию сети. При этом эти два вида идентификации должны взаимно влиять друг на друга. Данный подход может быть реализован при конвейерной работе двух основных этапов.

Поскольку мультиагентный подход с непрямой связью хорошо зарекомендовал себя при решении задач дискретной оптимизации [6–8], то предлагается применить его для структурной идентификации, в то время как для параметрической идентификации предлагается использовать мультиагентный подход с прямой связью между агентами. Таким образом, предлагается использовать гибридный мультиагентный подход с прямой и непрямой связью между агентами для структурно-параметрической идентификации нейро-фаззи сетей.

Таблица 1

Различия между методом АСО и методом ВСО

Критерий	АСО	ВСО
Класс решаемых задач	Дискретная оптимизация	Дискретная и непрерывная оптимизация
Локальная оптимизация	Отсутствует	Выполняют агенты специального типа – занятые фуражиры
Связь между агентами	Непрямая: с помощью феромона, который характеризует внешнюю среду	Прямая: посредством выполнения виляющего танца, который выполняют определённые агенты
Дифференциация задач агентов	Отсутствует	Присутствует: одни агенты ищут новые решения, другие – разрабатывают старые
Динамическое изменение состояния среды	Выполняется посредством изменения количества феромона	Отсутствует

Структурная идентификация предполагает преобразование входной базы нечётких правил, поскольку структура нейро-фаззи сети изоморфна базе нечётких правил, то есть структура определяется базой нечётких правил. В связи с этим структурная идентификация будет выполняться в пространстве нечётких термов, которые будут объединяться в новые правила. В свою очередь параметрическая идентификация предполагает выбор наиболее оптимальных параметров функций принадлежности и весов нечётких правил.

Разработанный гибридный мультиагентный метод структурно-параметрического синтеза нейро-фаззи сетей с прямой и непрямой связью между агентами можно представить в виде последовательности следующих шагов.

Шаг 1. Инициализация. Задать параметры работы метода: $\beta, \rho, \gamma, t_b^{\max}, t_a^{\max}, t^{\max}$. Установить счётчики итераций: общий счётчик $t=0$; счётчик итераций параметрической идентификации $t_b=0$; счётчик итераций структурной идентификации $t_a=0$. Установить начальную структуру $FN(0)$, соответствующую исходной базе нечётких правил.

Шаг 2. Установить счётчик итераций: $t = t + 1$. Задать верхние и нижние пределы изменения значений для всех параметров нечёткой модели: $range^{\max}$ и $range^{\min}$.

Шаг 3. Создание агентов с начальными случайными решениями. Создать B_s агентов, размещённых в случайных позициях пространства поиска:

$$x_j^i = rand \cdot (range_j^{\max} - range_j^{\min}) + range_j^{\min}, \\ \forall i = \overline{1, B_s}, j = \overline{1, vCnt},$$

где x_j^i – координата в j -ом измерении i -го агента; $rand$ – случайное число в интервале $[0; 1]$.

Для всех позиций, в которые были помещены агенты случайным образом, рассчитывается их полезность. В данном случае полезность точки пространства поиска определяется ошибкой прогнозирования модели с параметрами, которые соответствуют значениям координат данной позиции:

$$prof^i = \varepsilon [FN(x_1^i, x_2^i, \dots, x_{vCnt}^i)], \forall i = \overline{1, B_s},$$

где $prof^i$ – полезность точки пространства поиска, в которой находится i -ый агент.

Установить текущую итерацию параметрической идентификации $t_b = 0$, текущую температуру $T = T_{init}$, количество агентов-разведчиков $B = B_s$.

Шаг 4. Установить: $t_b = t_b + 1$.

Шаг 5. Выбор рабочих агентов. Отбор рабочих выполняется за счёт моделирования процедуры имитации отжига, то есть агент относится к рабочим, если выполняется следующее условие:

$$\exp\left(-\frac{|prof^i - prof^{best}|}{T}\right) > rand, \forall i = \overline{1, B},$$

где $prof^{best}$ – полезность позиции, в которой обеспечивается наименьшая ошибка прогнозирования.

Шаг 6. Скрещивание. Данный шаг позволяет создать агентов, которые бы находились в позициях с лучшими характеристиками за счёт пересечения между координатами позиций определённой группы агентов. Для скрещивания используются отобранные на шаге 5 рабочие агенты и агент, который находится в лучшей точке пространства поиска. На основе скрещивания создаются агенты, которые перемещаются в позиции, определяемые следующим образом:

$$x_j^{new} = x_j^{work} + rand(-1; 1) \cdot (x_j^{work} - x_j^{best}), \\ \forall j = \overline{1, vCnt}, work = \overline{1, B_w},$$

где B_w – количество агентов, которые были отнесены к рабочим; $rand(-1; 1)$ – случайное число, выбранное из интервала $[-1; 1]$.

Для созданных новых агентов рассчитывается полезность их пребывания в выбранных точках пространства поиска:

$$prof^{new} = \varepsilon [FN(x_1^{new}, x_2^{new}, \dots, x_{vCnt}^{new})], \forall i = \overline{1, B_n},$$

где B_n – количество агентов, созданных в результате скрещивания.

Шаг 7. Моделирование обмена информацией. В результате выполнения обмена информацией достигается вербовка одних агентов другими. Завербованные агенты используются для улучшенного изучения уже найденных областей пространства поиска. Моделирование обмена информацией производится путём выполнения последовательности шагов 7.1–7.4.

Шаг 7.1. Нормирование полезностей агентов:

$$np^i = 1 - \frac{prof^i}{prof_{\max} - prof_{\min}},$$

где np^i – нормированная полезность i -го агента; $prof_{\max}$ и $prof_{\min}$ – максимальная и минимальная полезности среди всех агентов, соответственно.

Шаг 7.2. Добавление шумов в полученные нормированные полезности и корректировка нормированных полезностей:

$$np^i = \begin{cases} 1, & \text{если } np^i + w^i > 1; \\ np^i + w^i, & \text{если } e_n < np^i + w^i < 1; \\ 0, & \text{если } 0 < np^i + w^i < e_n, \end{cases}$$

где w^i – шум в полезности. Шум равномерно распределён в пределах $(-w; +w)$. Значение w выбирается экспериментально, e_n – минимальный порог полезности, выбираемый экспериментально.

Шаг 7.3. Определяются достоинства решения каждого агента:

$$L^i = \max\{np^i - \eta \cdot \overline{np}, 0\},$$

где L^i – достоинство решения i -го агента; η – коэффициент, управляющий влиянием величины \overline{np} на L^i ; \overline{np} – среднее значение нормированной полезности всех агентов, допущенных к моделированию обмена информацией.

Шаг 7.4. Выбор агентов, которые считаются выполнившими вербовку других агентов для улучшенного изучения найденных ранее решений. Агент считается выполнившим вербовку, если:

$$\frac{L^i}{\beta} > \gamma \cdot \overline{np}, \forall i = \overline{1, B_c},$$

где B_c – количество агентов, допущенных к вербовке; $\beta > 0$ – коэффициент, понижающий влияние достоинства решения на вероятность выполнения вербовки; $\gamma \in (0; 1)$ – пороговой коэффициент, определяющий, насколько выше должно быть достоинство решения i -го агента относительно средней полезности решений всех B_c агентов.

Шаг 8. Перезапуск агентов, выполняющих параметрическую идентификацию. Создаются агенты, которые будут рассматриваться как агенты-разведчики для следующей итерации.

К новым агентам-разведчикам будут относиться:

– агенты, выполнившие посредством танца вербовку;

– агенты, которые стали занятыми фуражирами вследствие вербовки. Поскольку такие агенты должны выполнять улучшенное изучение уже существующей области с решением, то при создании решений для данных агентов должны учитываться решения завербовавших их агентов. В связи с этим для завербованных агентов решение создаётся следующим образом:

$$x_j = x_j^d + range \cdot rand - \frac{range}{2}, \forall j = \overline{1, vCnt},$$

где $range$ – это предел, в котором величина переменной может отклоняться от значения данной j -ой переменной x_j^d в решении агента, моделировавшего обмен информацией;

– агенты, решение которых создаётся случайным образом:

$$x_j = rand \cdot (range_j^{\max} - range_j^{\min}) + range_j^{\min}, j = \overline{1, vCnt}.$$

Также для всех созданных агентов рассчитывается полезность выбранного решения.

Шаг 9. Обновление параметров, влияющих на параметрическую идентификацию:

- изменяется текущая температура: $T = \alpha \cdot T$;
- изменяется предел $range$:

$$range = range \cdot \frac{t_b^{\max} - t_b}{t_b^{\max}}.$$

Шаг 10. Проверка на достижение оптимальных параметров для текущей структуры сети. Провер-

ка считается успешной, если выполняется одно из следующих условий:

- достигнуто максимальное количество итераций: $t_b = t_b^{\max}$;
- текущая температура равна конечной температуре: $T = T_{final}$;
- ошибка прогнозирования лучшей нейро-фаззи сети находится в допустимых рамках для параметрического синтеза: $\varepsilon_{best} < \varepsilon^*$.

Если проверка на останов дала успешный результат, тогда установить $\varepsilon(t) = \varepsilon_{best}$ и выполнить переход на шаг 11, в противном случае – на шаг 4.

Шаг 11. Сравнение качества прогнозирования текущей структуры нейро-фаззи сети с качеством прогнозирования предыдущей сети. Проверяется условие:

$$\varepsilon(t-1) < \varepsilon(t),$$

где $\varepsilon(t)$ – ошибка лучшей нейро-фаззи сети, полученной на итерации t .

Если условие выполняется, тогда установить: $FN(t) = FN(t-1)$; $\varepsilon(t) = \varepsilon(t-1)$. То есть, если на предыдущей итерации была получена нейро-фаззи сеть с лучшими характеристиками, то на следующей итерации должно выполняться сравнение с сетью, полученной на этой итерации. Если же вновь полученная сеть (на итерации t) характеризуется лучшими характеристиками, то на следующей итерации сравнение будет производиться с ней. За счёт такого подхода выявляется лучшая структура и лучшие параметры для неё на протяжении всей работы метода.

Шаг 12. Создание пространства поиска. Для каждого класса возможных значений выходной переменной создаётся отдельный граф поиска, по которому перемешаются отдельные агенты. У каждого графа поиска вершинами являются лингвистические термы, а рёбра формируются при перемещении агентов и наличие ребра свидетельствует о том, что связанные им термы входят в правило, сформированное агентом, который перемешался между ними. Для каждой вершины каждого графа поиска рассчитывается эвристическая значимость:

$$\eta_p^q = \frac{\sum_{o=1}^N \min(\mu_p(o), \mu_q(o))}{\sum_{o=1}^N \mu_q(o)}, \forall p = \overline{1, T}, q = \overline{1, K},$$

где η_p^q – значение эвристической значимости лингвистического терма p для описания класса q ; o – экземпляр входной выборки, содержащей N экземпляров; $\mu_p(o), \mu_q(o)$ – значение функции принадлежности объекта o терму p и классу q , соответственно; T – количество лингвистических термов; K – количество классов.

В каждом графе поиска каждому узлу графа поиска ставится в соответствие начальное значение количества феромонов: τ_{init} :

$$\tau_p^q(1) = \tau_{init}, \forall p = \overline{1, T}, q = \overline{1, K},$$

где $\tau_p^q(1)$ – значение количества феромонов для p -го терма в пространстве поиска для q -го класса на первой итерации поиска.

Шаг 13. Установить счётчик итераций для структурной идентификации: $t_a = t_a + 1$.

Шаг 14. Перемещение агентов.

Шаг 14.1. Выбор следующего терма для перемещения в него. При перемещении j -ый агент принимает решение о перемещении в k -ый лингвистический терм i -го пространства поиска при выполнении следующего условия:

$$P_k^{i,j} = \frac{\eta_k^i \cdot \tau_k^i(t_a)}{\sum_{p \in R^j} \eta_p^i \cdot \tau_p^i(t_a)} > rand,$$

где $P_k^{i,j}$ – вероятность добавления k -го терма в правило j -го агента в пространстве поиска для i -го класса; R^j – множество термов, которые могут быть добавлены в правило j -го агента. Формирование данного множества определяет вид правил, которые могут составляться в процессе поиска, то есть предполагается, что если правило может включать выражения типа ИЛИ, то после добавления терма из данного множества исключается только данный терм, если же предполагается, что правило не может включать выражения типа ИЛИ, то кроме выбранного терма, исключаются и все термы, использующиеся для описания соответствующего атрибута.

Шаг 14.2. Проверка на окончание перемещения. Перемещение может быть закончено, если агент уже посетил все возможные вершины графа поиска или если достигнуто удовлетворительное количество покрываемых объектов сформированным правилом. В связи с этим для каждого экземпляра o , относящегося к классу i , рассчитывается степень соответствия сформированного правила R_j и экземпляра o :

$$match(R_j, o) = \min(matchAttr(R_j^1, o^1), \dots, matchAttr(R_j^p, o^p), \dots, matchAttr(R_j^Q, o^Q)),$$

где $match(R_j, o)$ – степень соответствия между правилом j -го агента R_j и экземпляром o ; $matchAttr(R_j^p, o^p)$ – мера соответствия между p -ым атрибутом в правиле R_j и соответствующим атрибутом экземпляра o :

$$matchAttr(R_j^p, o^p) = \begin{cases} 1, & \text{если } R_j^p = \emptyset; \\ \max_q(\min(\mu_q(R_j^p), \mu_q(o^p))), & \forall q = \overline{1, Q^p}, \text{ иначе,} \end{cases}$$

где q – отдельный терм, относящийся к области описания атрибута p ; Q^p – количество термов, относящихся к области описания атрибута p .

Далее проверяется условие:

$$match(R_j, o) \geq inMatchMin,$$

где $inMatchMin$ – заданный параметр, который определяет, какое минимальное значения соответствия является достаточным, чтобы считать, что правило R_j в достаточной степени описывает объект o .

Если условие выполняется, то считается, что данный объект o покрывается правилом R_j .

На основании получаемых данных увеличивается счётчик $cntMatch$, в котором хранится количество экземпляров, покрываемых правилом R_j .

Проверяется условие:

$$cntMatch \geq inCntMatchMin_i,$$

где $inCntMatchMin_i$ – предельное минимальное количество экземпляров i -го класса, которое должно покрываться правилом.

Если указанное условие выполняется, то считается, что правило покрывает необходимое количество экземпляров, и j -ый агент завершил своё перемещение, в противном случае – перемещение продолжается.

Шаг 15. Формирование баз правил. Создаётся $nBases$ баз правил, при этом для описания каждого класса выходного значения выбирается одно правило из соответствующего пространства поиска.

Шаг 16. Оценка качества сформированных баз правил. Для оценки качества баз правил для каждого экземпляра исходной выборки определяется правило, наиболее подходящее данному экземпляру, на основе которого определяется расчётный выходной класс для экземпляра. Далее оценивается качество базы правил:

$$Q = \frac{cntMatch}{N},$$

где $cntMatch$ – количество экземпляров, для которых класс был определён верно с помощью заданной базы правил; Q – качество прогнозирования класса экземпляров на основе соответствующей базы правил.

Шаг 17. Добавление феромонов. Добавление феромонов осуществляется для каждого терма, входящего в правило, которое в свою очередь входит в базу правил RB :

$$\tau_p^q(t) = \tau_p^q(t) + Q_{RB} \cdot \tau_p^q(t), \forall p \in R, \forall R \subset RB,$$

где $\tau_p^q(t)$ – количество феромонов для терма p в пространстве поиска для класса q , который определяется с помощью соответствующего правила.

Шаг 18. Испарение феромонов. Испарение феромонов выполняется в соответствии с формулой:

$$\tau_p^q(t+1) = \rho \cdot \tau_p^q(t), \forall p = \overline{1, T}, q = \overline{1, K},$$

где ρ – коэффициент испарения, который задаётся при инициализации.

Шаг 19. Проверка на останов работы агентов по структурной идентификации. Проверка на останов может считаться успешной в одном из двух следующих случаев:

- достигнуто необходимое качество полученной базы нечётких правил: $Q_{high} \geq Q_{threshold}$;
- достигнуто максимальное количество итераций для структурной идентификации: $t_a = t_a^{max}$.

Если проверка на завершение работы по структурной идентификации дала успешный результат, то выполняется переход к шагу 21, в противном случае – к шагу 20.

Шаг 20. Перезапуск агентов, выполняющих структурную идентификацию. Все данные о перемещении агентов во всех пространствах поиска обновляются, агенты размещаются в случайные точки пространств поиска. Переход к шагу 13.

Шаг 21. Проверка на останов работы метода. Останов предлагается производить при выполнении одного из следующих условий:

- построенная нейро-фаззи модель характеризуется требуемой точностью прогнозирования: $\varepsilon < \varepsilon_{доп.}$;
- достигнуто максимальное количество итераций: $t = t^{max}$.

Если проверка на останов была успешной, то выполняется переход к шагу 22, в противном случае – переход к шагу 2.

Шаг 22. Останов.

Таким образом, предложенный метод обеспечивает конвейерный подход при структурно-параметрической идентификации нейро-фаззи модели, за счёт чего обеспечивается взаимовлияние результатов, получаемых при параметрической и структурной идентификации, что обеспечивает получение нейро-фаззи модели, характеризующейся высокой точностью прогнозирования.

4. Эксперименты и результаты

Предложенный гибридный мультиагентный метод структурно-параметрического синтеза нейро-фаззи моделей был программно реализован в среде разработки Matlab 7.0 и использовался для проведения экспериментов по моделированию параметров работы авиадвигателей.

С целью сокращения сроков и снижения стоимости испытаний авиационных двигателей проводятся исследования взаимосвязи основных параметров двигателей (оборотов турбины компрессора n_{tk} , температуры газа перед турбиной T_3 , расхода газа через турбину G_t , температуры на входе в двигатель $T_{вх}$, количества ступеней $N_{ст}$, угла установки лопаток входного направляющего аппарата $\alpha_{ВНА}$, приведенной мощности $N_{пр}$, расхода воздуха G_v , степени сжатия воздуха π_k), наблюдаемых в процессе испытаний, от проходных сечений сопловых аппаратов CA1, CA2, CA3, CA4, контролируемых в производстве, и адиабатического давления B_o .

Сбор экспериментальных данных был выполнен для 125 испытанных двигателей. Результаты получены для номинального режима.

Параметры испытанных двигателей и результаты проливки сопловых аппаратов приведены в табл. 2.

Для построения модели использовалась нейро-фаззи сеть Сугено. Синтезируемая сеть содержала 5 входов, каждый из которых описывался с помощью трёх функций принадлежности. Также сеть содержала базу правил и выход с соответствующей функцией принадлежности.

В качестве функции принадлежности для входов использовалась колоколообразная функция вида:

$$\mu(x, a, b, c) = \frac{1}{1 + \left| \frac{x - c}{a} \right|^{2b}},$$

Таблица 2

Фрагмент обучающей выборки

№	n_{tk}	T_3	G_t	$T_{вх}$	$N_{ст}$	$\alpha_{ВНА}$	$N_{пр}$	G_v	π_k	B_o	CA1	CA2	CA3	CA4
1	95,17	812,1	379,2	-4,8	99,22	6,1	1760,2	8,34	8	763,5	34,86	54,8	98,4	102,19
2	95,03	873,3	379,6	0,7	99,79	6	1753,6	8,3	7,96	763	34,66	55,61	98,4	102,8
3	95,48	835,7	392,8	-1,7	100,01	5,8	1742,1	8,46	7,86	764,5	34,66	55,49	99,15	103,57
4	95,23	830,3	387,6	10,6	100	6,9	1720	8,41	7,75	746	34,88	54,35	99,05	103,57
5	95,02	820,8	382,3	11,2	98,81	7	1731,8	8,35	7,75	743,5	34,88	54,98	98,4	102,5
6	95,31	824,8	384,8	10,8	99,84	5,8	1719,7	8,48	8,04	762	34,88	54,83	99,15	103,37
7	95,21	818,5	382,3	13,3	98,29	5,8	1725,8	8,45	7,98	762,5	34,88	54,35	98,66	101,84
8	95,27	792,5	385,6	14,6	98,61	6,8	1727,3	8,09	7,98	757	34,8	54,41	98,95	103,47
9	95,1	804,2	387,5	15,2	98,86	6,3	1702,5	8,11	7,95	760,5	34,7	54,53	99,15	102,45
10	95,02	812,2	386,9	15,1	99,01	6,5	1728,4	8,41	7,87	749	34,84	54,5	98,4	102,3
...
121	95,15	811	391	6,9	99	6,3	1834	8,28	7,96	754	34,88	54,59	98,35	103,57
122	95,4	795	374	4,3	99,5	5,8	1725	8,27	7,71	762	34,72	54,15	98,55	103,28
123	95,25	840	390	8,6	98,82	6	1737	8,36	7,96	765,5	34,74	54,92	98,9	102,45
124	95,2	819	392	7,2	98,5	6,1	1726	8,35	7,61	766,5	34,6	54,77	99,1	103,57
125	94,8	824	392	7,9	98,88	7,4	1748	8,35	7,85	766	34,6	55,82	98,95	102,63

где a, b, c – параметры, определяющие вид функции: параметр c определяет положение центра функции, значение параметра b обычно положительное.

Полученные значения параметров функций принадлежности для каждого из входов в случае построения нейро-фаззи модели для режима «номинальный» для моделирования n_{tk} представлены в табл. 3.

Средняя относительная ошибка построенной нейро-фаззи модели для n_{tk} составляет $E_{\text{абс.},\text{ннс}} = 0,00001$, что значительно ниже ошибки аналогичной регрессионной модели ($E_{\text{абс.},\text{регр.}} = 0,19505$), что подтверждает эффективность применения нейро-фаззи моделей на практике при диагностировании авиадвигателей.

Средняя относительная ошибка для данных тестовой выборки составила 0,00094, что незначительно хуже ошибки, рассчитанной для данных обучающей выборки. Исходя из рассчитанных значений ошибок нейро-фаззи модели для обучающей и тестовой выборок, можно судить о ее высоких обобщающих способностях и адекватности решаемой задаче.

Для остальных исследуемых зависимостей построены аналогичные нейро-фаззи модели, которые также характеризовались низкой ошибкой при прогнозировании.

Таблица 3

Значения параметров функций принадлежности входов полученной нейро-фаззи модели для прогнозирования n_{tk} в режиме «номинальный»

№ входа	№ μ_f	a	b	c
1	1	6,625	2	740
	2	6,625	2	753,25
	3	6,625	2	766,5
2	1	0,15	2	34,3
	2	0,15	2	34,6
	3	0,15	2	34,9
3	1	0,4775	2	54,15
	2	0,4775	2	55,105
	3	0,4775	2	56,06
4	1	0,875	2	96,7
	2	0,875	2	98,45
	3	0,875	2	100,2
5	1	0,52	2	101,79
	2	0,52	2	102,83
	3	0,52	2	103,87

Выводы

С целью решения актуальной задачи структурно-параметрического синтеза нейро-фаззи сетей в статье разработан гибридный мультиагентный метод с прямой и непрямой связью между агентами.

Научная новизна работы заключается в том, что был разработан новый метод структурно-параметрического синтеза, основанный на гибридном

мультиагентном подходе, который заключается в том, что выполняется конвейерная работа структурной и параметрической идентификации с использованием агентов с непрямой связью и агентов с прямой связью между ними, соответственно. При этом результаты, получаемые на одном из этапов, влияют на работу другого этапа, за счёт чего обеспечивается нахождение оптимальной структуры нейро-фаззи сети и оптимальных параметров к ней.

Практическая ценность работы заключается в том, что решена актуальная задача моделирования основных параметров авиадвигателей в зависимости от проходных сечений сопловых аппаратов и высоты лопаток. Полученные результаты свидетельствуют об эффективности предложенного метода, поскольку построенные модели характеризовались низкой ошибкой прогнозирования.

Работа выполнена в рамках госбюджетной НИР «Научно-методические основы и математическое обеспечение для автоматизации и моделирования процессов управления и поддержки принятия решений на основе процедур распознавания и эволюционной оптимизации в нейросетевом и нечеткологическом базисах» (№ гос. регистрации 0106U008621).

Список литературы: 1. Zadeh, L. Fuzzy sets / L. Zadeh // Information and Control. – 1965. – № 8. – P. 338–353. 2. Mamdani, E. H. An Experiment in Linguistic Synthesis with Fuzzy Logic Controller / E. H. Mamdani, S. Assilian // Int. J. Man-Machine Studies. – 1975. – Vol. 7, № 1. – P. 1–13. 3. Bellman, R. E. Decision-Making in Fuzzy Environment / R. E. Bellman, L. A. Zadeh // Management Science. – 1970. – № 4. – P. 141–160. 4. Yuan, Y. Induction of fuzzy decision trees / Y. Yuan, M. Shaw // Fuzzy Sets and Systems. – 1995. – № 69. – P. 125–139. 5. Dorigo, M. Optimization, Learning and Natural Algorithms / M. Dorigo. – Milano: Politecnico di Milano, 1992. – 140 p. 6. Субботін, С. О. Ітеративні, еволюційні та мультиагентні методи синтезу нечіткологічних і нейромережних моделей: монографія [текст] / С. О. Субботін, А. О. Олійник, О. О. Олійник; під заг. ред. С. О. Субботіна. – Запоріжжя: ЗНТУ, 2009. – 375 с. 7. Dorigo, M. The Ant System: Optimization by a colony of cooperating agents / M. Dorigo, V. Maniezzo, A. Colomi // IEEE Transactions on Systems, Man, and Cybernetics. – 1996. – Part B, № 26 (1). – P. 29–41. 8. Олейник, Ал. А. Сравнительный анализ методов оптимизации на основе метода муравьиных колоний [текст] / Ал. А. Олейник // Комп'ютерне моделювання та інтелектуальні системи: Зб. наук. праць / За ред. Д. М. Пізи, С. О. Субботіна. – Запоріжжя: ЗНТУ, 2007. – С. 147–159. 9. Passino, K. M. Modeling and analysis of nest-site selection by honey bee swarms: The speed and accuracy trade-off / K. M. Passino, T. D. Seeley // Behavioral Ecology and Sociobiology. – 2006. – № 59. – P. 427–442. 10. Nakrani, S. On Honey Bees and dynamic allocation in an internet server colony / S. Nakrani, C. Tovey // Adaptive Behavior. – 2004. – № 12. – P. 223–240. 11. Subbotin, S. A. PSO with Control of Velocity Change for Feature Selection / S. A. Subbotin, Al. A. Oleynik // Induc-

tive Modeling : Proceedings of 2nd International Conference ICIM'2008, 15–19 May 2008. – Kyiv, 2008. – P. 81–83.
12. *De Castro, L. N.* Artificial Immune Systems. Part I. Basic Theory And Applications / L. N. De Castro, F. J. Von Zuben. – Technical Report No. Rt Dca 01/99. – Brazil : Feec/Unicamp, 1999. – 95 p.
13. *Олейник, А. А.* Метод оптимизации на основе моделирования перемещения бактерий с применением эволюционных операторов [текст] / А. А. Олейник, С. А. Субботин // Бионика интеллекта: науч.-техн. журнал. – 2008. – № 2 (69). – С. 137–144.

Поступила в редколлегию 30.10.2009

УДК 004.93

Гібридний мультиагентний підхід для побудови нейро-фаззі моделей у задачах діагностики / О.О. Олійник, С.О. Субботін // Біоніка інтелекту: наук.-техн. журнал. – 2009. – № 2 (71). – С. 85-92.

У статті досліджується мультиагентний підхід для моделювання складних об'єктів та систем. Розроблено гібридний мультиагентний метод структурно-параметричного синтезу нейро-фаззі мереж. Вирішено практичну задачу моделювання основних параметрів авіадвигунів.

Табл. 3. Бібліогр.: 13 найм.

UDC 004.93

Neuro-fuzzy model synthesis based on hybrid multiagent approach for diagnostic problem solving / A.A. Oleynik, S.A. Subbotin // Bionics of Intelligence: Sci. Mag. – 2009. – № 2 (71). – P. 85-92.

Hybrid multiagent approach for complex objects and systems modeling is investigated in the paper. Hybrid multiagent method of structural-parametrical synthesis of neuro-fuzzy nets is developed. The real-world problem of air-engine parameters modeling is solved.

Tab. 3. Ref.: 13 items.

УДК 004.89



МОДЕЛЬ СТРУКТУРИРОВАННОГО ПРЕДСТАВЛЕНИЯ ЗНАНИЙ О ПРОЦЕДУРАХ

М.Ю. Вишняк¹, С.С. Довгань²

¹ХНУРЭ, г. Харьков, Украина, mvishnyak95@gmail.com

²ХНУРЭ, г. Харьков, Украина, svtlana.dovgan@gmail.com

Рост роли знаний в эффективной деятельности предприятия и лавинообразное увеличение объемов знаний вообще диктует необходимость развития и совершенствования методов работы со знаниями. Знаниями обладает человек, а все, что можно выразить в символах – описания знаний. Предлагается модель, позволяющая структурировано аккумулировать и обновлять описания знаний о процедурах, а затем их использовать для ускоренного обучения персонала.

ЗНАНИЯ, УПРАВЛЕНИЕ ЗНАНИЯМИ, ПРОЦЕДУРЫ, ОПИСАНИЕ ЗНАНИЙ, ОБУЧЕНИЕ, СТРУКТУРНОЕ ПРЕДСТАВЛЕНИЕ ЗНАНИЙ

Введение

Последние десятилетия наблюдается очевидная трансформация мировой экономической системы: человечество переходит из индустриальной эпохи в информационную, в которой информация и знания занимают ведущие позиции. Становится недостаточным обладать лишь сырьем и финансовыми активами. Чтобы занять лидирующее положение на рынке, необходимо владеть уникальными знаниями, удельный вес которых в общей стоимости производимого продукта постоянно увеличивается.

Экономики, рынки и организации – это организмы, а не механизмы, и скорее биология, чем физика или инженерное искусство, – является подходящей метафорой и научным образцом для их исследования и понимания [1]. Предприятие теперь рассматривается не как механизм, созданный для функционирования в некоторых определенных условиях, которые считаются практически неизменными, а как биологический саморазвивающийся организм, приспособляющийся к постоянно изменяющимся условиям. Способность адаптироваться у предприятия, как и у организмов и их сообществ, во многом зависит и определяется знаниями: умением получать их, вырабатывать, сохранять и применять.

Скорость роста объемов знаний постоянно увеличивается (рис. 1), сокращается период времени, за который объем знаний удваивается. По статистике в первой половине XX века объем знаний удвоился в течение 50 лет, с 1950 – удваивался каждые 10 лет, с 1970 – каждые 5 лет, а с 1991 – ежегодно [2].

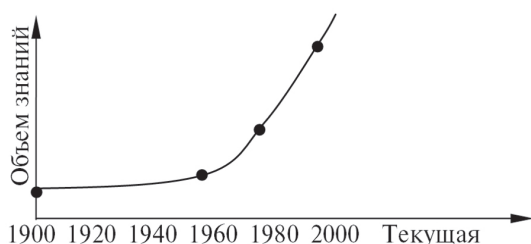


Рис. 1. Скорость удваивания знаний

Стремительный рост объемов знаний, необходимых для ведения конкурентоспособного бизнеса, вызвал интерес к управлению знаниями [3–6].

Для предприятий важным является достижение определенных целей, получение результата. Как известно, это достигается выполнением определенных совокупностей действий или процедур. Поэтому следует особое внимание уделять знаниям, касающимся процедур. Умение оперировать с большими объемами знаний, а именно: аккумулировать, сохранять, передавать и применять их, является насущной потребностью современных предприятий.

Целью работы является развитие методологии структурирования знаний, в частности структурирования знаний о процедурах.

Для достижения поставленной цели необходимо решить следующие задачи: 1) уточнить понятие «знание»; 2) провести анализ существующих форм представления знаний о процедурах, выявить их достоинства и недостатки; 3) построить модель структурированного представления описаний знаний.

1. Знания и их описания

На заре индустриализации инструменты, оборудование были существенно проще, чем сейчас. Рабочим не требовалось какое-то специальное образование. Их могли быстро обучить, показав, что и как делать, а дальше предполагалось только совершенствование навыков. Объекты, товары, технологии и оборудование разрабатывали немногочисленные инженеры, основу знаний которых составляло университетское образование. В процессе инженерной практики эти знания оттачивались и немного расширялись при знакомстве с новыми открытиями и другими разработками, которые в каждой из отраслей появлялись не часто. Поэтому считалось, что главным производителем знаний являются ученые, а инженеры находят знаниям практическое применение. Результаты своих работ ученые излагали на бумаге. Долгое время это и считалось знанием, что получило отражение,

например, в философском словаре [7]: «Знание – продукт общественной материальной и духовной деятельности людей; идеальное выражение в знаковой форме объективных свойств и связей мира, природного и человеческого». В этом определении полагается, что знание можно отделить от субъекта, обладающего им, и представить посредством знаков. Например учебник физики или биологии в соответствии с определением относится к знаниям.

По мере роста роли человеческого фактора и знаний в деятельности предприятий стала осознаваться необходимость повышения эффективности работы со знаниями, и старые представления перестали удовлетворять. Многие авторы обращались к обсуждению вопроса, а что же такое знание. Сейчас появляются авторы, которые считают, что знаниями обладает только субъект [8], а книга не есть знание, но из книги можно почерпнуть знание, приложив некоторое усилие. В попытках найти различие между понятиями «информация» и «знание» было предложено считать, что «знание – это целенаправленное координированное действие. Собственно знание можно продемонстрировать только действием. ... Знание процесса выпекания хлеба можно показать только его выпеканием и никак иначе» [9]. Из этого следует, что все, что не требует демонстрации действием к знанию не относится, а относится к информации. Например знания «живые организмы состоят из клеток» или «вода кипит при 100°C» следует относить к информации. Хотя это вопрос спорный и само понятие «информация» имеет достаточно много трактовок.

Известно, что интерпретация фактов, видение мира, выполнение аналогичных действий разными людьми часто не совпадают, а иногда даже сильно разнятся. Это следствие того, что различные люди, даже занятые в одной и той же области, обладают различающимися знаниями. Поскольку не все знания человеку удастся представить в знаковой форме и существует постоянная проблема расхождения между высказыванием и его смыслом, то несколько специалистов по одному и тому же вопросу составят различающиеся описания знаний. Поэтому, учитывая неразделимость знания и субъекта, вместо распространенного термина «формализованные знания» корректнее использовать термин «описание знаний».

Понимая существующую неоднозначность, будем придерживаться следующих посылок:

- знаниями обладает субъект, который применяет их для принятия решений и выполнения действий (физических и ментальных);

- субъект обладает способностями приобретать знания, хранить, применять, совершенствовать;

- знания субъекта, представленные в знаковой форме, будем считать описаниями знаний. Знания, релевантные получению некоторого результа-

та, могут иметь различные описания, и каждое из этих описаний не является полным отображением самого знания.

Одна из основных проблем работы со знаниями состоит в определении способов представления и использования знаний, которыми обладает эксперт. Проблема осложняется тем, что при достижении полезных выводов эксперты в основном опираются на неформализуемые сведения [10]. Наличие и тесная взаимосвязь явных и неявных знаний вызывает потребность в поиске методов перевода неявных знаний в явные. Формально это можно выразить следующим образом.

Пусть A – решаемая проблема (требуемый результат); N – количество экспертов, отобранных для выработки предложений по решению проблемы; M_i – множество явных (формально представляемых) знаний i -го эксперта; G_i – множество неявных знаний i -го эксперта, которые не удастся представить формально; $(m_{ia} \cup g_{ia})$ – подмножество знаний конкретного эксперта, необходимых для решения проблемы A .

Множество знаний, имеющихся у N экспертов, можно определить как $\bigcup_{i=1}^N (M_i \cup G_i)$ и, соответственно, подмножество знаний N экспертов, необходимых для решения проблемы A как $\bigcup_{i=1}^N (m_{ia} \cup g_{ia})$.

Тогда условие решения проблемы A имеет вид:

$$\bigcup_{i=1}^N (m_{ia} \cup g_{ia}) \subset \bigcup_{i=1}^N (M_i \cup G_i). \quad (1)$$

В частном случае, если $\bigcup_{i=1}^N G_i = \emptyset$ и, соответственно, $\bigcup_{i=1}^N g_{ia} = \emptyset$, тогда (1) превращается в

$$\bigcup_{i=1}^N m_{ia} \subset \bigcup_{i=1}^N M_i. \quad (2)$$

Из (1) и (2) следует, что при создании баз знаний предприятий нужно стремиться к (2), то есть на первый план выходит проблема формализации знаний и перевода максимально возможного количества неявных знаний в явные.

2. Обучение на основе описаний знаний

Одним из применений описания знаний является обучение персонала, работающего на определенном предприятии.

Каждому предприятию требуется персонал, обладающий знаниями, позволяющими выполнить работу на высокопрофессиональном уровне. Высокая скорость изменений окружающей среды, быстрый моральный износ оборудования и высокие темпы устаревания технологий приводят к потребности постоянного обучения персонала.

Многие ведущие компании придерживаются стратегии обучения персонала непосредственно на рабочих местах, тем самым сокращая временные затраты на обучение.

Основная задача каждого из сотрудников предприятия – достижение поставленной перед ним цели, получение требуемого результата. Между знаниями и результатом существует тесная связь (рис. 2).



Рис. 2. Отношения между знаниями и результатом

Недостающие для получения результата знания субъект может приобрести в процессе обучения. Многие способы обучения используют в той или иной форме описания явных знаний. Ускоренное узконаправленное обучение, которое нацелено на быстрое получение нужных знаний для конкретного результата, требует соответствующего компактного описания приобретаемых знаний. Результат связан с описаниями через обучение на основе описаний (рис. 3).



Рис. 3. Цепочка «описания-знания-результат»

Для усвоения знаний, изложенных в описании, (освоение необходимого знания) требуется методика, а сама процедура освоения заключается в выполнении действий, предписанных методикой (рис. 4.).

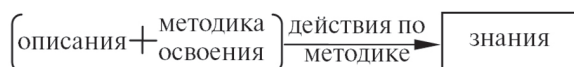


Рис. 4. Схема освоения знаний без наставника

3. Формы представления знаний о процедурах

Для описания знаний используются следующие основные формы:

– текстовая (словесная). Является наиболее простой формой. Но она не достаточно удобна для восприятия и анализа, так как текстовая информация воспринимается человеческим мозгом последовательно. Кроме того, человеческое сознание устроено так, что оно эффективно может работать только с образами. При восприятии и анализе текстовой информации человеческий мозг раскладывает ее на ряд образов, на что уходит дополнительное время и умственные усилия;

– математическая. Компактна, выразительна, но требует однозначности, точности и специальных знаний для изложения и восприятия;

– графическая. Наиболее удобная форма представления, так как сразу передает необходимый образ (модель), а словесная и числовая формы требуют мысленного воссоздания образа. Но при исключительно графической форме представления часто возникает потребность в дополнительных пояснениях и уточнениях из-за возникающих трудностей представления всех необходимых знаний в графическом виде;

– табличная. Может объединять в себе как текстовую, графическую, так и математическую формы. Обеспечивает компактность представления и удобство сопоставления.

Поскольку каждая из форм имеет наряду с достоинствами и некоторые недостатки, на практике чаще всего используется комбинированная форма представления знаний.

Вне зависимости от выбранной формы представления к модели знаний предъявляются следующие общие требования:

1. Модель должна обладать достаточной описательной мощностью, чтобы представить все извлеченные знания.
2. Модель должна точно соответствовать требованиям цели и не содержать избыточных элементов.
3. Модель должна быть понятна сотруднику, не имеющему особых навыков.

Наиболее простой формой представления знаний о процедурах является текст, но все значимые для предприятия процедуры содержат большое количество этапов. Чтобы корректно отобразить процедуру, этапы должны быть описаны в правильном порядке с учетом возможных переходов как к последующим этапам, так и к предыдущим. Недостатками использования текстовой формы для представления знаний о процедурах являются громоздкость описаний процедур и перенасыщенность модели, что значительно затрудняет их восприятие обучаемым и повышает вероятность неполного извлечения релевантной информации.

Второй сравнительно простой и наиболее наглядной формой представления процедурных знаний является блок-схема (графическое представление). Недостатками блок-схем являются их громоздкость, неудобство добавления новых процедур и невозможность хранения специфической информации.

Для решения сложных проблем используются методы системного анализа [11], предоставляющие в распоряжение визуальные графо-аналитические средства для построения моделей систем и процессов, позволяющие использовать комбинированные формы представления знаний. Существует две технологии системно-структурного анализа: технология 3 – View Modeling (3VM); технология, ос-

нованная на стандартах системного анализа ICAM Definition.

Технология системно-структурного анализа 3VM позволяет представить модель системы в виде трех взаимосвязанных диаграмм: функциональная диаграмма потоков данных DFD; информационная диаграмма «сущность-связь» ERD; динамическая диаграмма переходов состояний STD.

Технология, основанная на стандартах системного анализа ICAM Definition включает: стандарт функционального моделирования IDEF0 (SADT); стандарт информационного моделирования IDEF1; стандарт моделирования сценариев IDEF3 (PFDD и OSTN).

Среди наиболее признанных и часто используемых стандартных методов системного структурного анализа, которые могут быть использованы для моделирования процедурных знаний, выделяют стандарты системы федеральных стандартов ICAM Definition [12]:

– стандарт функционального моделирования IDEF0. Наиболее распространенная методика моделирования бизнес-процессов, относящаяся к семейству IDEF. Результатом применения IDEF0 к некоторой системе является модель этой системы, состоящая из иерархически упорядоченного набора диаграмм, текста документации и словарей, связанных друг с другом с помощью перекрестных ссылок. Проведенное моделирование процедуры увольнения персонала показало, что явным недостатком методики IDEF0 является отсутствие в модели элементов логических переходов;

– стандарт моделирования сценариев IDEF3 предназначен для описания потоков работ (Work Flow Modeling). Широко используется для создания моделей бизнес-процессов организации на нижнем уровне – при описании сценариев и работ, выполняемых в подразделениях и на рабочих местах. Моделирование процедуры увольнения персонала показало, что к недостаткам методики относится невозможность указания управляющего воздействия и механизма. Недостаток не является существенным, так как при низком уровне детализации практически на все процессы оказывает воздействие один и тот же механизм и применяется одно и то же управляющее воздействие, которые могут быть указаны на более высоких уровнях иерархии вплоть до контекстной модели в IDEF0 методике. Другим недостатком является то, что перекрестки имеют ограничение на количество входов и в этом случае происходит неоправданное нагромождение модели.

4. Модель представления знаний о процедурах

Процедура всегда нацелена на получение конкретного результата, то есть результат – выход процедуры. Человек, как целеполагающее существо,

исходит из требуемого результата и подыскивает процедуру, которая дает возможность его получить. Когда у человека появляются какие-то потребности или желания, он первоначально оперирует целями (результатами). Руководитель, обычно, дает задание подчиненным в терминах результата (цели): «подготовь отчет ...», «установите окна ...» и тому подобное. Поскольку цели не равноценны и могут быть выстроены в иерархию, когда достижение цели более высокого уровня осуществляется на основе достижения целей нижнего уровня, то традиционно строилось дерево целей. В общем случае цель более низкого уровня может быть составляющей для достижения более чем одной цели высшего уровня. Поэтому цели (результаты) будем упорядочивать в виде иерархической сети.

Для некоторых целей важным является порядок достижения подцелей. Например для цели «построить дом» подцелями являются: «заложить фундамент», «возвести каркас», «покрыть крышу», «выполнить внешнюю отделку», «подвести коммуникации», «выполнить разводку коммуникаций», «выполнить внутреннюю отделку». Схема последовательности подцелей (рис. 5) прикрепляется к соответствующей цели. Каждая цель (результат) (рис. 6) содержит связи с целями верхнего и нижнего уровней, а также имеет 5 слотов: «наименование», «описание», «параметры», «схема подцелей нижнего уровня», «ссылка на процедуру». Ссылка на процедуру позволяет быстро находить описание процедуры, производящей данный результат.

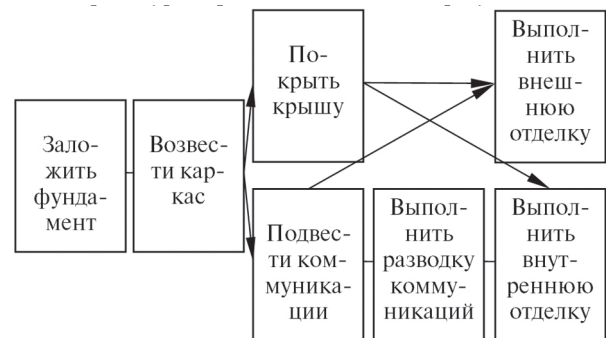


Рис. 5. Схема последовательности подцелей



Рис. 6. Логическая схема блока «цель»

Для освоения процедуры используется структурированное ее описание, логическая схема которого показана на рис. 7.

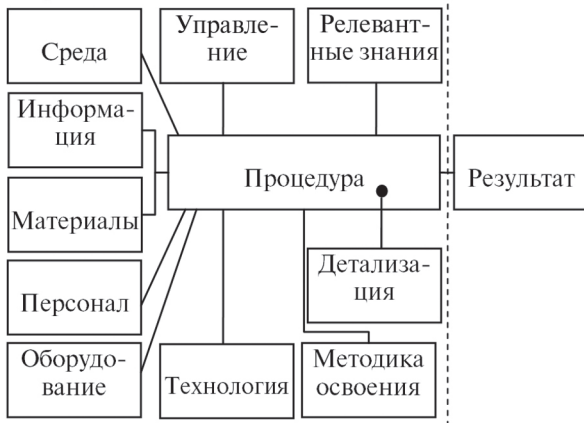


Рис. 7. Логическая схема структурированного описания процедуры

Каждая процедура имеет ссылку на результат, являющийся ее выходом. Соответственно и результат ссылается на производящую его процедуру. Блок описания каждой процедуры представлен фреймом. Слот «Детализация» содержит графическое представление действий, которые составляют процедуру. Некоторые из действий могут быть процедурами более низкого уровня. Графическое представление показывает последовательность выполнения действий, включает необходимые логические элементы, параметры и характеристики действий. Слот «Персонал» содержит требования к персоналу, выполняющему процедуру, а именно: должности, навыки, умения и базовые знания. Под технологией понимается описание того, каким образом выполняется процедура, методы, программное обеспечение, документация на рабочих местах и тому подобное. Слот «Оборудование» включает перечень оборудования, инструмента, оснастки и тому подобное необходимых для выполнения процедуры. Материалы – это вход процедуры, к ним относятся все материальные ресурсы, которые обрабатываются процедурой или необходимы для ее выполнения (например энерго-ресурсы). Второй вход – информация. Информация может быть ресурсом, поступающим на вход, если назначение процедуры – обработка информации. Но «Информация» может включать и некоторые сообщения, документы, необходимые для выполнения процедуры. Например для процедуры «Рыть траншею» необходим план траншеи с привязкой к местности. Важной составляющей описания процедуры являются релевантные знания. В явном виде этого блока не содержит ни один из существующих стандартов описания процедур. К релевантным знаниям относятся установки, правила, закономерности и тому подобное. Основное их назначение – дать возможность адаптировать процедуру к конкретной ситуации. Слот «Среда» содержит описание производственной среды, в которой выполняется процедура, а также технику

безопасности, экологическую безопасность и тому подобное. Поскольку процедура выполняется в реальных условиях под воздействием большого количества факторов, которые сложно предусмотреть и учесть, то для повышения гарантии получения заданного результата следует ввести управление. Кроме нормативно-правовых возможностей и ограничений, управление включает управляющие воздействия, контролируемые параметры, способы оценивания.

Модель предназначена для быстрого приобретения знаний, необходимых для получения результата на основе выполнения соответствующей процедуры. Поэтому в качестве компонента модели в описание включается методика освоения процедуры по ее описанию.

Структура предложенной модели позволяет легко ее модифицировать. Можно добавлять новые блоки в описание процедуры, если такие будут выявлены, а также может быть изменено содержание каждого блока. Поэтому эту модель можно отнести к эволюционирующим моделям.

Выводы

Непрерывное обучение персонала на рабочих местах является одной из важных составляющих повышения эффективности труда сотрудников и конкурентоспособности предприятия. Рассмотренная модель позволяет представлять описания знаний в удобной форме для их использования и обновления. Очевидно, что наличие неявных знаний накладывает определенные ограничения на процесс обучения, построенный исключительно на описаниях знаний, но структурированное представление описаний способствует расширению и усилению возможностей по самообучению сотрудников. Такой формат увеличивает уровень объективности описаний. Знания, представленные, например, в книгах, могут иметь различную логику изложения, и на них сильно влияет субъективная точка зрения автора. Используя предложенную модель, специалисты могут согласовывать отдельные структурные элементы, добиваясь тем самым повышения объективности.

Список использованной литературы 1. Желены М. Новая экономика [текст] / М. Желены // Информационные технологии в бизнесе / под общ. ред. М.Желены. – СПб.: Питер, 2002. – 1120 с. – С. 272–283. 2. Бондаренко, М.Ф. Моделирование и проектирование бизнес-систем: методы, стандарты, технологии: учеб. пособие [текст] / М.Ф. Бондаренко, С.И. Маторин, Е.А Соловьева / под общ. ред. О.Г. Попко. – Харьков: Компания СМИТ, 2004. – 272 с. 3. Глухов, В.В. Экономика знаний [текст] / В.В. Глухов, С.Б. Коробко, Т.В. Маринина. – СПб.: Питер, 2003. – 527 с. 4. Управление знаниями: пер. с англ – М.: Альпина Бизнес Букс, 2006. – 208 с. 5. Нонака И., Такеучи Х. Компания – создатель знания. Зарождения и развитие инноваций в японских фирмах [текст] / И. Нонака,

Х. Такеучи.: пер. с англ. – М.: Олимп-Бизнес, 2003. – 384 с. 6. *Wiig, K.M.* People-focused knowledge management: how effective decision making leads to corporate success / K.M. Wiig. – Boston: Butterworth-Heinemann, 2004. – 392 p. 7. *Философский словарь* [текст] / под ред. И.Т. Фролова. – М.: Политиздат, 1987. – 590 с. 8. *Морген, В.* Знание, определения понятия [текст] / В. Морген // Информационные технологии в бизнесе / под общ. ред. М.Желены. – СПб.: Питер, 2002. – 1120 с. – С. 198–210. 9. *Желены, М.* Знание против информации [текст] / М. Желены // Информационные технологии в бизнесе / под общ. ред. М.Желены. – СПб.: Питер, 2002. – 1120 с. – С. 211–218. 10. *Эрнест, К.Дж.* Один подход к экспертным системам управления с использованием нечеткой логики [текст] / К.Дж Эрнест // под общ. ред. Р. Р.Ягера. – М.: Радио и связь, 1986. – С. 133–143. 11. *Лямец, В.И.* Системный анализ. Вводный курс [текст] / В.И. Лямец, А.Д. Тевяшев. – Харьков: ХТУРЭ, 1998. – 252 с. 12. *Маклаков, С.В.* Моделирование бизнес-процессов в Win 4.0 [текст] / С.В Маклаков. – М.: ДИАЛОГ – МИФИ, 2002. – 224 с.

Поступила в редколлегию 15.10.2009

УДК 004.82

Модель структурированного подання знань про процедури / М.Ю. Вишняк, С.С. Довгань // Біоніка інтелекту: наук.-техн. журнал. – 2009. – № 2 (71). – С. 93-98.

Показано зростаючу роль знань у діяльності сучасного підприємства. Стрімкий зріст об'єму знань потребує

поліпшення методів та технологій роботи з ними. Обговорюється поняття «знання». Пропонується використовувати замість поняття «формалізовані знання» більш коректне поняття «опис знань». В загальному вигляді показано проблему формалізації знань. Розглянуто схеми навчання персоналу на основі описів знань. Проведено аналіз, виділено позитивні якості та недоліки існуючих форм представлення знань. Запропоновано модель структурованого представлення описів релевантних знань для здобуття вмінь виконувати процедуру.

Л. 7. Бібліогр.: 12 найм.

UDC 004.82

Model of structured presentation of knowledge about procedures / M.Y. Vishnyak, S.S. Dovgan' // Bionics of Intelligence: Sci. Mag. – 2009. – № 2 (71). – P. 93-98.

The increasing role of knowledge in activity of a modern enterprise is noted. The impetuous growth of knowledge extent demands improvement of methods and technologies involved in knowledge processing. The meaning of term «knowledge» is discussed. It is suggested to use more consistent term «knowledge description» instead of «formalized knowledge». The problem of knowledge formalization is given in the general case. Schemes of staff training based on knowledge descriptions were studied. Existing forms of knowledge representation were analyzed and their highs and lows were pointed out. Model of structured presentation relevant knowledge description, needed for procedural techniques acquisition, is proposed.

Fig. 7. Ref.: 12 items.

УДК 57:51-76; 57.02.001.57; 517.958:57



МОДЕЛИРОВАНИЕ ДИФFUЗИИ С ФАЗОВЫМ ПЕРЕХОДОМ НА СФЕРИЧЕСКОЙ ПОВЕРХНОСТИ

А.В. Клименко

ХНУРЭ, г. Харьков, Украина, klymenko@kture.kharkov.ua

В данной работе представлены результаты математического моделирования диффузии по сферической поверхности, сопряженной с формированием кластера реагирующими частицами. Математическая модель базируется на законах Фика. Численное решение задачи позволило получить зависимость размера кластера от времени и начальной концентрации частиц, а также установить диапазоны безразмерных параметров, в которых построенная модель является применимой к моделированию реальных систем.

ДИФFUЗИЯ, ЗАКОНЫ ФИКА, ДВИГАЮЩАЯСЯ ГРАНИЦА, ЧИСЛЕННОЕ МОДЕЛИРОВАНИЕ

Введение

Физическая диффузия молекул или химически-активных центров по поверхностям является важным и недостаточно исследованным феноменом, кроме случая плоских поверхностей. Это также относится к ситуациям, когда химические объекты не диффундируют (то есть не изменяют свое положение в пространстве), а обмениваются состояниями активации, что приводит к их кажущейся диффузии, как «прыгающая» диффузия электронов. Однако при современном развитии нанотехнологий создаваемые объекты обычно имеют замкнутые искривленные поверхности, на которых расположены активные центры или реагенты, которые могут диффундировать [1-4]. Аналогичные явления наблюдаются и в живом мире [5-9]. Во всех перечисленных случаях движущая сила, приводящая к возникновению диффузионных потоков, возникает в результате наличия определенной функциональности в одной специфической части искривленной поверхности. В этом отношении диффузия по искривленным поверхностям является чрезвычайно интересным и нетривиальным процессом для математического моделирования.

В данной работе рассмотрен характерный случай, представляющий собой парадигму целого класса задач. Он включает изотропную сферическую поверхность с диффундирующими по ней частицами, которые «реагируют» с последующим формированием активного кластера на поверхности [10]. Данная ситуация осложнена двумя факторами, которые прежде не рассматривались в совокупности: (i) двумерная поверхность является искривленной, и (ii) область пространства, доступного для диффузии, изменяется во времени из-за роста кластера за счет частиц, которые агломерируются на его границе.

Данное модельное представление отражает серию возможных экспериментальных ситуаций из разных областей. Например оно непосредственно соответствует диффузии протеинов в липидных мембранах естественных и искусственных вези-

кул. Кластеризация протеинов приводит к росту «кометного» хвоста, состоящего из молекул актина и используемого для перемещения везикулы внутри клетки [6-8, 10]. Та же парадигма применима к зарядке поверхности молекулы-дендримера электродом [11], а также к гетерогенному анализу, когда активные центры изначально наносятся на инертные наносферы, что позволяет достичь существенного каталитического эффекта [1-4], как, например, в современных средствах очистки автомобильных выхлопных газов.

Приведенные примеры иллюстрируют значительный интерес к разработке единой методологии для решения целого ряда задач из различных областей нано-наук, где активные центры подвергаются двумерной диффузии по замкнутым сферическим поверхностям и реагируют друг с другом с образованием активного кластера на одном из полюсов сферы.

Целью данной работы является исследование описанного модельного представления при помощи двух различных подходов. Один из них основан на использовании классических аналитических законов Фика, справедливых для макросистем, и численном решении соответствующей задачи. Однако особенности рассматриваемой проблемы, обусловленные движущейся границей активного кластера и сходимостью линий тока на полюсе сферы, противоположном центру кластера, приводят к необходимости преобразования пространства, доступного для диффузии. Второй подход основан на использовании броуновского движения. Несмотря на то, что он приводит к значительно большим длительностям расчета, чем первый аналитический подход, он позволяет учитывать эффекты, связанные с конечными размерами диффундирующих частиц.

1. Модель

Рассмотрим двумерную поверхность сферического тела радиуса R , на которой равномерно распределено N_0 сравнительно небольших час-

тиц, таких как молекулы, ионы, электроны и тому подобных. Предположим, что данные частицы перемещаются по поверхности случайным образом (причем все направления движения являются равновероятными) под действием некоторых случайных факторов (столкновений частиц друг с другом или другими телами, перемещений в результате туннелирования и тому подобных), а также обладают способностью объединяться в кластер (рис. 1). Кластеризация частиц приводит к возникновению ненулевого градиента концентрации частиц и, таким образом, их направленному движению. Для сохранения общности модели мы не будем рассматривать конкретные причины и механизмы кластеризации частиц.

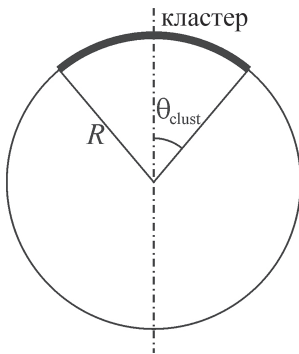


Рис. 1. Поперечное сечение сферической поверхности, доступной для диффузии и ограниченной формируемым кластером

При сделанных предположениях перемещение частиц по двумерной сферической поверхности является эквивалентным диффузии частиц с некоторым макроскопическим коэффициентом диффузии D . При достаточно большом количестве диффундирующих частиц в системе ее поведение может быть описано статистическими законами Фика для диффузии, сформулированными для сферической поверхности.

Также мы будем предполагать, что кластер является неподвижным, так как уже при агломерации нескольких частиц масса и размер кластера настолько превосходят эти же характеристики одной частицы, что относительная инерция кластера и вязкость среды предотвращают сколько-нибудь заметные перемещения.

Так как кластер представляет собой поверхность сферического сегмента, то соответствующий зенитный угол края кластера $\theta_{\text{clust}}(t)$ может быть легко вычислен при предположении, что центр кластера совпадает с «северным полюсом» сферы (см. рис. 1), для произвольного числа объектов $N(t)$, кластеризовавшихся ко времени t , и их известной общей площади (то есть $S^{\text{aggr}}(t) = N(t)b^2$, где b^2 – площадь поверхности, занимаемая одним объектом в кластере; это значение может отличаться от πr_0^2). Тогда в момент времени τ площадь поверхности кластера равна

$$S^{\text{aggr}}(t) = N(t)b^2 = 2\pi R^2(1 - \theta_{\text{clust}}(t)). \quad (1)$$

Следовательно,

$$\theta_{\text{clust}}(t) = \arccos \left[1 - \frac{S^{\text{aggr}}(t)}{2\pi R^2} \right] = \arccos \left[1 - 2\rho \frac{N(t)}{N_0} \right], \quad (2)$$

где $\rho = N_0 b^2 / 4\pi R^2$ – это часть сферической поверхности, покрытая кластером в пределе $\tau \rightarrow \infty$, то есть, когда все N_0 объектов, изначально присутствовавших на поверхности сферы, объединились в кластер. Следует отметить, что параметр ρ также представляет собой отношение начальной поверхностной концентрации свободных частиц и концентрации частиц в кластере. Действительно, начальная концентрация свободных частиц равна $N_0 / 4\pi R^2$, тогда как в кластере в любой момент времени концентрация равна $1/b^2$.

2. Формальная постановка задачи

В начальный момент времени диффундирующие объекты равномерно распределены по сферической поверхности с концентрацией $\Gamma_0 = N_0 / 4\pi R^2$. При $t > 0$ происходит агломерация частиц, и их концентрация внутри кластера является постоянной равной Γ_0 / ρ . Рост кластера с течением времени может быть представлен временной зависимостью зенитного угла $\theta_{\text{clust}}(t)$, определяющего положение границы кластера согласно уравнению (2).

Диффузия частиц по сферической поверхности описывается следующим дифференциальным уравнением в частных производных, записанным в сферических координатах:

$$\frac{\partial \Gamma}{\partial t} = \frac{D}{R^2} \left(\frac{\partial^2 \Gamma}{\partial \theta^2} + \text{ctg} \theta \frac{\partial \Gamma}{\partial \theta} \right), \quad (3)$$

где D – коэффициент диффузии, природа которого не имеет значения для формулировки задачи [10].

Начальные условия, соответствующие моменту времени $t = 0$, имеют следующий вид:

$$\Gamma(\theta, 0) = \Gamma_0; \quad (4a)$$

$$\theta_{\text{clust}}(0) = \theta_0, \quad (4b)$$

где θ_0 – зенитный угол, соответствующий начальному размеру кластера. Формально должно выполняться равенство $\theta_0 = 0$ в соответствии с предположением, что изначально кластера в системе нет. Однако подобное условие невозможно реализовать в численном алгоритме решения задачи из-за разрывности коэффициента $\text{ctg} \theta$ в правой части уравнения (3). Таким образом, введение очень малого ($\theta_0 \ll \pi$), но конечного значения θ_0 необходимо для успешного решения рассматриваемой задачи диффузии-кластеризации, как будет показано ниже [12].

Несмотря на то, что в данную модель можно включить любое представление закона скорости

агломерации путем задания соответствующего граничного условия, для упрощения презентации результатов мы будем рассматривать ситуацию, когда присоединение частиц к кластеру термодинамически необратимо и не лимитируется кинетикой этого процесса в рассматриваемых диапазонах времени. Из этого следует, что концентрация свободных частиц на границе двумерного «раствора» равняется нулю:

$$\Gamma|_{\theta=\theta_{\text{clust}}} = 0. \quad (5)$$

Скорость роста кластера определяется величиной потока частиц к его границе. Количество частиц, прибывающих к границе кластера за элементарный интервал времени dt равно:

$$dN = 2\pi R \sin \theta_{\text{clust}}(t) D \left(\frac{\partial \Gamma}{\partial l} \right)_{l_{\text{clust}}} dt, \quad (6)$$

где производная вычисляется по направлению дуги окружности, соответствующей сферической поверхности в осевом сечении (см. рис. 1), а l может быть выражено как $l = R\theta$. При необратимой и диффузионно-контролируемой агломерации это количество соответствует следующей добавленной площади поверхности кластера:

$$dS = \frac{\rho dN}{\Gamma_0} = \frac{2\pi R \rho D}{\Gamma_0} \sin \theta_{\text{clust}}(t) \left(\frac{\partial \Gamma}{\partial l} \right)_{l_{\text{clust}}} dt. \quad (7)$$

С другой стороны, эта добавленная площадь может быть представлена в следующем виде:

$$dS = 2\pi R^2 \sin \theta_{\text{clust}}(t) d\theta_{\text{clust}}. \quad (8)$$

Приравнивая выражения в правых частях (7) и (8), представляя $dl = R d\theta$ и перегруппируя члены, получим следующее уравнение

$$\frac{d\theta_{\text{clust}}}{dt} = \frac{\rho}{\Gamma_0} \frac{D}{R^2} \frac{\partial \Gamma}{\partial \theta} \Big|_{\theta=\theta_{\text{clust}}}, \quad (9)$$

которое определяет скорость роста кластера.

Последнее граничное условие на «южном полюсе» сферы представляет собой условие отсутствия потока вследствие вертикальной осевой симметрии системы:

$$\frac{\partial \Gamma}{\partial \theta} \Big|_{\theta=\pi} = 0. \quad (10)$$

2.1. Безразмерная модель

Введем безразмерные переменные для времени и поверхностной концентрации частиц:

$$\tau = \frac{Dt}{R^2}; \quad \gamma = \frac{\Gamma}{\Gamma_0}, \quad (11)$$

которые позволяют обобщить результаты вычислений на бесконечное количество комбинаций реальных параметров D , t , R и Γ_0 . Уравнение диффузии (3) в новых переменных имеет вид:

$$\frac{\partial \gamma}{\partial \tau} = \frac{\partial^2 \gamma}{\partial \theta^2} + \text{ctg} \theta \frac{\partial \gamma}{\partial \theta} \quad (12)$$

и справедливо для $\tau > 0$ и $\theta_{\text{clust}}(\tau) < \theta < \pi$.

Начальные условия (4) приобретают следующий вид ($\tau = 0$):

$$\gamma(\theta, 0) = 1; \quad (13a)$$

$$\theta_{\text{clust}}(0) = \theta_0. \quad (13b)$$

Граничные условия (5) и (10) сохраняют свой вид в безразмерной форме:

$$\gamma|_{\theta=\theta_{\text{clust}}} = 0, \quad (14a)$$

$$\frac{\partial \gamma}{\partial \theta} \Big|_{\theta=\pi} = 0. \quad (14b)$$

Замыкающее соотношение (9), определяющее скорость роста кластера, преобразуется к следующему виду:

$$\frac{d\theta_{\text{clust}}}{d\tau} = \rho \frac{\partial \gamma}{\partial \theta} \Big|_{\theta=\theta_{\text{clust}}}. \quad (15)$$

3. Численное решение задачи

Описанная задача осложняется тем, что ее область определения изменяется с течением времени из-за увеличения размеров кластера, который ограничивает область сферической поверхности, доступной для диффузии свободных частиц. Данная задача аналогична задаче Стефана о фазовых превращениях [13], а уравнение (15) играет роль условия Стефана, определяющего скорость границы раздела фаз. Таким образом, рассматриваемая задача может быть решена с использованием численных подходов, аналогичных используемым для решения задачи Стефана, так как аналитическое решение поставленной задачи не может быть получено в конечном виде. Используемый здесь метод основан на методе фиксирования фронта [13], то есть использует преобразование координат, переводящее исходную задачу в пространство с фиксированными границами. Второй причиной использования преобразования координат является необходимость обеспечения высокого разрешения численной сетки в критичных зонах поверхности, то есть в диффузионном слое, примыкающем к границе кластера, а также возле «южного полюса» сферы, где коэффициент при первой производной по θ в уравнении (12) имеет особенность. Эта особенность, очевидно, компенсируется нулевой производной концентрации при $\theta = \pi$ (как указано в граничном условии (14b)). Однако котангенс достигает больших значений вблизи этой границы, где производная концентрации может быть отлична от нуля при достаточно больших значениях τ . Это может приводить к значительным численным погрешностям при применении приближенных конечно-разностных вычислений. Следует от-

метить, что данная ситуация аналогична той, что наблюдается при численном решении диффузионно-конвективного уравнения с преобладающим конвективным членом [14].

Основываясь на приведенных выше рассуждениях, введем отображение $\theta \rightarrow \Omega$ угловой координаты из времязависимого отрезка $\theta \in [\theta_{clust}(\tau), \pi]$ в фиксированный отрезок $\Omega \in [0, 1]$, так что граница кластера $\theta = \theta_{clust}(\tau)$ отображается в точку $\Omega = 0$, а «южный полюс» сферы отображается в точку $\Omega = 1$. Исходя из структуры уравнения (12), следующее отображение дает необходимое сжатие/растяжение исходного пространства для преодоления трудностей, описанных выше:

$$\Omega(\theta, \theta_{clust}(\tau)) = \frac{1}{2} \left\{ 1 - \frac{\tan(\alpha)}{\tan \left[\alpha + (\pi - 2\alpha) \frac{\theta - \theta_{clust}(\tau)}{\pi - \theta_{clust}(\tau)} \right]} \right\}, \quad (16)$$

где α – параметр. Следует отметить, что данное преобразование зависит от неизвестного зенитного угла границы кластера, который зависит от времени. Таким образом, новая координата Ω зависит как от θ , так и от τ , хотя ее диапазон изменения $[0, 1]$ остается фиксированным.

Параметр $\alpha \in (0, \pi/2)$ определяет степень сжатия вычислительной сетки в диффузионном слое и вблизи $\theta = \pi$. Рис. 2 демонстрирует влияние параметра α на форму кривой отображения (16) с произвольно выбранным значением $\theta_{clust} = \pi/20$. Когда $\alpha \rightarrow \pi/2$, преобразование приближается к линейному, тогда как при $\alpha \rightarrow 0$ преобразование координат стремится к котангенсу, «сжато» в отрезок $[0, 1]$. Следует заметить, что умеренные значения α позволяют значительно улучшить точность вычислений, как будет показано ниже. Это достигается за счет того, что преобразование (16), показанное на рис. 2 для $\alpha = 0,2$ как улучшает разрешение вычислительной сетки в диффузионном слое возле границы кластера, так и сглаживает особенности котангенса при $\theta = 0$ и $\theta = \pi$, так как является достаточно хорошей аппроксимацией этой функции. Однако слишком малые значения α (то есть $\alpha < 0,1$) приводят к значительным численным ошибкам из-за того, что в этом случае шаг сетки в окрестности $\theta = \pi/2$ в исходном пространстве (то есть в районе «экватора» сферы) становится слишком большим. Таким образом, выбор значения α существенно влияет на погрешности вычислений. В результате проведенных численных экспериментов было установлено, что значение $\alpha = 0,2$ позволяет добиться минимальных погрешностей аппроксимации.

В результате применения преобразования координат (16) основное уравнение (12) приобретает вид:

$$\frac{\partial \gamma}{\partial \tau} = \left(\frac{\partial \Omega}{\partial \theta} \right)^2 \frac{\partial^2 \gamma}{\partial \Omega^2} + \left[\frac{1}{\tan \theta(\Omega)} + \frac{\partial^2 \Omega}{\partial \theta^2} - \frac{\partial \Omega}{\partial \theta_{clust}} \frac{d\theta_{clust}}{d\tau} \right] \frac{\partial \gamma}{\partial \Omega}. \quad (17)$$

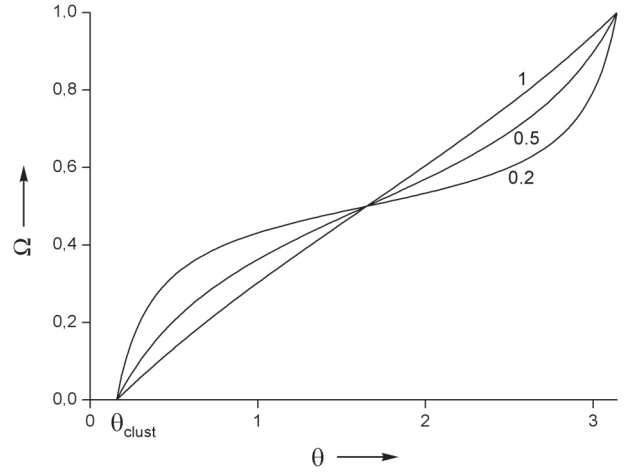


Рис. 2. Влияние параметра α на форму отображения реального пространства θ в Ω ; $\alpha = 1; 0,5; 0,2$

Последнее слагаемое в квадратных скобках в (17) отражает тот факт, что фиксированная вычислительная область в пространстве Ω соответствует постепенно уменьшающейся области в пространстве θ . Это слагаемое, таким образом, играет роль искусственной конвекции частиц, создаваемой за счет сжатия области, доступной для диффузии, в результате роста кластера. Выражения для нелинейных коэффициентов в уравнении (17) не приводятся здесь в силу их громоздкости. Однако они могут быть легко получены дифференцированием преобразования координат (16). Следует отметить также нелинейную зависимость от положения границы кластера $\theta_{clust}(\tau)$.

При использовании преобразования координат и зависимых переменных граничные условия принимают следующий вид:

$$\gamma|_{\Omega=0} = 0, \quad (18a)$$

$$\frac{\partial \gamma}{\partial \Omega} \Big|_{\Omega=1} = 0, \quad (18b)$$

а времязависимое положение границы кластера $\theta_{clust}(\tau)$ удовлетворяет следующему уравнению:

$$\frac{d\theta_{clust}}{d\tau} = \frac{\rho}{\sin 2\alpha} \frac{\pi - 2\alpha}{\pi - \theta_{clust}(\tau)} \frac{\partial \gamma}{\partial \Omega} \Big|_{\Omega=0}. \quad (19)$$

Таким образом, исходная линейная задача в области с движущейся границей была преобразована в нелинейную задачу (17)-(19) в фиксированной области, которая может быть решена с использованием более простых численных подходов при достижении большей точности. Данная задача была решена с использованием чисто неявной схемы конечных разностей на равномерной сетке по

переменной Ω . Вследствие нелинейности задачи (17)-(19) распределение концентрации и положение границы кластера на каждом шаге по времени находились с использованием итерационного обобщенного метода Ньютона.

3.1. Сходимость

Задача (17)-(19) представляет собой краевую задачу для нелинейной системы обыкновенного дифференциального уравнения и уравнения в частных производных, для которой невозможно провести общий анализ погрешностей для используемой конечно-разностной схемы. Поэтому согласованность и сходимость результатов вычислений проверялась путем проведения численного эксперимента.

Несмотря на то, что уравнение (19) по построению формально удовлетворяет закону сохранения вещества, численное решение неизбежно нарушает этот закон вследствие накопления ошибок аппроксимации (особенно в областях с большими значениями производной концентрации) и округления. Согласованность и сходимость используемого численного метода следуют из результатов, представленных на рис. 3, где показаны зависимости относительной ошибки в количестве кластеризованных частиц, представленном величиной $\theta_{\text{clust}}(\tau)$, и в общем количестве частиц на поверхности сферы от количества $N\Omega$ точек сетки по пространственной переменной и от параметра α при фиксированном разбиении по времени ($N\tau = 10000$).

Результаты, показанные на рис. 3, демонстрируют высокую скорость сходимости по $N\Omega$ для всех трех рассмотренных значений α (0,05, 0,2 и 1,2). При $\alpha = 0,2$ скорость сходимости достигает почти второго порядка (то есть относительная ошибка пропорциональна $N\Omega^{-1.9} \approx N\Omega^{-2}$). Также из данных результатов следует, что при этом значении параметра сжатия сетки достигается наивысшая скорость сходимости и наименьшие абсолютные значения ошибки для всех значений $N\Omega$, что подтверждает приведенный выше качественный анализ. Численный эксперимент также показал (данные здесь не приведены), что сходимость по $N\tau$ (количество разбиений по переменной τ) имеет первый порядок, а абсолютные значения ошибки оказываются наименьшими для $\alpha = 0,2$. Таким образом, для достижения точности 0,1% как в вычисленном положении границы кластера, так и в законе сохранения вещества необходимо использовать значение параметра $\alpha = 0,2$ и численную сетку размером $N\Omega \times N\tau = 160 \times 1300$.

4. Результаты

Поведение рассмотренной системы во времени может быть описано серией зависимостей относительного количества кластеризованных частиц от τ и ρ :

$$\frac{N(\tau)}{N_0} = \frac{1 - \cos(\theta_{\text{clust}}(\tau))}{1 + \rho - (1 - \rho)\cos\theta_0}. \quad (20)$$

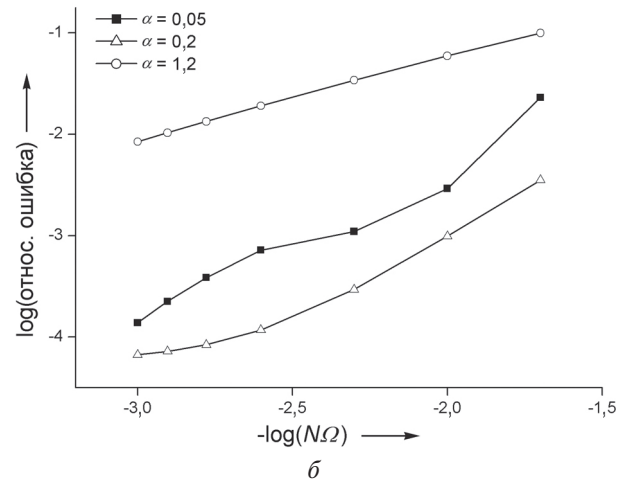
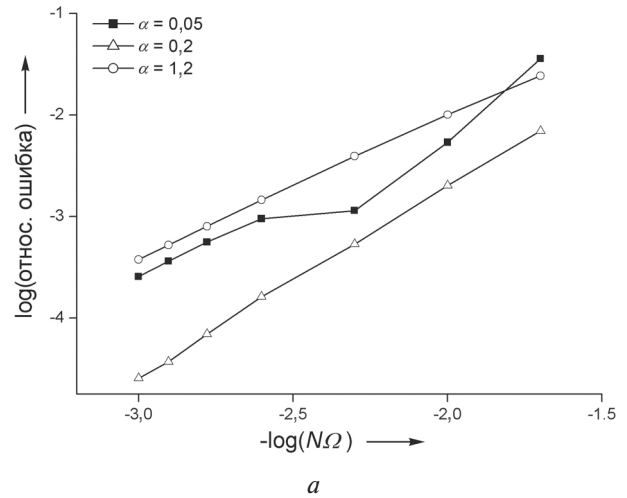


Рис. 3. (а) Сходимость θ_{clust} при изменении разбиения по Ω в точке $\tau = 0,72$ для трех значений α . (б) Сходимость общего количества вещества (количество частиц по сравнению с начальным количеством N_0) в точке $\tau = 0,72$ для трех значений α . Другие параметры имели следующие значения: $\rho = 0,8$, $\theta_0 = \pi/500$, $N\tau = 10000$.

На рис. 4 показаны кривые $N(\tau)/N_0$, вычисленные для значений ρ из диапазона 0,05...0,8 (слева направо). Очевидно, что чем меньше разница между концентрациями свободных и кластеризованных частиц (то есть чем выше значение ρ), тем быстрее растет кластер и, соответственно, тем быстрее кривая $N(\tau)/N_0$ достигает своего максимального значения, равного 1.

На рис. 5 показаны зависимости зенитного угла $\theta_{\text{clust}}(\tau)$ от времени, соответствующие кривым $N(\tau)/N_0$ на рис. 4. Как видно из рис. 5, $\theta_{\text{clust}}(\tau)$ достигает своего предельного значения

$$\theta_{\text{clust}}^{\text{max}} = \arccos[(1 - \rho)\cos\theta_0 - \rho], \quad (21)$$

когда кластер содержит все частицы, которые изначально были равномерно распределены по поверхности сферы.

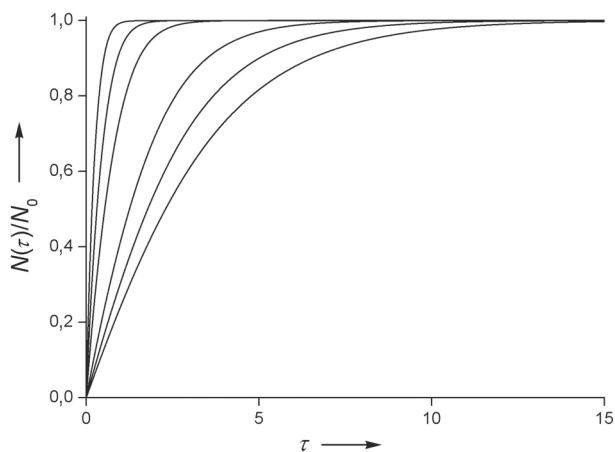


Рис. 4. Зависимость развития величины $N(\tau)/N_0$ во времени от ρ . Кривые соответствуют значениям ρ (слева направо): 0,8; 0,67; 0,5; 0,2; 0,1; 0,05.

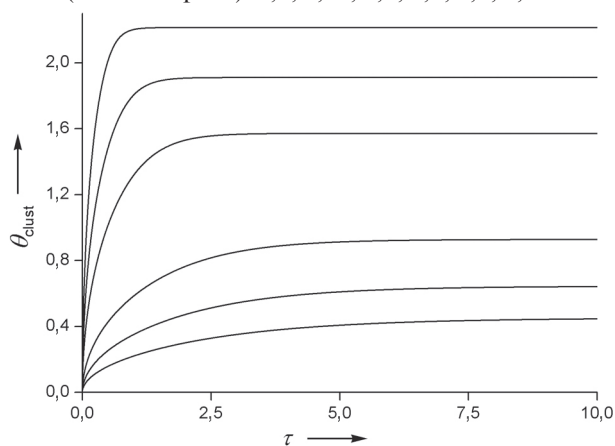


Рис. 5. Зависимость развития величины $\theta_{clust}(\tau)$ с течением времени от ρ . Кривые соответствуют значениям ρ (сверху вниз): 0,8; 0,67; 0,5; 0,2; 0,1; 0,05

Все приведенные вычисления проводились с начальным углом $\theta_0 = \pi/500$, что формально устанавливает ненулевое начальное условие для $\theta_{clust}(\tau)$ (см. выше). Однако численные эксперименты показали, что результирующие зависимости $N(\tau)/N_0$ и $\theta_{clust}(\tau)$ от времени при условии, что $\theta_0 \ll \theta_{clust}^{max}$, кроме чрезвычайно малых времен, которые не представляют реального интереса в данном контексте.

Следует заметить, что хотя рассмотренная модель позволяет проводить вычисления для любых значений параметра ρ из интервала $(0, 1)$, полученные результаты не будут соответствовать реальному физическому поведению при больших значениях ρ . В действительности, последнее означает, что при кластеризации частицы занимают почти столько же места, как и в свободном состоянии, то есть их размерами нельзя пренебрегать. В этом случае законы диффузии Фика более неприменимы, и возникает необходимость применения других подходов. В работе [15] было установлено, что предложенная здесь модель позволяет получать физически значимые результаты при условии, что $\rho < 0,1$.

Область применения предложенной модели может быть расширена на системы с неподвижными активными границами, как, например, кольцевой активный интерфейс между деформированной сферической молекулой-дендримером и электродом [11]. В этом случае для фиксирования границы θ_0 должно быть установлено в некоторое конечное значение, а параметр ρ установлен равным нулю. Скорость перемещения активной границы в данной ситуации равна нулю в соответствии с уравнением (15).

Выводы

Рассмотренная в данной работе модель описывает замкнутую диффузионную систему, которая встречается в различных практических ситуациях в физической химии и биологии, где описанные процессы происходят в наномасштабах. Было показано, что динамика диффузии и агломерации частиц на сферической поверхности может быть описана законами Фика при условии, что размерами частиц можно пренебречь, то есть когда параметр ρ не превышает 0,1, что соответствует покрытию частицами не более 10% сферической поверхности. Данное ограничение, однако, не уменьшает ценности данной модели и полученных результатов для практических применений, где концентрации обычно находятся в указанных пределах. Было показано, что предложенная модель охватывает также случай фиксированной кольцевой активной границы, расположенной на поверхности сферы, что значительно расширяет круг возможных применений.

Список литературы: 1. Ertl, G. Reactions at Surfaces: From Atoms to Complexity (Nobel Lecture) [Текст] / G. Ertl // Angew. Chem. Int. Ed. – 2008. – Vol. 47. – P. 3524-3535. 2. Somorjai, G.A. Evolution of the surface science of catalysis from single crystals to metal nanoparticles under pressure [Текст] / G.A. Somorjai, J.Y. Park // J. Chem. Phys. – 2008. – Vol. 128. – Article No. 182504. 3. Attard, G.A. The characterisation of supported platinum nanoparticles on carbon used for enantioselective hydrogenation: a combined electrochemical-stm approach [Текст] / G.A. Attard, A. Ahmadi, D.J. Jenkins, O.A. Hazzazi, P.B. Wells, K.G. Griffin, P. Johnston, J.E. Gillies // ChemPhysChem – 2003. – Vol. 4. – P. 123-130. 4. Murray, R.W. Nanoelectrochemistry: metal nanoparticles, nanoelectrodes, and nanopores [Текст] / R.W. Murray // Chem. Rev. – 2008. – Vol. 108. – P. 2688-2720. 5. Burgoyne, R.D. Secretory granule exocytosis [Текст] / R.D. Burgoyne, A. Morgan // Physiol. Rev. – 2003. – Vol. 83. P. 581-632. 6. Co, C. Mechanism of actin network attachment to moving membranes: barbed end capture by n-wasp wh2 domains [Текст] / C. Co, D.T. Wong, S. Gierke, V. Chang, J. Taunton // Cell – 2007. – Vol. 128. – P. 901-913. 7. Akin, O. Capping protein increases the rate of actin-based motility by promoting filament nucleation by the arp2/3 complex [Текст] / O. Akin, R.D. Mullins // Cell – 2008. – Vol. 133. – P. 841-851. 8. Revenu, C. The co-workers of actin filaments: from cell structures to signals [Текст] / C. Revenu, R. Athman,

S. Robine, D. Louvard // Nature Reviews Molecular Cell Biology – 2004. – Vol. 5. – P. 635-646. 9. Bannai, H. Bioconjugated quantum dots for multiplexed and quantitative immunohistochemistry [Текст] / H. Bannai, S. Lyyvi, C. Schweizer, M. Dahan, A. Triller // Nature Protocols – 2007. – Vol. 1. – P. 2628–2634. 10. Amatore, C. Theory of long-range diffusion of proteins on a spherical biological membrane: application to protein cluster formation and actin-comet tail growth [Текст] / C. Amatore, A.I. Oleinick, O.V. Klymenko, I. Svir // ChemPhysChem – 2009. – Vol. 10. – P. 1586-1592. 11. Amatore, C. Ultrafast voltammetry of adsorbed redox active dendrimers with nanometric resolution: an electrochemical microtome [Текст] / C. Amatore, Y. Bouret, E. Maisonhaute, J.I. Goldsmith, H.D. Abruca // Chem. Eur. J. – 2001. – Vol. 7. – P. 2206-2226. 12. Amatore, C. Diffusion within nanometric and micrometric spherical-type domains limited by nanometric ring or pore active interfaces. Part 1. Conformal mapping approach [Текст] / C. Amatore, A. Oleinick, I. Svir // J. Electroanal. Chem. – 2005. – Vol. 575. – P. 103-123. 13. Самарский, А.А. Вычислительная теплопередача [Текст] / А.А. Самарский, П.Н. Вабищевич. – М.: Editorial URSS, 2003. – 784 с. 14. Morton, K.W. Numerical solution of convection-diffusion problems [Текст] / K.W. Morton. – Chapman&Hall, 1996 – 372 p. 15. Amatore, C. Diffusion with moving boundary on spherical surfaces [Текст] / C. Amatore, O.V. Klymenko, A.I. Oleinick, I. Svir // ChemPhysChem – 2009. – Vol. 10. – P. 1586-1592.

Поступила в редколлегию 20.10.2009

УДК 57:51-76; 57.02.001.57; 517.958:57

Моделювання дифузії з фазовим переходом на сферичній поверхні / О.В. Клименко // Біоніка інтелекту: наук.-техн. журнал. – 2009. – № 2 (75). – С. 99-105.

У даній статті представлено результати математичного моделювання дифузії реагуючих частинок, що пов'язана з формуванням кластера на сферичній поверхні. Математична модель базується на законах Фіка. Чисельне розв'язання задачі дозволило отримати залежності розміру кластера від часу та початкової концентрації частинок, а також встановити діапазони безрозмірних параметрів, у яких побудована модель є застосовною до моделювання реальних систем.

Іл. 5. Бібліогр.: 15 найм.

UDK 57:51-76; 57.02.001.57; 517.958:57

Modelling diffusion coupled with phase transition on a spherical surface / A.V. Klymenko // Bionics of Intelligence: Sci. Mag. – 2009. – № 2 (71). – P. 99-105.

This work presents the results of mathematical modelling of diffusion on a spherical surface coupled with the formation of a cluster by reactive particles. The mathematical model is based on the Fick's laws. Numerical solution of the problem allowed obtaining dependencies of the cluster size on time and initial concentration of particles. Ranges of dimensionless parameters where the developed model is applicable to real systems were determined based on simulation results.

Fig. 5. Ref.: 15 items.

УДК 57:51-76; 57.02.001.57; 517.958:57



МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ ДИФФУЗИИ НЕЙРОТРАНСМИТТЕРА ПРИ ФИКСИРОВАННЫХ УГЛАХ ОТКРЫТИЯ ВЕЗИКУЛЫ

А.И. Олейник

ХНУРЭ, г. Харьков, Украина, oleinick@kture.kharkov.ua

В статье решена задача диффузионного массопереноса нейротрансмиттера при везикулярном экзоцитозе. Предложена новая математическая модель для фиксированного угла открытия везикулы. Численное решение задачи было получено при помощи построенного квазиконформного отображения.

ВЕЗИКУЛА, ДИФФУЗИЯ, МОДЕЛИРОВАНИЕ, КВАЗИКОНФОРМНОЕ ОТОБРАЖЕНИЕ, ИСКУССТВЕННЫЙ СИНАПС

Введение

Диффузия биологически активных веществ (нейротрансмиттеров, гормонов, протеинов) из везикул является чрезвычайно важным процессом и встречается в таких жизненно важных явлениях, как передача нервного возбуждения в синапсах, соединяющих нейроны или секреция гормонов; внутриклеточных процессах, связанных с доставкой активных веществ везикулой в необходимые участки клетки и других [1-4]. Патологии, связанные с поведением везикулы во время диффузии содержащегося в ней вещества, приводят к различным заболеваниям, например, болезни Альцгеймера, нарушению выделения адреналина и так далее.

Для изучения, а также диагностики этих процессов, как правило, используются спектроскопические и/или электрохимические методы [3, 4]. Последние являются особенно эффективными в случае использования для измерений микро- и наноэлектродов, так как позволяют получать точную количественную информацию о потоке активных веществ и его динамике [3, 4]. Однако упомянутая информация с достаточной точностью может быть получена из экспериментальных данных только при наличии соответствующей теории.

Целью данной работы являлось построение математической модели процесса диффузионного массопереноса вещества из везикулы при ее мгновенном раскрытии на фиксированный угол, а также численное решение поставленной задачи.

1. Математическая модель

Везикулярный экзоцитоз происходит по одной и той же схеме независимо от типа клеток, в которых совершается эмиссия того или иного вещества, хотя молекулярные механизмы, приводящие к экзоцитозу, как правило, существенно отличаются [1, 2]. Исходным состоянием везикулы является ее свободное расположение в цитоплазме, с последующим креплением к мембране клетки при помощи SNARE протеинов [5]. При получении сигнала, передаваемого, как правило, при помощи активации соответствующего протеина (синаптоагмин в нейронах), мембрана везикулы сливается с мембраной клетки с образованием поры (рис. 1).

Вещество, содержащееся в везикуле, начинает диффундировать через пору во внеклеточное пространство. В случае нейронов нейротрансмиттер диффундирует через синаптическую щель и достигает рецепторов, расположенных в пост-синаптической области и реагирующих на его концентрацию. Радиус поры, сразу после ее образования, составляет порядка нескольких ангстрем и увеличивается с течением времени.

Для изучения данного процесса нейрон или секреторная клетка изолируются, и в непосредственной близости к мембране клетки помещается микроэлектрод. Потенциал последнего устанавливается на плато окислительно-восстановительной волны детектируемого вещества содержащегося в везикуле.

Диффундируя из везикулы через пору, вещество под действием градиента концентрации и в силу того, что близость электрода ограничивает пространство доступное для диффузии, попадает в окрестность поверхности электрода, где протекает реакция переноса электрона (РПЭ). Регистрируемый на микроэлектроде ток характеризует концентрацию вещества и интенсивность его потока, а, следовательно, степень раскрытия везикулы. Описанный экспериментальный метод называется методом искусственного синапса, так как по характерному расположению элементов системы напоминает нейронный синапс [3, 4].

Следует особо отметить тот факт, что внутренность везикулы заполнена полиэлектролитным гелем, в котором содержится соответствующее биологически активное вещество. Коэффициент диффузии вещества в полиэлектролитном геле (D), как правило, на два-три порядка меньше соответствующего коэффициента во внеклеточной жидкости (D_{out}). Когда радиус везикулы (r_{ves}) сравним с расстоянием между мембраной и поверхностью электрода (l), что имеет место в эксперименте, характеристическое время диффузии внутри везикулы $\tau_1 = r_{ves}^2 / D$ будет намного большим характеристического времени диффузии вне везикулы $\tau_2 = l^2 / D_{out}$. Поэтому условие равенства нулю

концентрации на поверхности электрода (так как потенциал электрода соответствует плато окислительно-восстановительной волны) можно перенести на открытую часть везикулы. Следовательно, при описанных выше условиях для моделирования «отклика», регистрируемого микроэлектродом, достаточно рассмотреть диффузию вещества внутри везикулы с условиями равенства концентрации вещества нулю на открытой поверхности везикулы (сферическая форма границы будет сохраняться благодаря полиэлектролитному гелю) и равенства нулю потока вещества на остальной части границы. Также будем считать в дальнейшем, что раскрытие везикулы на угол α_{op} (рис. 1) происходит в момент времени $t = 0$ и угол открытия в дальнейшем не изменяется.

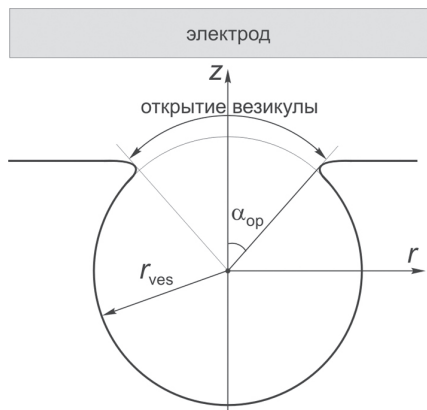


Рис. 1. Схематическое изображение открытой везикулы и расположения электрода

Используя осевую симметрию системы, запишем математическую модель в цилиндрических координатах. Для удобства вычислений, а также для того, чтобы полученные результаты имели большую общность, введем следующие безразмерные переменные:

$$R = \frac{r}{r_{ves}}, \quad Z = \frac{z}{r_{ves}}, \quad \tau = \frac{Dt}{r_{ves}^2}, \quad C = \frac{c}{c_0}, \quad (1)$$

где c_0 — исходная концентрация вещества в закрытой везикуле.

Таким образом, уравнение диффузионного массопереноса вещества будет иметь следующий вид:

$$\frac{\partial C}{\partial \tau} = \frac{\partial^2 C}{\partial R^2} + \frac{1}{R} \frac{\partial C}{\partial R} + \frac{\partial^2 C}{\partial Z^2}. \quad (2)$$

Начальные и граничные условия:

$$\tau = 0: \quad \forall R, Z: \quad R^2 + Z^2 \leq 1, \quad C = 1; \quad (3a)$$

$$\tau > 0: \quad R = 0, \quad -1 \leq Z \leq 1, \quad \frac{\partial C}{\partial R} = 0; \quad (3b)$$

$$R^2 + Z^2 = 1, \quad \alpha \leq \alpha_{op}, \quad C = 0; \quad (3в)$$

$$R^2 + Z^2 = 1, \quad \alpha > \alpha_{op}, \quad \frac{\partial C}{\partial \vec{\nu}} = 0, \quad (3г)$$

где $\vec{\nu}$ — вектор единичной внутренней нормали.

Ток на микроэлектроде определяется интегралом от градиента концентрации по его поверхности [6]. Однако, учитывая вышесказанное, интеграл может быть записан по открытой поверхности везикулы (S_{op}):

$$i(t) = nFD \int_{S_{op}} \frac{\partial c}{\partial \vec{\nu}} dS = nFDc_0 2\pi r_{ves} \int_{L_{op}} \frac{\partial C}{\partial R} R \sin \alpha dL, \quad (4)$$

где n — количество электронов участвующих в РПЭ от одной молекулы биологически активного вещества; F — константа Фарадея; L_{op} — дуга, соответствующая открытой части границы (рис. 1).

2. Квазиконформное отображение

Область моделирования является полукруг с чередующимися на полуокружности граничными условиями Дирихле и Неймана (см. уравнения (3)). В точке разрыва граничных условий поле концентраций будет иметь особенность, обусловленную сильными градиентами концентрации в окрестности этой точки. Эта особенность не проявляется лишь при очень малых временах, когда наблюдается планарная диффузия в области открытия везикулы. Указанное резкое изменение поля концентрации в окрестности точки, где меняется граничное условие, плохо поддается численному описанию и является источником ошибок. Поэтому для численного решения данной задачи необходимо использование неравномерных или адаптивных сеток. Для получения подходящей неравномерной сетки использовалось преобразование Шварца-Кристоффеля для круга [7]:

$$\omega^* = g(X) = K \int_0^X \prod_{i=1}^m \left(1 - \frac{\zeta}{X_i} \right)^{\alpha_i - 1} d\zeta, \quad (5)$$

где $X = R + iZ$; $\omega^* = \xi^* + i\eta^*$; X_i — координаты вершин, расположенных на окружности (рис. 2а); α_i — угол многоугольника в вершине ω_i ($\omega_i = g(X_i)$); K — масштабный множитель; ζ — переменная интегрирования. В нашем случае $m = 4$ и все углы прямые.

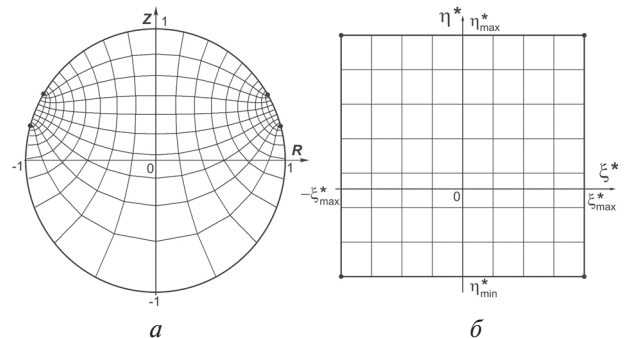


Рис. 2. Область моделирования в реальных и преобразованных координатах

В силу соображений симметрии будем рассматривать только правую половину области моделиро-

вания. Преобразование (5) конформно отображает правый полукруг единичного радиуса на прямоугольник $[0; \xi_{\max}^*] \times [\eta_{\min}^*; \eta_{\max}^*]$ (рис. 2б), причем начало координат остается неподвижным.

Для удобства полученный прямоугольник масштабировался в единичный квадрат путем двух последовательных преобразований:

$$\tilde{\omega} = \tilde{\xi} + i\tilde{\eta} = \frac{1}{\xi_{\max}^*} [\xi^* + i\eta^*] \quad (6a)$$

в область моделирования $[0; 1] \times [\tilde{\eta}_{\min}; \tilde{\eta}_{\max}]$ и посредством

$$\omega' = \xi' + i\eta' = \tilde{\xi} + i \frac{\tilde{\eta} - \tilde{\eta}_{\min}}{\tilde{\eta}_{\max} - \tilde{\eta}_{\min}} \quad (6б)$$

в область $[0; 1] \times [0; 1]$.

На рис. 3а изображена сетка в реальных координатах, соответствующая равномерной прямоугольной сетке в пространстве ω' для угла открытия $\alpha_{op} = 120^\circ$. Большой угол открытия везикулы был выбран, чтобы продемонстрировать, что с увеличением угла открытия шаг сетки возле активной поверхности увеличивается. Соответственно, точность расчета потока вещества через эту поверхность ухудшается. Поэтому для углов открытия $\alpha_{op} > \pi/2$ применялась дополнительная функция сжатия вдоль координаты η' , задаваемая кубическим сплайном Безье [8] в параметрической форме:

$$B_3(t) = P_0(1-t)^3 + 3P_1t(1-t)^2 + 3P_2t^2(1-t) + P_3t^3, \quad (7a)$$

$$\omega = \xi + i\eta = \xi' + iB_3[t(\eta')], \quad (7б)$$

где $P_i = (\eta'; \eta)^T$ – векторы координат точек, задающих сплайн. Типичные значения точек P_i для углов больших $\pi/3$ были следующими: $P_0 = (0; 0)^T$, $P_1 = (0,5; 0,75)^T$, $P_2 = (0,5; 0,95)^T$, $P_3 = (1; 1)^T$.

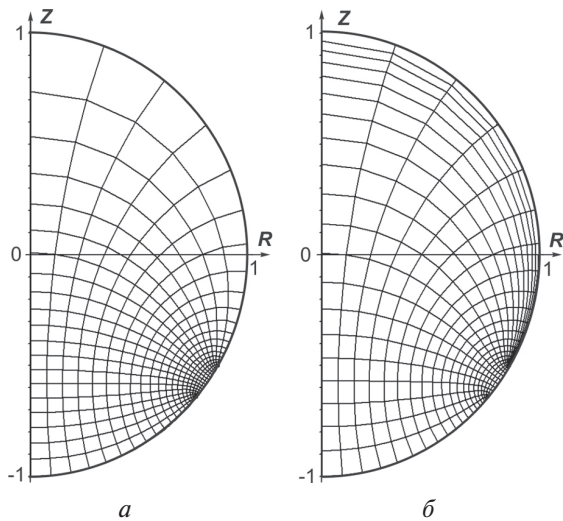


Рис. 3. Вычислительная сетка до (а) и после (б) применения дополнительного преобразования (6)

Сетка, изображенная на рис. 3б, соответствует указанному набору точек P_i . При этом четко вид-

но, что разрешение сетки в окрестности активной поверхности значительно улучшилось. Для малых углов открытия везикулы, то есть меньших $\pi/3$, использовался сплайн (7а), задаваемый точками $P_0 = (0; 0)^T$, $P_1 = (0,3; 0,05)^T$, $P_2 = (0,4; 0,4)^T$, $P_3 = (1; 1)^T$.

Следует также отметить, что результирующее отображение (7б) является квазиконформным [9] (равно, как и промежуточное (6)) в силу дополнительного преобразования мнимой части.

3. Математическая модель в преобразованных координатах

Математическая модель в преобразованных координатах имеет следующий вид:

$$\begin{aligned} \frac{\partial C}{\partial \tau} = \det J^* & \left[\frac{1}{\xi_{\max}^*} \frac{\partial^2 C}{\partial \xi^2} + \right. \\ & \left. + \frac{1}{(\tilde{\eta}_{\max} - \tilde{\eta}_{\min})^2} \left\{ \frac{1}{f'^2(\eta)} \frac{\partial^2 C}{\partial \eta^2} - \frac{f''(\eta)}{[f'(\eta)]^3} \frac{\partial C}{\partial \eta} \right\} \right] + \quad (8) \\ & + \frac{1}{R(\xi, \eta)} \left[\frac{1}{\xi_{\max}^*} \frac{\partial C}{\partial \xi} \frac{\partial \xi^*}{\partial R} + \frac{1}{(\tilde{\eta}_{\max} - \tilde{\eta}_{\min}) f'(\eta)} \frac{\partial C}{\partial \eta} \frac{\partial \eta^*}{\partial R} \right], \end{aligned}$$

где для удобства записи мы обозначили

$$\eta' = f(\eta) = B_3^{-1}[t(\eta)],$$

а якобиан преобразования (5) и его частные производные вычислялись как:

$$\det J^* = |g'(X)|^2, \quad (9a)$$

$$\frac{\partial \xi^*}{\partial R} = \frac{\partial \eta^*}{\partial Z} = \text{Re}[g'(X)], \quad (9б)$$

$$-\frac{\partial \xi^*}{\partial Z} = \frac{\partial \eta^*}{\partial R} = \text{Im}[g'(X)]. \quad (9в)$$

Начальные и граничные условия имеют вид:

$$\tau = 0: \quad 0 \leq \xi \leq 1, \quad 0 \leq \eta \leq 1, \quad C = 1; \quad (10a)$$

$$\tau > 0: \quad \xi = 0, \quad 0 \leq \eta \leq 1, \quad \frac{\partial C}{\partial \xi} = 0; \quad (10б)$$

$$0 \leq \xi \leq 1, \quad \eta = 0, \quad \frac{\partial C}{\partial \eta} = 0; \quad (10в)$$

$$\xi = 1, \quad 0 \leq \eta \leq 1, \quad \frac{\partial C}{\partial \xi} = 0; \quad (10г)$$

$$0 \leq \xi \leq 1, \quad \eta = 1, \quad C = 0. \quad (10д)$$

Выражение для тока, регистрируемого микроэлектродом, (4) в преобразованных координатах будет иметь вид:

$$i(t) = 2\pi nFDc_0 r_{ves} \frac{\xi_{\max}^*}{(\eta_{\max}^* - \eta_{\min}^*) f'(1)} \int_0^1 \frac{\partial C}{\partial \eta} R(\xi, \eta) d\xi. \quad (11)$$

4. Результаты и обсуждение

Проверка правильности численных результатов была проведена при помощи сравнения их с некоторыми предельными случаями и известными аналитическими результатами.

Так, рассмотренная система является эквивалентной электрохимической системе с поверхностью электрода, соответствующей открытию везикулы, и РПЭ, протекающей на этой поверхности в условиях хроноамперометрического эксперимента. Поэтому в окрестности точки $t = 0$ наблюдается планарная диффузия и отклик системы описывается уравнением Коттреля [6]:

$$i(t) = nFS c_0 \sqrt{\frac{D}{\pi t}} = nF2\pi r_{ves}^2 (1 - \cos \alpha_{op}) c_0 \sqrt{\frac{D}{\pi t}} \quad (12)$$

где $S = 2\pi r_{ves}^2 (1 - \cos \alpha_{op})$ – площадь сферического сегмента, открытого во внеклеточное пространство.

Известно, что нагретые тела конечных размеров после некоторого начального периода остывают по экспоненциальному закону [10]. В силу эквивалентности уравнений теплопроводности и диффузии, аналогичное поведение должно наблюдаться и для рассмотренной системы, то есть в заключительной фазе эксперимента (когда значительная часть содержащегося в везикуле вещества уже эмитирована) ток на электроде будет убывать экспоненциально.

Другим предельным случаем является ситуация полного открытия везикулы, то есть когда $\alpha_{op} = \pi$ и математическая модель может быть записана с использованием одного пространственного измерения. Решение этой (нестационарной) задачи известно в аналитическом виде [11] и модифицированное для нашей конкретной ситуации запишется в следующем виде:

$$i(t) = -nFD4\pi r_{ves}^2 \left. \frac{\partial c}{\partial r} \right|_{r=r_{ves}} = nFDc_0 8\pi r_{ves} \sum_{k=1}^{\infty} \exp\left[-\frac{\pi^2 k^2 Dt}{r_{ves}^2}\right] \quad (13)$$

На рис. 4 приведены вычисленные зависимости тока от времени в двойном логарифмическом масштабе для трех характерных ситуаций, когда угол открытия составляет $\pi/6$, $\pi/2$ и $5\pi/6$ (кривые 1-3 соответственно). Как и ожидалось, амплитуда тока при малых временах тем больше, чем больше угол открытия, так как ток при этих условиях пропорционален площади открытия везикулы. Серыми штриховыми линиями показаны коттрелевские токи, полученные из уравнения (12). В двойном логарифмическом масштабе данные зависимости представляют собой прямые с угловым коэффициентом равным $(-1/2)$, так как $i(t) \sim 1/\sqrt{t}$. Вычисленные токи хорошо согласуются с предельными коттрелевскими токами.

В силу того, что везикула содержит конечное количество вещества, большие токи в начале эксперимента компенсируются более резким спадом с увеличением времени. Штрихпунктирной ли-

нией (кривая 4 на рис. 4) показано аналитическое решение (13) для полного открытия везикулы, которая демонстрирует правильность поведения численных результатов с увеличением угла открытия везикулы. Характерные плато, которые видны на рис. 4 (кривые 2 и 3), обусловлены конечной точностью представления вещественных чисел, которая составляла 16-17 значащих цифр.

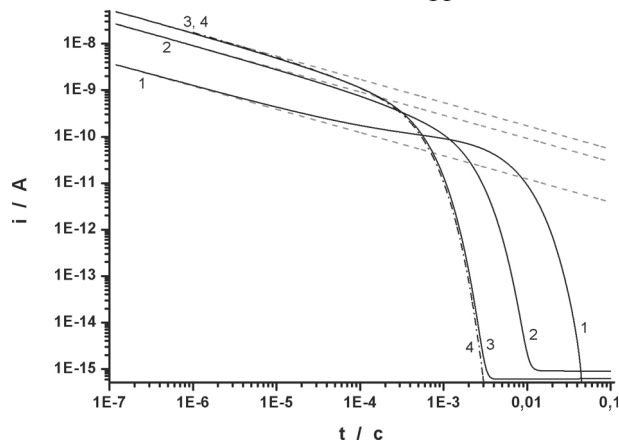


Рис. 4. Зависимость тока от времени в двойном логарифмическом масштабе:
 1 – $\alpha_{op} = \pi/6$; 2 – $\alpha_{op} = \pi/2$; 3 – $\alpha_{op} = 5\pi/6$;
 4 – $\alpha_{op} = \pi$. Штриховые линии – коттрелевские токи (уравнение (12)), штрихпунктирная – ток, вычисленный по уравнению (13)

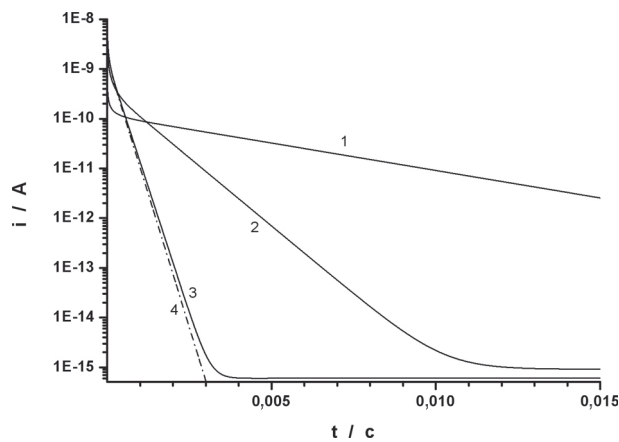


Рис. 5. Зависимость тока от времени в логарифмическом масштабе по току:
 1 – $\alpha_{op} = \pi/6$; 2 – $\alpha_{op} = \pi/2$;
 3 – $\alpha_{op} = 5\pi/6$; 4 – $\alpha_{op} = \pi$

Рис. 5 содержит те же численные результаты, что представлены на рис. 4 вместе с аналитическим решением (13), но с логарифмическим масштабом только по оси токов. Как было сказано выше, после прохождения начальной фазы эксперимента, токи убывают экспоненциально, о чем и свидетельствуют прямолинейные участки (в логарифмическом масштабе) показанных зависимостей на рис. 5.

Численные результаты также должны удовлетворять закону сохранения заряда. В силу того, что общее количество заряда, содержащееся в везикуле, может быть вычислено по формуле

$Q = (4/3)\pi r_{ves}^3 n F c_0$, а также дается интегралом тока [6]

$$Q = \int_0^{T_e} i(t) dt, \quad (14)$$

где T_e — общая длительность эксперимента, в течение которой везикула полностью опустошается. Результаты проверки показали, что закон сохранения выполняется и относительная ошибка в значении количества заряда, найденного по вычисленным токам с использованием уравнения (14), не превышает 0,25%.

Выводы

В работе предложена новая математическая модель диффузионного переноса биологически активного вещества из везикулы при ее открытии на фиксированный угол. Для численного решения данной задачи был предложен гибкий подход, основанный на применении квазиконформного отображения, позволяющего получать точные результаты для произвольного угла открытия везикулы. Сравнение полученных численных результатов с предельными случаями и аналитическими решениями подтверждают высокую точность результатов. Полученные результаты являются важными и необходимыми для решения следующей, более сложной задачи, а именно, когда открытие везикулы является динамическим процессом, зависящим от времени.

Список литературы: 1. Lehninger, A. Principles of Biochemistry [Текст] / A. Lehninger. — W.H. Freeman & Co, 2004. — 1110 p. 2. Люин, Б. Гены [Текст] / Б. Люин. — М.: Мир, 1987. — 544 с. 3. Amatore, C. Ultramicroelectrodes: Their Basic Properties and their Use in Semi-Artificial Synapses [Текст] / C. Amatore // C.R. Acad. Sci. Ser. Iib. — 1996. — V. 323. — P. 757-771. 4. Amatore, C. Time-resolved dynamics of the vesicle membrane during individual exocytotic secretion events, as extracted from amperometric monitoring of adrenaline exocytosis from chromaffin cells [Текст] / C. Amatore, Y. Bouret, L. Midrier // Chem. Eur. J. — 1999. — V. 5(7). — P. 2151 — 2162. 5. Zenisek, D. Transport, capture and exocytosis of single synaptic vesicles at active zones [Текст] / D. Zenisek, J. A. Steyer, W. Almers // Nature. — 2000. — V. 406. — P. 849 — 854. 6. Bard, A. Electrochemical methods: Foundations and Applications [Текст] / A. Bard, L. Faulkner. — John Wiley &

Sons, Inc., 2001. — 834 p. 7. Driscoll, T.A. Schwarz-Christoffel mapping [Текст] / T.A. Driscoll, L.N. Trefethen. — Cambridge University Press, 2002. — 132 p. 8. Хилл, Ф. OpenGL. Программирование компьютерной графики [Текст] / Ф. Хилл. — СПб.: Питер, 2002. — 1088 с. 9. Лаврентьев, М.А. Методы теории функций комплексного переменного [Текст] / М.А. Лаврентьев, Б.В. Шабат. — М.: Наука, 1973. — 736 с. 10. Тихонов, А.Н. Уравнения математической физики [Текст] / А.Н. Тихонов, А.А. Самарский — М.: Наука, 1966. — 724 с. 11. Полянин, А.Д. Справочник по точным решениям уравнений тепло- и массопереноса [Текст] / А.Д. Полянин, А.В. Вязьмин, А.И. Журов, Д.А. Казенин. — М.: Факториал, 1998. — 368с.

Надійшла до редколегії 21.10.2009

УДК 57:51-76; 57.02.001.57; 517.958:57

Математичне моделювання дифузії нейротрансмітера у разі фіксованих кутів відкриття везикули / О.І. Олійник // Біоніка інтелекту: наук.-техн. журнал. — 2009. — № 2 (71). — С. 106-110.

У статті запропонована математична модель дифузії нейротрансмітера у нейтронному синапсі в процесі везикулярного екзоцитозу. Наведено модельне уявлення для електрохімічного штучного синапсу — широківживаний метод дослідження емісії/секреції біологічно активних речовин клітинами — що дозволяє обчислювати струми для порівняння з даними, отриманими експериментальним шляхом. Дану задачу було розв'язано чисельно за допомогою квазиконформного відображення. Вірність чисельного розв'язку було доведено порівнянням з відомими розв'язками для деяких граничних випадків та перевіркою виконання закону збереження заряду.

Лл. 5. Бібліогр.: 11 найм.

UDK 57:51-76; 57.02.001.57; 517.958:57

Mathematical modelling of neurotransmitter diffusion from a vesicle at fixed opening angles / A.I. Oleinick // Bionics of Intelligence: Sci. Mag. — 2009. — № 2 (71). — P. 106-110.

A mathematical model of neurotransmitter diffusion within a neuronal synapse during vesicular exocytosis is proposed. The model view for the electrochemical artificial synapse, which is a widespread method for studying cell emission/secretion of biologically active species, is given. This allows the computation of current for comparison with experimental data. The problem was solved numerically via the application of the proposed in this work quasiconformal mapping. The validity of the numerical solution was proven by its comparison with known analytical solutions for limiting cases and by ensuring the fulfilment of the charge conservation law.

Fig. 5. Ref.: 11 items.

УДК 517.957:612.821



ОПРЕДЕЛЕНИЕ СТРУКТУРНОЙ БЛИЗОСТИ УЧАСТКОВ ЭЭГ СИГНАЛА ДЛЯ РАСПОЗНАНИЯ УМСТВЕННЫХ ЗАДАЧ, ВЫПОЛНЯЕМЫХ ЧЕЛОВЕКОМ

С.С. Кротких¹, Л.О. Кириченко²

1 ХНУРЭ, г. Харьков, Украина, seemann@inbox.ru;

2 ХНУРЭ, г. Харьков, Украина, ludmila@kture.kharkov.ua

Разработан метод определения структурной близости участков ЭЭГ сигнала на основе расчета кросс-корреляционного интеграла в восстановленном фазовом пространстве. Данный метод позволяет выделить участки электроэнцефалограммы, на которых фиксируется выполнение схожих умственных задач. Для предварительной обработки ЭЭГ сигнала и удаления артефактов был применен метод «Гусеница».

ИНДЕКС СТАЦИОНАРНОСТИ, УЧАСТОК ЭЭГ, КРОССКОРРЕЛЯЦИОННЫЙ ИНТЕГРАЛ, АРТЕФАКТ, ВОССТАНОВЛЕНИЕ АТТРАКТОРА СИСТЕМЫ

Введение

В настоящее время для анализа функций головного мозга широко используется электрический энцефалограф (ЭЭГ), который позволяет измерить изменение электрического потенциала в отдельных участках головного мозга человека. ЭЭГ сигнал можно рассматривать как фиксацию последовательности умственных задач, выполняемых мозгом субъекта. Данная интерпретация позволяет поставить задачу о возможности разработки интерфейса мозг-компьютер. Однако задача определения участка ЭЭГ сигнала, соответствующего выполнению определенной умственной задачи, в настоящее время еще остаётся нерешенной. Это обусловлено тем, что ЭЭГ сигнал является индивидуальным для каждого человека, и высокоточное обнаружение выполнения мозгом той или иной задачи является невозможным без исследования состояний других систем организма. К тому же различные умственные задачи или условия, в которых находится человек, имеют свои уникальные отражения в результирующем ЭЭГ сигнале. Таким образом, наличие тех или иных изменений в сигнале может свидетельствовать об изменении условий или внутреннего физиологического состояния человека [1, 2].

Головной мозг человека обладает электрическим полем, которое характеризуется электрическими скачками от нескольких миллиампер до нескольких сотен миллиампер. Анализ электрического поля мозга широко применяется при диагностике различных неврологических заболеваний. Каждый нейрон мозга генерирует изменение электрического потенциала, этот процесс можно измерить. Измерение может быть проведено инвазивно – посредством имплантации специальных электродов, которые могут измерять активность отдельных участков мозга, и неинвазивно – с использованием электроэнцефалограммы (ЭЭГ) или магнетоэнцефалограммы (МЭГ). Так как МЭГ требует дорогостоящей аппаратуры, предпочтение отдается ЭЭГ.

ЭЭГ измеряет электрический потенциал с помощью нескольких электродов, расположенных в определенных точках на поверхности головы субъекта. Количество электродов, в зависимости от модели энцефалографа, может быть от 8 до 32. Если субъект жив, электрическая мозговая активность присутствует постоянно. За исключением воздействий мускульной активности и моргания глаз сигнал распределен на частотах от 0,5 Гц до 40 Гц. Этот промежуток принято разделять на частотные диапазоны, называемые ритмами. Ритмы могут отличаться друг от друга не только частотой, но и амплитудой.

Общепринята классификация ритмов ЭЭГ по некоторым основным диапазонам [3]. Под понятием «ритм» на ЭЭГ подразумевается определенная полоса частот, соответствующая некоторому состоянию мозга. К ритмам ЭЭГ взрослого бодрствующего человека относятся альфа (α), бета (β), мю (μ), гамма (γ) – ритмы. Присутствие каждого из диапазонов зависит от условий и состояния субъекта (табл. 1).

Таблица 1

Основные ритмы ЭЭГ человека

Частотный диапазон	Частоты
Гамма (μ)	40 – 70 Гц
Альфа (α), Мю (μ)	8 – 13 Гц
Бета (β)	14 – 30 Гц
Тета (θ)	4 – 7 Гц
Дельта (δ)	0,5 – 3 Гц

Целью данной работы является разработка метода идентификации умственных процессов, зафиксированных на определенных участках ЭЭГ сигнала, основанного на определении структурной схожести между двумя различными участками сигнала с помощью вычисляемого «коэффициента стационарности». Данный коэффициент принимает значения от 0 до 100 и определяет степень близости двух временных рядов. Его минимальное

значение говорит о полной идентичности рассматриваемых рядов. Значение коэффициента стационарности близкое к 100 означает слабую схожесть между сравниваемыми участками.

Для реализации цели были поставлены и решены следующие задачи: распознавание и удаление артефактов из записи ЭЭГ сигнала; восстановление аттрактора динамической системы, описанной сигналом; идентификация участков ЭЭГ сигнала со схожей внутренней динамикой.

1. Удаление артефактов с помощью метода «гусеница»

Как правило, ЭЭГ сигнал кроме основного сигнала содержат артефакты (примеси), не имеющие отношения к характеристике функции головного мозга и затрудняющие анализ основного сигнала. Подобные примеси вызваны неточностями измерительных приборов, а также помехами от движения глаз и мускульной активности на поверхности головы испытуемого. Перед проведением анализа исходный сигнал необходимо подвергнуть предварительной обработке и исключить помехи. Для удаления артефактов используются различные методы анализа компонентов сигнала, включая методы на основе фильтров Фурье и вейвлет-фильтров. Однако использование этих методов зачастую приводит к потере данных из-за удаления зашумленных участков из основного сигнала. К тому же некоторые из них успешно применяются только в случае стационарных сигналов.

Одним из методов анализа временных рядов является метод «Гусеница» или SSA (Singular Spectrum Analysis). Данный метод разрабатывался одновременно в России и США и известен своей эффективностью применительно к анализу нестационарных рядов. Целью метода является разложение временного ряда на интерпретируемые аддитивные составляющие. При этом метод не требует стационарности ряда, знания модели тренда, а также сведений о наличии в ряде периодических составляющих и их периодов. При таких достаточно слабых предположениях метод «Гусеница» может решать различные задачи, такие как: выделение тренда, обнаружение периодических компонент, сглаживание ряда, построение полного разложения ряда в сумму тренда, периодик и шума. [4]

Платой за такой широкий спектр возможностей является, во-первых, существенно неавтоматическая группировка компонентов сингулярного разложения траекторной матрицы ряда для получения составляющих исходного ряда; во-вторых, отсутствие модели не позволяет проверить гипотезы о наличии в ряде той или иной составляющей (этот недостаток объективно присущ большинству непараметрических методов). Для проверки подобных гипотез требуется построение модели,

которое, в свою очередь, может быть проведено на основе информации, получаемой с помощью метода «Гусеница».

Для анализа временного ряда выбирается целый параметр длины окна L . При достаточно большой длине ряда и большом L результаты не будут зависеть от длины окна. Затем на основе ряда строится траекторная матрица, столбцами которой являются скользящие отрезки ряда длины L : с первой точки по L -ю, со второй по $L+1$ -ю и так далее. Следующим шагом является сингулярное разложение траекторной матрицы в сумму элементарных матриц. Каждая элементарная матрица задается набором из собственного числа и двух сингулярных векторов: собственного и факторного.

Предположим, что исходный временной ряд является суммой нескольких рядов. Теоретические результаты позволяют, при некоторых условиях, определить по виду собственных чисел, собственных и факторных векторов, что это за слагаемые и какой набор элементарных матриц соответствуют каждому из них. Суммируя элементарные матрицы внутри каждого набора и затем переходя от результирующих матриц к ряду, мы получаем разложение ряда на аддитивные слагаемые, например, на сумму тренда, периодики и шума, или на сумму низкочастотной и высокочастотной составляющих. Возможность разбить совокупность элементарных матриц на группы, соответствующие интерпретируемым аддитивным составляющим ряда, тесно связана с понятием разделимости рядов.

Рассмотрим базовый алгоритм метода «Гусеница».

Шаг 1. Процедура вложения. На этом этапе исходный временной ряд длиной N f_0, \dots, f_{N-1} переводится в последовательности многомерных векторов.

Пусть L – длина окна, $1 < L < N$. Процедура вложения образует $K = N - L + 1$ векторов вложения:

$$X_i = (f_{i-1}, \dots, f_{i+L-2})^T, 1 \leq i \leq K,$$

имеющих размерность L . Траекторная матрица L состоит из векторов вложения в качестве столбцов. Другими словами, траекторная матрица – это матрица вида:

$$X = (x_{ij})_{i,j=1}^{L,K} = \begin{pmatrix} f_0 & f_1 & f_2 & \dots & f_{K-1} \\ f_1 & f_2 & f_3 & \dots & f_K \\ f_2 & f_3 & f_4 & \dots & f_{K+1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ f_{L-1} & f_L & f_{L+1} & \dots & f_{N-1} \end{pmatrix}.$$

Шаг 2. Сингулярное разложение. Цель этого шага – получить сингулярное разложение полученных векторов. Пусть $S = XX^T$. Обозначим $\lambda_1, \dots, \lambda_L$ собственные числа матрицы S , взятые в неубывающем порядке и U_1, \dots, U_L – ортонорми-

рованную систему собственных векторов матрицы S , соответствующих собственным числам.

Пусть $d = \max\{i, \text{таких что } \lambda_i > 0\}$. Если обозначить $V_i = X^T U_i / \sqrt{\lambda_i}, i=1, \dots, d$, то сингулярное разложение матрицы S может быть записано как:

$$X = X_1 + \dots + X_d, \quad (1)$$

где $X_i = \sqrt{\lambda_i} U_i V_i^T$. Каждая из матриц X_i имеет ранг 1, поэтому их можно назвать элементарными матрицами. Набор $(\sqrt{\lambda_i}, U_i, V_i)$ называется i -й собственной тройкой сингулярного разложения (1).

Шаг 3. Группировка. На основе разложения (1) делим все множество $\{1, \dots, d\}$ на m непересекающихся подмножеств I_1, \dots, I_m .

Пусть $I = \{i_1, \dots, i_m\}$. Тогда результирующая матрица X_I , соответствующая группе I , определяется как:

$$X = X_{i_1} + \dots + X_{i_p}.$$

Такие матрицы вычисляются для $I = I_1, \dots, I_m$, тем самым разложение (1) может быть записано в сгруппированном виде

$$X = X_{I_1} + \dots + X_{I_m}. \quad (2)$$

Процедура выбора I_1, \dots, I_m называется группировкой собственных троек.

Шаг 4. Диагональное усреднение. На последнем шаге базового алгоритма каждая матрица сгруппированного разложения (2) переводится в новый ряд длины N .

Пусть Y – матрица размерности $L \times K$ с элементами $y_{ij}, 1 \leq i \leq L, 1 \leq j \leq K$. Положим $L^* = \min(L, K)$, $K^* = \max(L, K)$, $N = L + K - 1$. Пусть $y_{ij}^* = y_{ij}$, если $L < K$, и $y_{ij} = y_{ij}^*$ в противном случае.

Диагональное усреднение переводит матрицу Y в ряд g_0, \dots, g_{N-1} по формуле:

$$g_k = \begin{cases} \frac{1}{k+1} \sum_{m=1}^{k+1} y_{m, k-m+2}^* \\ \frac{1}{L^*} \sum_{m=1}^{L^*} y_{m, k-m+2}^* \\ \frac{1}{N-K^*} \sum_{m=k-K^*+2}^{N-K^*+1} y_{m, k-m+2}^* \end{cases}$$

Данное выражение соответствует усреднению элементов матрицы вдоль диагоналей $i + j = k + 2$: выбор $k = 0$ дает $g_0 = y_{11}$, для $k = 1$ получаем $g_1 = (y_{12} + y_{21})/2$ и так далее. Заметим, что если матрица Y является траекторной матрицей некоторого ряда (h_0, \dots, h_{N-1}) (другими словами, если матрица Y является ганкелевой), то $g_i = h_i$ для всех i .

Применив диагональное усреднение к результирующим матрицам X_{I_k} , мы получаем ряды $\tilde{F}^{(k)} = (g_0^{(k)}, \dots, g_{N-1}^{(k)})$ и, следовательно, исходный ряд f_0, \dots, f_{N-1} складывается в сумму m рядов:

$$f_n = \sum_{k=1}^m g_n^{(k)}.$$

2. Применение методов нелинейной динамики к исследованию свойств ЭЭГ сигнала

Исследование ЭЭГ сигнала может быть проведено с помощью методов, развитых в теории динамического хаоса при предположениях, что мозг (или его часть) рассматривается как нелинейная динамическая система, чувствительная к начальным условиям, и ЭЭГ является траекторией этой динамической системы в фазовом пространстве. Таким образом, предполагается, что биоэлектрическая активность описывается неявно заданной хаотической системой и, несмотря на отсутствие системы уравнений, моделирующих биоэлектрическую активность мозга, можно изучать поведение системы по ее выходным данным. [5]

Главная идея применения методов нелинейной динамики к анализу траектории динамической системы состоит в том, что основная структура, содержащая в себе всю информацию о системе, а именно, аттрактор динамической системы, может быть восстановлена через измерение только одного компонента этой динамической системы. Процедура реконструкции фазового пространства и восстановление аттрактора системы сводится к построению псевдофазового пространства. Во многих практических ситуациях восстановление аттрактора может оказаться единственным способом, позволяющим производить его наблюдение [6].

Функция восстановления (процедура Паккарда-Такенса) определяется как

$$F(t) = \{x_i(t), x_i(t+\tau), \dots, x_i(t+(m-1)\tau)\},$$

где $x_i(t+n\tau)$ — i -я компонента траектории системы, а τ — период дискретизации, выбираемый некоторым произвольным образом.

Восстановление аттрактора можно производить практически при любом значении τ , однако все же существуют определенные ограничения. Если значение τ слишком мало, автокорреляция между значениями $x_i(t+k\tau)$ и $x_i(t+(k+1)\tau)$ близка к единице, и восстановленный аттрактор оказывается ограниченной областью вблизи диагонали пространства, в котором производится восстановление. Если же значение τ слишком велико, а система является хаотической, то значения $x_i(t+k\tau)$ и $x_i(t+(k+1)\tau)$ оказываются некоррелированными и структура аттрактора исчезает.

Предположим, что данный временной ряд порожден некоторой динамической системой и, что m — наименьшая размерность фазового пространства, в которое можно погрузить реальный аттрактор динамической системы, называемой размерностью вложения. Тогда аттрактор системы, восстановленный посредством временного ряда $x_i(t)$, формируется из векторов $\{x_i(t), x_i(t+\tau), \dots, x_i(t+(m-1)\tau)\}$ в m -мерном псевдофазовом пространстве. Если временной ряд

действительно является наблюдаемой реализацией хаотической динамической системы, то реальный аттрактор динамической системы и аттрактор, восстановленный в псевдофазовом пространстве по временному ряду, при правильном подборе размерности вложения m являются топологически эквивалентными и обладают одинаковыми фрактальными размерностями, ляпуновскими показателями и другими численными характеристиками нелинейной системы.

Один из наиболее употребительных методов, применяемых на практике для выяснения наличия хаотической детерминированности и определения фрактальной размерности аттрактора, состоит в изучении свойств корреляционного интеграла $C_m(\epsilon)$ и поведения корреляционной размерности $d_C(m)$ в зависимости от размерности вложения m . Для расчета корреляционного интеграла необходимо построить m -мерное псевдофазовое пространство. Корреляционный интеграл $C_m(\epsilon)$ — это вероятность того, что пара точек на восстановленном аттракторе в m -мерном пространстве находится в пределах расстояния ϵ друг друга. Для этого вычисляются расстояния между парами точек, и определяется $C_m(\epsilon) = \lim_{\epsilon \rightarrow 0} \frac{L_{ij}}{N^2}$, где L_{ij} — число пар точек (i, j) , для которых расстояние между ними меньше ϵ , N — общее число точек в фазовом пространстве. Зависимость корреляционного интеграла от ϵ при малых ϵ подчиняется степенному закону, то есть $\lim_{\epsilon \rightarrow 0} C_m(\epsilon) = a\epsilon^{d_C}$. Поэтому корреляционную размерность $d_C(m)$ определяют по наклону графика зависимости $C_m(\epsilon)$ от $1/\epsilon$ в двойном логарифмическом масштабе.

Если график функции $\log C_m(\epsilon)$ относительно $\log(1/\epsilon)$ имеет отчетливо выраженный линейный участок, это указывает на самоподобную геометрию аттрактора, что, в свою очередь, говорит о хаотической детерминированности временного ряда. При увеличении размерности вложения корреляционная размерность $d_C(m)$ увеличивается. Однако для хаотических данных корреляционная размерность будет, в конечном счете, насыщаться при ее истинном значении. Фрактал, вложенный в пространство с более высокой размерностью, сохраняет свою истинную размерность из-за нелинейных корреляций между точками. Поэтому для детерминированного хаотического временного ряда корреляционная размерность сходится к ее истинному значению, а значение m , при котором $d_C(m)$ перестает изменяться, является размерностью вложения.

3. Определение индекса стационарности

Как было сказано выше, для анализа временных рядов широко используется автокорреляционный интеграл, который можно определить следующим образом:

$$C_m = P(\|\overrightarrow{x^m(i)} - \overrightarrow{x^m(j)}\| < \epsilon) = \frac{1}{N^2} \sum_{i,j} \theta(\epsilon - \|\overrightarrow{x^m(i)} - \overrightarrow{x^m(j)}\|),$$

где θ — функция Хэвисайда (равна единице, если ее параметр положителен, и нулю, если параметр отрицателен), $\overrightarrow{x^m}$ — вектор изображающей точки в псевдофазовом пространстве размерности m , N — количество точек, ϵ — расстояние между точками, $\|\dots\|$ — норма, определяющая расстояние между точками. Таким образом, автокорреляционный интеграл представляет собой вероятность того, что две точки в восстановленном фазовом пространстве будут ближайшими соседями (находятся на расстоянии, меньшем, чем ϵ друг от друга).

Для двух различных временных рядов $x(i)$ и $y(i)$ кросскорреляционный интеграл определяется следующим образом:

$$C_m(x, y) = P(\|\overrightarrow{x^m(i)} - \overrightarrow{y^m(j)}\| < \epsilon) = \frac{1}{N^2} \sum_{i,j} \theta(\epsilon - \|\overrightarrow{x^m(i)} - \overrightarrow{y^m(j)}\|).$$

То есть, кросскорреляционный интеграл — это вероятность нахождения точек ряда x в восстановленном фазовом пространстве на расстоянии, меньшем ϵ , с точками временного ряда y . Если два сигнала имеют одни и те же динамические свойства, тогда значения $C_m(x, x)$, $C_m(x, y)$ и $C_m(y, y)$ должны быть приблизительно равны при любой размерности m , то есть

$$C_m(x, x) - C_m(x, y) \approx 0, \\ C_m(y, y) - C_m(x, y) \approx 0.$$

Эти условия являются основой для понятия индекса стационарности, предложенного в [2]. Процедура вложения дает представление обобщенного вероятностного распределения точек во временном ряду, а кросскорреляционный интеграл позволяет рассчитать меру вероятности того, что точки одной траектории в фазовом пространстве находятся достаточно близко к точкам другой траектории, давая представление о мере подобия между траекториями. Таким образом, можно получить меры схожести обобщенного вероятностного распределения двух временных рядов, по которым были построены траектории в фазовом пространстве.

Негативное воздействие высокой автокорреляции в исходном сигнале на корреляционный интеграл устраняется в определенной мере благодаря применению окна Тейлера. Покажем это, переписав корреляционный интеграл следующим образом:

$$C_m(\epsilon, N) = \frac{2}{N^2} \sum_{n=1}^N \sum_{i=1}^{N-n} \theta(\epsilon - \|\overrightarrow{x^m(i)} - \overrightarrow{y^m(j)}\|).$$

В более обобщенном виде его можно записать так:

$$C_m(\varepsilon, N) = \frac{2}{N^2} \sum_{n=W}^N \sum_{i=1}^{N-n} \theta(\varepsilon - \|x^m(i) - y^m(j)\|).$$

Здесь W – окно Тейлера. При длине окна $W > \tau(2/N)^{2/m}$, где N – общее число точек, τ – время автокорреляции, m – размерность вложения и $\varepsilon > \sigma\sqrt{\pi}(2/N)^{1/m}$, где σ – выборочная дисперсия временного ряда, корреляционная функция учитывает только случайно близко расположенные точки векторов, а не те, которые близки из-за высокой автокорреляции [7]. Расчеты длины окна Тейлера и требования к минимальному расстоянию между точками нацелены на то, чтобы снизить негативное воздействие автокорреляции на восстановленный аттрактор системы.

Индекс стационарности (ИС) определяется следующим образом:

$$\text{ИС} = \frac{C_m(x, x) - C_m(x, y)}{C_m(x, x) + C_m(x, y)} \cdot 100.$$

Этот параметр позволяет оценить степень структурной близости между двумя участками сигнала. Индекс стационарности может принимать значения из диапазона от 0 до 100. Значение ИС тем больше, чем больше разница между основными динамиками участков x и y . В частности, ИС равен 100, когда $C_m(x, y) = 0$. В этом случае сравниваемые участки настолько различны, что ни одна пара точек, принадлежащих этим участкам, не лежит в достаточно малой близости друг от друга.

4. Результаты исследований

4.1 Удаление артефактов

В качестве исходного сигнала для проведения практического эксперимента были использованы электроэнцефалографические сигналы, записанные в лабораторных условиях у пациентов без патологий функции мозга. Запись производилась с частотой 120 Гц в течение 16 сек. Анализируемый сигнал был получен с электрода, расположенного в районе левой лобной доли испытуемого. Программное приложение, использованное для анализа сигнала ЭЭГ, написано на языке C# и имеет графический интерфейс, позволяющий загружать файлы с данными, проводить предварительную обработку и выполнять расчёт индекса стационарности участков сигнала.

Исходная ЭЭГ содержит достаточно большой диапазон частот, что свидетельствует о наличии артефактов, которые не представляют интереса для анализа. Для удаления артефактов был применен метод «Гусеницы», который позволяет выявить и исключить неинформативные составляющие, такие как помехи от движения глаз, сокращения мышц, посторонние шумы. На рис. 1 приведен

участок ЭЭГ сигнала, предположительно содержащий артефакты.

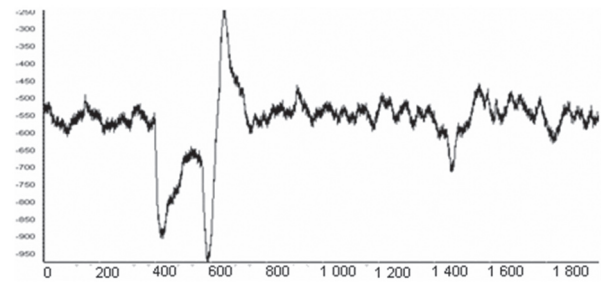


Рис. 1. Участок сигнала ЭЭГ

Для того чтобы определить наличие артефактов и исключить их из сигнала, построим сингулярное разложение этого участка. На основе полученного разложения восстановим соответствующие временные ряды. Подсигнал, восстановленный по сингулярной тройке с наибольшим значением сингулярного числа, имеет наибольший вклад в результирующий сигнал. Он показан на рис. 2а и представляет собой низкочастотную составляющую сигнала. Построим также график суммы двух первых подсигналов, имеющих наибольший вес в результирующем сигнале (рис. 2б). Очевидно, что восстановленный сигнал находится намного ближе к исходному сигналу в местах амплитудных скачков, чем на первом графике, в то время как остальные участки графика к исходному сигналу приблизились незначительно. Последующие подсигналы имеют более высокую частоту, поэтому не несут информации о низкочастотных колебаниях, которыми являются артефакты, вызванные морганием глаз или сокращением мышц. Таким образом, можно ввести критерий наличия артефакта в точке x_i :

$$|x_i^{sub1} - x_i^{sub1+sub2}| > \delta, \quad (3)$$

где x_i^{sub1} – значение восстановленного ряда с наибольшим значением сингулярного числа в точке i ; $x_i^{sub1+sub2}$ – значение суммы двух восстановленных рядов с наибольшими значениями сингулярного числа в точке i ; δ – некоторое выбранное пороговое значение.

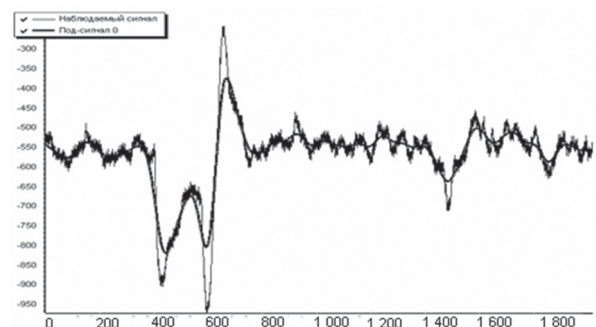


Рис. 2а. Сигнал, восстановленный по сингулярной тройке с наибольшим сингулярным числом (чёрная линия) и наблюдаемый ЭЭГ сигнал (серая линия)

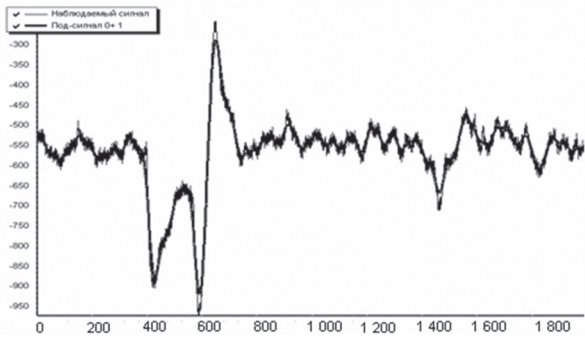


Рис. 26. Сумма сигналов, восстановленных по двум сингулярным тройкам (чёрная линия) и наблюдаемый ЭЭГ сигнал (серая линия)

Построив график левой части неравенства (3), отметим, что в районе точек 370, 420, 570, 590, 650, 1460 наблюдаются существенные отклонения x_i^{sub1} от $x_i^{sub1+sub2}$ (рис. 3). Таким образом, можно утверждать, что участок ЭЭГ сигнала на промежутках (320-455), (520-670), (1450-1470) имеет высокоамплитудные скачки, которые можно расценить как артефакты и исключить из анализируемого сигнала.

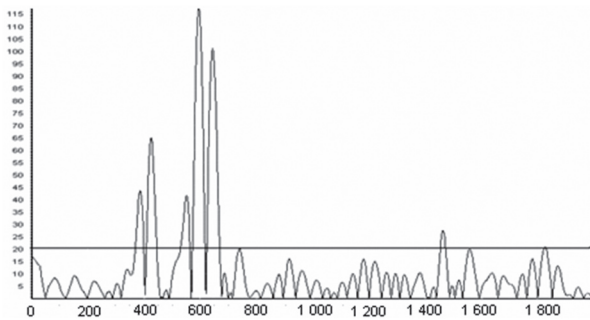


Рис. 3. Разность между сигналами sub1 и sub1+sub2

С помощью преобразования Фурье сигнал был разложен на частотные составляющие с целью исключить влияние высокочастотных колебаний и колебаний, не принадлежащих к известным ритмам [3]. Таким образом, подготовленный к анализу сигнал содержит такие частотные ритмы: $\sigma(0,5-3\text{Гц})$, $\theta(4-7\text{Гц})$, $\alpha(8-13\text{Гц})$, $\mu(40-70\text{Гц})$.

После удаления артефактов и частотной фильтрации сигнал выглядит более гладким, чем исходный. Временной ряд, прошедший предварительную обработку и подготовленный для выделения стационарных участков, представлен на рис. 4.

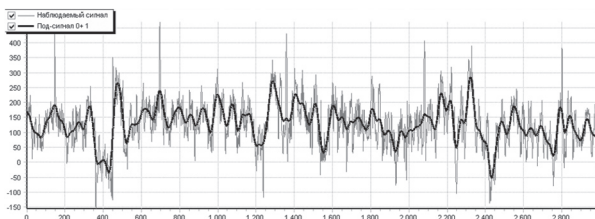


Рис. 4. Исходный ЭЭГ сигнал (серая кривая) и фильтрованный сигнал (черная кривая)

4.2 Определение участков стационарной близости

Реализацию ЭЭГ сигнала необходимо разбить на отдельные участки для определения их стационарной близости. Количество и соответствующая длина участков выбирается в соответствии с поставленной задачей. Для исходного временного ряда были вычислены параметр длины окна Тейлера и размерность пространства вложения m [8]. В нашем случае временной ряд был разделен на 18 равных частей. Для каждого из полученных участков был вычислен индекс стационарности относительно остальных участков.

В связи с тем, что целью исследования является определение участков временного ряда, схожих между собой по структуре и внутренней динамике, целесообразно построить график таким образом, чтобы можно было легко определить номера этих участков. На рис. 5 представлена пузырьковая диаграмма, на которой по обеим осям отложены номера участков. Радиус пузырьков, лежащих на пересечении номеров участков, указывает на степень структурной связи между этими участками. Чем больше радиус пузырька, тем сильнее зависимость между соответствующими участками. Диаграмма имеет диагональный вид, так как верхняя ее половина – зеркальное отражение нижней.

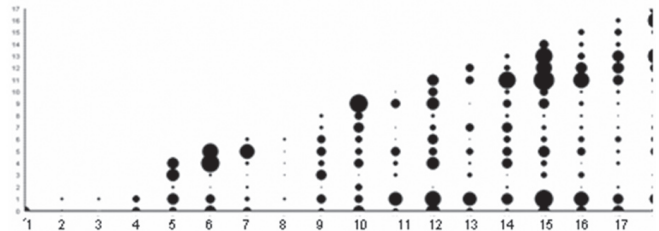


Рис. 5. График индексов стационарности.

По осям отложены номера участков временного ряда

На рис. 5 видно, что, например, один из пузырьков с максимальным радиусом расположен на пересечении участков 11 и 15. Отсюда можно с достаточной большой вероятностью сделать вывод, что в моменты времени, соответствующие участкам 11 и 15, фиксируется выполнение некоторых схожих процессов. На рис. 6 черным цветом выделены участки 11 и 15. Стоит отметить, что по одной лишь кривой сигнала невозможно распознать структурную схожесть этих участков.

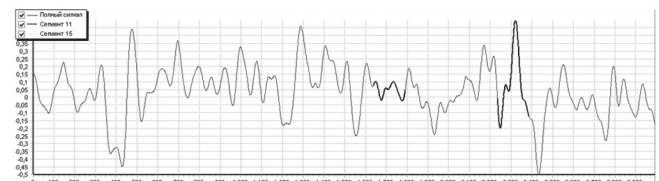


Рис. 6. Анализируемый временной ряд с отмеченными участками стационарной близости

Выводы

Результаты использования индекса стационарности показали, что данный метод может быть использован для решения задачи выделения участков ЭЭГ сигнала со схожей внутренней динамикой. Следующим шагом после определения меры схожести между участками ЭЭГ сигнала является создание базы данных эталонных участков ЭЭГ сигналов для дальнейшего сравнения их динамики с динамикой соответствующих участков анализируемого сигнала. Схожесть участков анализируемого сигнала с одним из эталонных свидетельствовала бы о выполнении соответствующей умственной задачи и привела бы к выполнению системой некоторых заранее заложенных действий.

Определение наиболее близких по структуре участков электроэнцефалографического сигнала является одной из сложных задач, решение которой поможет сделать большой шаг вперед к возможности разработки систем, управляемых усилием мысли. Подобные системы призваны дать возможность людям с дисфункцией двигательной системы общаться с внешним миром. Другим применением может быть альтернативное устройство управления компьютером, которое может стать намного удобнее привычной клавиатуры и мыши.

Список литературы: 1. Torsten Felzer On the Possibility of Developing a Brain-Computer Interface (BCI) / Torsten Felzer // Technical University of Darmstadt, Department of Computer Science, 2001. – P.27-32 2. Bunty V. E. Kiremire Nonstationarity detection: The use of the cross correlation integral in ECG, and EEG profile analysis / Bunty V. E. Kiremire, Tshilidzi Marwala // Congress on Image and Signal Processing. The University of the Witwatersrand, – 2008. – P.373-378. 3. Зенков, Л.П. Функциональная диагностика нервных болезней: руководство для врачей [текст] / Л.П. Зенков, М.А. Ронкин – М.: Медицина, 1991. – 640с. 4. Голяндина, Н.Э. Метод «Гусеница»-SSA: анализ временных рядов [текст] : учеб. пособие. / Н.Э. Голяндина – СПб.: СПбГУ, 2004. – 76 с. 5. Кануников, И.Е. Применение теории динамического хаоса для анализа электроэнце-

фалогрaмм [текст] / И.Е. Кануников, Е.В. Антонова, Д.Р. Белов, Ю.Г. Марков // Вестник СПбГУ. Сер. 3 (биология). – 1998. – Вып.1. – №3. – С.55-61 6. Анищенко, В.С. Сложные колебания в простых системах: Механизмы возникновения, структура и свойства динамического хаоса в радиофизических системах. [текст] / В.С. Анищенко – М.: Наука. Гл. ред. физ.-мат. лит. – 1990. – 246 с 7. Theiler. Spurious dimension from correlation algorithms applied to limited time-series data. / Theiler // Physical Review A, 34(3). – 1986. – P.123-130 8. Кротких С.С., Удаление артефактов при определении фрактальной размерности электро-энцефалографического сигнала [текст]: XIII международный молодежный форум «Радиоэлектроника и молодежь». / С.С. Кротких, Л.О. Кириченко – 2009. – С. 282.

Поступила в редколлегию: 29.10.2009

УДК 517.957:612.821

Визначення структурної близькості ділянок ЕЕГ сигналу для розпізнавання розумових задач, що виконуються людиною / Кротких С.С., Кириченко Л.О. // Біоніка інтелекту: наук.-техн. журнал. – 2009. – № 2 (71). – С. 111-117.

У роботі численно визначено ступінь близькості між ділянками ЕЕГ сигналу з використанням методів нелінійної динаміки. Графічні методи представлення результатів полегшують пошук ділянок, що мають схожі структурні та динамічні властивості. Одержані результати дозволяють більш точно ідентифікувати процеси, що відбуваються у мозку досліджуваного, і проводити діагностику складних захворювань.

Л. 6. Бібліогр.: 8 назв.

UDK 517.957:612.821

The determining of the structural closeness between the EEG signal segments for identification of human mental performance. / Krotkih S.S., Kirichenko L.O. // Bionics of Intelligence: Sci. Mag. – 2009. – № 2 (71). – P. 111-117.

In article it has been attempted to numerically measure the closeness between segments of EEG signal by using the nonlinear dynamics methods. The graphic methods of data presenting are intended to ease the visual determining of the closeness between the segments. The obtained results may enable a more reliable brain disease diagnostics as well as provide an opportunity to communicate through thought.

Fig. 6. Ref.: 8 items

УДК 004.93



СТРУКТУРНО-ПАРАМЕТРИЧЕСКАЯ ИДЕНТИФИКАЦИЯ НЕЙРО-НЕЧЕТКИХ МОДЕЛЕЙ ДЛЯ ДИАГНОСТИКИ

С.А. Субботин

Запорожский национальный технический университет, г. Запорожье, Украина, subbotin@zntu.edu.ua

Предложены методы построения, дообучения и упрощения нейро-нечетких диагностических моделей по прецедентам. Разработанные методы позволяют в неитеративном режиме устанавливать значения весовых коэффициентов нейро-нечетких сетей, не требуют вычисления производных, выбора начальной точки поиска, что позволяет ускорить процесс получения нейро-нечеткой модели по прецедентам.

ДИАГНОСТИКА, РАСПОЗНАВАНИЕ ОБРАЗОВ, НЕЧЕТКАЯ СИСТЕМА, ТЕРМ, НЕЙРО-НЕЧЕТКАЯ СЕТЬ, КЛАССИФИКАЦИЯ, ПРИЗНАК, МЕТОД ПОТЕНЦИАЛЬНЫХ ФУНКЦИЙ

Введение

Автоматизация процессов принятия решений является средством повышения эффективности процедур технической и биомедицинской диагностики, поскольку снижает негативное влияние человеческого фактора (усталость, невнимательность), повышает производительность труда, а также делает процесс принятия решений хорошо формализованным, понятным и предсказуемым.

Нейро-нечеткие сети [1–3] как инструмент автоматизации принятия решений в задачах технической и биомедицинской диагностики нашли широкое применение благодаря сочетанию таких свойств нейронных сетей, как способность к обучению и обобщению по примерам, со свойством логической прозрачности (интерпретируемости) нечетких моделей.

Вместе с тем при построении нейро-нечетких моделей на основе экспериментальных данных возникает необходимость выделения нечетких термов признаков, что, как правило, на практике возлагается на пользователя либо достигается в результате кластер-анализа [2, 3], а также требует последующей высокитеративной коррекции весов модели и параметров нечетких термов, расчета производных и выбора начальной точки поиска в пространстве управляемых параметров модели, что является весьма нетривиальной задачей.

Целью работы является создание методов автоматического построения нейро-нечетких моделей, не требующего участия пользователя при выделении нечетких термов, а также позволяющего осуществлять структурно-параметрическую идентификацию нейро-нечетких сетей в неитеративном режиме.

1. Постановка задачи синтеза нейро-нечеткой модели

Пусть задана обучающая выборка $\langle x, y \rangle$, где $x = \{x^s\}$; $y = \{y^s\}$; $x^s = \{x_j^s\}$; x_j^s – значение j -го признака s -го экземпляра выборки; y^s – значение выходного признака, сопоставленное s -му экземпляру обучающей выборки; $y^s \in \{k\}$, $k = 1, 2, \dots, K$; K – число классов.

Необходимо сформировать разбиение признакового пространства на кластеры, для каждого кластера выделить нечеткие термы по признакам, определить схему нечеткого вывода номера класса из значений функций принадлежности к нечетким термам, представить нечеткую систему в нейробазисе (выполнить структурно-параметрическую идентификацию нейро-нечеткой модели).

2. Формирование разбиения признакового пространства

Определение нечетких термов требует разбиения пространства признаков на прямоугольные блоки (кластеры), проекции которых на оси признаков позволяют выделить интервалы значений признаков для соответствующих областей, на основе комбинации которых можно будет определить принадлежность распознаваемого экземпляра к блокам, а на их основе – принадлежности к классам. Формирование разбиения предлагается осуществлять путем выполнения последовательности шагов 1–12.

Шаг 1. Инициализация. Задать обучающую выборку $\langle x, y \rangle$.

Шаг 2. Определить минимальные и максимальные значения для каждого признака на множестве экземпляров обучающей выборки $\min(x_j)$ и $\max(x_j)$, на основе которых определить длину интервала наблюдаемых значений каждого признака: $r_j = \max(x_j) - \min(x_j)$, $j = 1, 2, \dots, N$.

Шаг 3. Определить по координатным расстояниям между экземплярами обучающей выборки, на основе которых найти для каждого признака:

$$R_j = \min_{\substack{s=1, 2, \dots, S; \\ g=s+1, s+2, \dots, S}} |x_j^s - x_j^g| \text{ или}$$

$$R_j = \min_{\substack{s=1, 2, \dots, S; \\ g=s+1, s+2, \dots, S}} \left\{ |x_j^s - x_j^g| \mid y^s \neq y^g \right\} \text{ или}$$

$$R_j = \frac{2}{S(S-1)} \sum_{s=1}^S \sum_{g=s+1}^S |x_j^s - x_j^g|,$$

$$j = 1, 2, \dots, N.$$

Шаг 4. Определить количество интервалов разбиения диапазона значений j -го признака n_j , исходя из формулы:

$$n_j = \begin{cases} \text{round}\left(\frac{r_j}{R_j}\right), & \text{если } \text{round}\left(\frac{r_j}{R_j}\right) \leq S; \\ S, & \text{в противном случае.} \end{cases}$$

Шаг 5. Рассчитать координаты левой и правой границ для p -го интервала j -го признака по формулам:

$$l_{j,p} = \min(x_j) + (p-1) \frac{r_j}{n_j},$$

$$r_{j,p} = \min(x_j) + p \frac{r_j}{n_j},$$

$$j = 1, 2, \dots, N; p = 1, 2, \dots, n_j.$$

Шаг 6. Задать для p -го интервала значений j -го признака ($j = 1, 2, \dots, N; p = 1, 2, \dots, n_j$) номер класса:

$$K_{j,p} = \begin{cases} 0, & \text{если } \neg \exists x^s, s = 1, 2, \dots, S: l_{j,p} \leq x_j^s \leq r_{j,p}; \\ y^s, & \text{если } \forall y^s = y^g; s, g = 1, 2, \dots, S: l_{j,p} \leq x_j^s \leq r_{j,p} \\ & \text{и } l_{j,p} \leq x_j^g \leq r_{j,p}; \\ -1, & \text{если } \exists y^s: y^s \neq y^g; s, g = 1, 2, \dots, S: l_{j,p} \leq x_j^s \leq r_{j,p} \\ & \text{и } l_{j,p} \leq x_j^g \leq r_{j,p}. \end{cases}$$

Шаг 7. Сформировать прямоугольные блоки $\{B_q\}$, $q = 1, 2, \dots, Q$, $Q = \prod_{j=1}^N n_j$, в N -мерном пространстве признаков на пересечении соответствующих интервалов значений признаков. Занести в $B_{q,j}$ номер интервала j -го признака, который соответствует q -му блоку.

Шаг 8. Определить номера классов для прямоугольных блоков в N -мерном пространстве признаков:

$$K_q = \begin{cases} K_{j,p}, & \text{если } \forall K_{j,p} = K_{i,t}; p = B_{q,j}, t = B_{q,i}, \\ & i = 1, 2, \dots, N; j = i + 1, \dots, N; \\ -1, & \text{в противном случае.} \end{cases}$$

Установить коэффициент уверенности классификации для блоков:

$$\alpha_q = \begin{cases} 1, & \text{если } K_q > 0; \\ 0, & \text{если } K_q \leq 0. \end{cases}$$

Шаг 9. Пока $\exists q: K_q = -1$, для тех блоков, у которых $K_q = -1$, выполнять процедуру разрешения конфликтов классов на основе одного из следующих способов.

Способ 1. Установить номер класса в соответствии с формулой:

$$K_q = \arg \max_{k=1,2,\dots,K} S_q^k,$$

где S_q^k — количество экземпляров k -го класса, попавших в q -ый блок.

Установить коэффициент уверенности:

$$\alpha_q = \frac{S_q^{K_q}}{\sum_{d=1}^K S_q^d}.$$

Достоинствами данного способа являются простота и высокая скорость вычислений, а недостатком — увеличение ошибки распознавания вследствие подавления одним классом других классов в блоке (огрубление аппроксимации границ классов).

Способ 2. Для исходного блока установить $K_q = 0$. Используя метод [1, 4, 5], разбить для q -го блока соответствующие ему интервалы значений признаков на интервалы с монотонным номером класса. Увеличить для каждого признака n_j : $n_j = n_j + n_j^*$, где n_j^* — количество интервалов с монотонным номером класса по j -му признаку внутри исходного блока. Занести в $l_{j,p}$ и $r_{j,p}$ границы новых интервалов, а в $K_{j,p}$ — номера классов, $j = 1, 2, \dots, N; p = n_j - n_j^*, \dots, n_j$. Образовать внутри исходного блока новые блоки B_q , а также соответствующим образом (как на шаге 8) определить для новых блоков номера классов K_q , и установить $\alpha_q = 1$, $q = Q+1, \dots, Q + \prod_{j=1}^N n_j^*$.

Занести в $B_{q,j}$ номер интервала j -го признака, который соответствует соответствующему q -му блоку. Установить: $Q = Q + \prod_{j=1}^N n_j^*$.

Достоинством данного способа является более высокая точность аппроксимации границ классов внутри исходного блока по сравнению с предыдущим способом, а недостатками — более низкая скорость и итеративный характер вычислений.

Способ 3. Для исходного блока установить $K_q = 0$. Рекурсивно выполнять шаги 2–9 для подмножества обучающей выборки, соответствующего разбиваемому блоку, корректируя соответствующие параметры n_j , $l_{j,p}$, $r_{j,p}$, $K_{j,p}$, B_q , K_q , $B_{q,j}$, α_q и Q .

Достоинством данного способа является очень высокая точность аппроксимации границ классов внутри исходного блока, а недостатками — существенно более низкая скорость, а также высокоитеративный и рекурсивный характер вычислений.

Шаг 10. Для тех блоков, у которых номер класса $K_q = 0$, определить расчетный номер класса, для чего предлагается использовать модифицированный нерекуррентный метод потенциальных функций [6, 7].

Шаг 10.1 Вычислить расстояние между q -ым и p -ым блоками, $q = 1, 2, \dots, Q$, $p = q+1, \dots, Q$, как:

$$R(B_q, B_p) = \sqrt{\sum_{j=1}^N r_j^{-1} (C_{q,j} - C_{p,j})^2} \quad \text{или}$$

$$R(B_q, B_p) = \sqrt{\sum_{j=1}^N r_j^{-1} |C_{q,j} - C_{p,j}|},$$

где $C_{q,j} = \frac{l_{j,B_{q,j}} + r_{j,B_{q,j}}}{2}$. Заметим, что $R(B_q, B_p) = R(B_p, B_q)$.

Шаг 10.2 Определить потенциал, наводимый совокупностью блоков, принадлежащих к k -му классу, на p -ый блок с неизвестной классификацией:

$$\Phi_p^k = \frac{1}{L_k} \sum_{q=1}^Q \left\{ S_q e^{-R^2(B_q, B_p)} \mid K_q = k, K_p = 0, q \neq p \right\},$$

где L_k – количество блоков, принадлежащих к k -му классу, S_q – количество экземпляров обучающей выборки, попавших в q -ый блок.

Шаг 10.3 Установить номер класса для p -го блока с неизвестной классификацией ($K_p = 0$) по формуле:

$$K_p = \arg \max_{k=1, 2, \dots, K} \Phi_p^k.$$

Шаг 10.4. Модифицировать значения коэффициентов уверенности для блоков:

$$\alpha_q = \{\Phi_q^{K_q} \mid \alpha_q = 0\}.$$

Шаг 11. Выполнить объединение смежных блоков, принадлежащих к одному и тому же классу: для $\forall q, p=1, 2, \dots, Q, q \neq p$: если $K_q > 0, K_q = K_p$ и $\exists j: |B_{q,j} - B_{p,j}| = 1, \forall i \neq j: B_{q,i} = B_{p,i}, i=1, 2, \dots, N, j=1, 2, \dots, N$; тогда объединить блоки q и p по j -му признаку:

– установить:

$$\alpha_q = \frac{\alpha_q \prod_{j=1}^N (r_{j, B_{q,j}} - l_{j, B_{q,j}}) + \alpha_p \prod_{j=1}^N (r_{j, B_{p,j}} - l_{j, B_{p,j}})}{\prod_{j=1}^N (r_{j, B_{q,j}} - l_{j, B_{q,j}}) + \prod_{j=1}^N (r_{j, B_{p,j}} - l_{j, B_{p,j}})},$$

$$n_j = n_j + 1,$$

$$l_{j, n_j} = l_{j, \min(B_{q,j}, B_{p,j})}, r_{j, n_j} = r_{j, \max(B_{q,j}, B_{p,j})}, B_{q,j} = n_j;$$

– удалить p -ый блок: $K_p = 0, \alpha_p = 0, B_{p,i} = 0, i=1, 2, \dots, N$.

Шаг 12. Останов.

3. Система нечеткого вывода

Для расчета принадлежностей распознаваемого экземпляра x^s к нечетким термам признаков целесообразно использовать трапецевидные функции принадлежности:

$$\mu_{j,p}(x^s) = \begin{cases} 0, x^s < l_{j,p}; \\ \frac{x^s - l_{j,p}}{l_{j,p}(2-\delta) - l_{j,p}}, l_{j,p} \leq x^s < l_{j,p}(2-\delta); \\ 1, l_{j,p}(2-\delta) \leq x^s \leq \delta r_{j,p}; \\ \frac{r_{j,p} - x^s}{r_{j,p} - \delta r_{j,p}}, \delta r_{j,p} < x^s \leq r_{j,p}; \\ 0, r_{j,p} < x^s, \end{cases}$$

где δ – некоторая константа, $0 < \delta \leq 1$.

Определив принадлежности экземпляра к термам признаков, необходимо определить его принадлежности к прямоугольным блокам:

$$\mu_q(x^s) = \min_{j=1, 2, \dots, N} \left\{ \min_{p=1, 2, \dots, n_j} \{ \mu_{j,p}(x^s) \mid B_{q,j} = p \} \right\},$$

$$q = 1, 2, \dots, Q;$$

после чего определить принадлежности к классам:

$$\mu_k(x^s) = \max_{q=1, 2, \dots, Q} \{ \alpha_q \mu_q(x^s) \mid K_q = k \},$$

$$k = 1, 2, \dots, K.$$

Четкий номер класса определим как

$$y = \arg \max_{k=1, 2, \dots, K} \{ \mu_k(x^s) \},$$

а уверенность в результате классификации

$$\mu_y = \max_{k=1, 2, \dots, K} \{ \mu_k(x^s) \}.$$

4. Синтез нейро-нечеткой сети

Рассмотренная система нечеткого вывода может быть представлена в нейробазисе в виде нейро-нечеткой сети, схема которой изображена на рисунке.

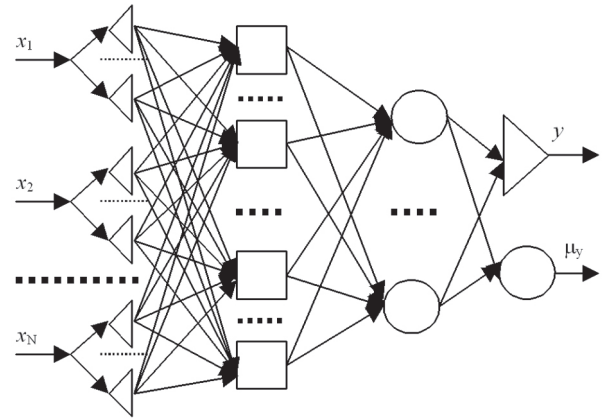


Схема нейро-нечеткой сети

На входы сети подаются значения признаков распознаваемого экземпляра. Нейроны первого слоя вычисляют принадлежности распознаваемого экземпляра к термам признаков $\mu_{j,p}(x^s)$ (фаззификация), нейроны второго слоя определяют принадлежности к блокам-кластерам $\mu_q(x^s)$, нейроны третьего слоя объединяют принадлежности к блокам в принадлежности к классам $\mu_k(x^s)$, после чего первый нейрон четвертого слоя осуществляет дефаззификацию результата, а второй нейрон четвертого слоя определяет достоверность классификации.

Функции постсинаптического потенциала нейронов сети будут задаваться формулами:

$$\phi_j^{(2,i)}(w_j^{(2,i)}, x_j^{(2,i)}) = \max\{w_j^{(2,i)}, x_j^{(2,i)}\},$$

$$i = 1, 2, \dots, Q; j = 1, 2, \dots, z;$$

$$\phi_j^{(3,i)}(w_j^{(3,i)}, x_j^{(3,i)}) = \min\{w_j^{(3,i)}, x_j^{(3,i)}\},$$

$$i = 1, 2, \dots, K; j = 1, 2, \dots, Q;$$

$$\phi_j^{(4,i)}(w_j^{(4,i)}, x_j^{(4,i)}) = \min\{w_j^{(4,i)}, x_j^{(4,i)}\},$$

$$i = 1, 2; j = 1, 2, \dots, K,$$

где $z = \sum_{j=1}^N n_j$; $x_j^{(n,i)}$ – значение сигнала на j -ом входе i -го нейрона n -го слоя сети; $w_j^{(n,i)}$ – вес j -го входа

i -го нейрона η -го слоя сети; $\varphi_j^{(\eta,i)}$ – функция постсинаптического потенциала j -го входа i -го нейрона η -го слоя сети.

Функции активации нейронов будут определяться по формулам:

$$\psi^{(2,i)}\left(\varphi_j^{(2,i)}(w_j^{(2,i)}, x_j^{(2,i)})\right) = \min\{\varphi_j^{(2,i)}(w_j^{(2,i)}, x_j^{(2,i)})\},$$

$$i = 1, 2, \dots, Q; j = 1, 2, \dots, z;$$

$$\psi^{(3,i)}\left(\varphi_j^{(3,i)}(w_j^{(3,i)}, x_j^{(3,i)})\right) = \max\{\varphi_j^{(3,i)}(w_j^{(3,i)}, x_j^{(3,i)})\},$$

$$i = 1, 2, \dots, K; j = 1, 2, \dots, Q;$$

$$\psi^{(4,1)}\left(\varphi_j^{(4,1)}(w_j^{(4,1)}, x_j^{(4,1)})\right) = \arg \max\{\varphi_j^{(3,i)}(w_j^{(3,i)}, x_j^{(3,i)})\};$$

$$\psi^{(4,2)}\left(\varphi_j^{(4,2)}(w_j^{(4,2)}, x_j^{(4,2)})\right) = \max\{\varphi_j^{(4,2)}(w_j^{(4,2)}, x_j^{(4,2)})\},$$

где $\psi^{(\eta,i)}$ – функция активации i -го нейрона η -го слоя сети.

Весовые коэффициенты нейроэлементов сети будут устанавливаться по формуле:

$$w_j^{(\eta,i)} = \begin{cases} 1, \eta = 2, B_{i,p} \neq j, i = 1, 2, \dots, Q, \\ \quad p = 1, 2, \dots, N, j = 1, 2, \dots, z; \\ 0, \eta = 2, B_{i,p} = j, i = 1, 2, \dots, Q, \\ \quad p = 1, 2, \dots, N, j = 1, 2, \dots, z; \\ \alpha_j, \eta = 3, i = K_j, i = 1, 2, \dots, K, j = 1, 2, \dots, Q; \\ 0, \eta = 3, i \neq K_j, i = 1, 2, \dots, K, j = 1, 2, \dots, Q; \\ 1, \eta = 4, i = 1, 2, j = 1, 2, \dots, K. \end{cases}$$

Как видно из приведенных формул, синтез и настройка параметров нейро-нечеткой сети осуществляются в неитеративном режиме, что позволяет избежать необходимости расчета производных целевой функции по весам сети, а также итеративного характера коррекции весов, присущего традиционно применяемым градиентным методам обучения на основе техники обратного распространения ошибки.

5. Дообучение нейро-нечеткой сети

Ранее синтезированная нейро-нечеткая сеть может быть дообучена на основе следующего метода.

Шаг 1. Задать дополнительную выборку $\langle x^*, y^* \rangle$, $x^* = \{x^{g*}\}$, $g = 1, 2, \dots, G$ для дообучения, а также нейро-нечеткую сеть с настроенными параметрами функций принадлежности и весами. Сохранить в Q^* текущее значение Q , а в z^* значение z .

Шаг 2. Произвести распознавание экземпляров дополнительной выборки. Исключить из дополнительной выборки те экземпляры для которых расчетное значение выходного признака будет совпадать с целевым, скорректировать соответствующим образом G .

Шаг 3. Последовательно для каждого экземпляра дополнительной выборки x^{g*} , $g = 1, 2, \dots, G$ выполнять шаги 3.1 – 3.3.

Шаг 3.1 Сформировать термы по признакам и добавить в сеть соответствующие им нейроны на первый слой, а также задать соответствующие функции принадлежности к термам, для чего предлагается использовать модифицированные функции Гаусса:

$$\mu_{j,p}(x^s) = 2e^{-0.5(x_j^s - x_j^{s*})^2},$$

$$l_{Q,j} = x^{g*}, r_{Q,j} = x^{g*},$$

$$j = 1, 2, \dots, N, p = n_j + 1.$$

Шаг 3.2 Установить: $\forall j = 1, 2, \dots, N: n_j = n_j + 1$, $B_{Q,j} = z + j$; $Q = Q + 1$, $K_j = y^{g*}$, $z = z + N$.

Шаг 3.3 Добавить на второй слой сети нейрон, соответствующий блоку-кластеру для текущего экземпляра дополнительной выборки.

Шаг 4. Установить для новых нейронов второго слоя функции постсинаптического потенциала:

$$\varphi_j^{(2,i)}\left(w_j^{(2,i)}, x_j^{(2,i)}\right) = \max\{w_j^{(2,i)}, x_j^{(2,i)}\},$$

$$i = Q^* + 1, \dots, Q; j = 1, 2, \dots, z,$$

а также функции активации:

$$\psi^{(2,i)}\left(\varphi_j^{(2,i)}(w_j^{(2,i)}, x_j^{(2,i)})\right) = \min\{\varphi_j^{(2,i)}(w_j^{(2,i)}, x_j^{(2,i)})\},$$

$$i = Q^* + 1, \dots, Q; j = 1, 2, \dots, z.$$

Установить веса новых нейронов второго слоя по формуле:

$$w_j^{(\eta,i)} = \begin{cases} 1, \eta = 2, i = 1, 2, \dots, Q^*, j = z^* + 1, \dots, z; \\ 1, \eta = 2, i = Q^* + 1, \dots, Q, j = 1, 2, \dots, z^*; \\ 0, \eta = 2, B_{i,p} = j, i = Q^* + 1, \dots, Q, \\ \quad p = 1, 2, \dots, N, j = z^* + 1, \dots, z; \\ 1, \eta = 2, B_{i,p} \neq j, i = Q^* + 1, \dots, Q, \\ \quad p = 1, 2, \dots, N, j = z^* + 1, \dots, z. \end{cases}$$

Шаг 4. Для всех вновь введенных блоков-кластеров добавить на третий слой один нейрон для нормирования их принадлежностей к классам, для которого функцию постсинаптического потенциала определим, как:

$$\varphi_j^{(3,K+1)}\left(w_j^{(3,K+1)}, x_j^{(3,K+1)}\right) = \min\{w_j^{(3,K+1)}, x_j^{(3,K+1)}\},$$

$$j = 1, 2, \dots, Q,$$

а функцию активации – как:

$$\psi^{(3,K+1)}\left(\varphi_j^{(3,K+1)}(w_j^{(3,K+1)}, x_j^{(3,K+1)})\right) =$$

$$= -\frac{1}{2} \max\{\varphi_j^{(3,K+1)}(w_j^{(3,K+1)}, x_j^{(3,K+1)})\},$$

$$j = 1, 2, \dots, Q.$$

Для нейронов третьего и четвертого слоев установить значения весов новых связей:

$$w_j^{(\eta,i)} = \begin{cases} 1, \eta = 3, i = K_j, i = 1, 2, \dots, K, j = Q^* + 1, \dots, Q; \\ 0, \eta = 3, i \neq K_j, i = 1, 2, \dots, K, j = Q^* + 1, \dots, Q; \\ 0, \eta = 3, i = K + 1, j = 1, 2, \dots, Q^*; \\ 1, \eta = 3, i = K + 1, j = Q^* + 1, \dots, Q; \\ 0, \eta = 4, i = 1, j = K + 1; \\ 1, \eta = 4, i = 2, j = K + 1. \end{cases}$$

Шаг 5. Останов.

Предложенный метод дообучения не требует итеративной коррекции весов, не предполагает использования и хранения исходной обучающей выборки, не требует расчета производных целевой функции ошибки.

6. Упрощение нейро-нечеткой сети

Нейро-нечеткая сеть, синтезированная на основе предложенных методов, может быть упрощена за счет исключения проверок принадлежности к некоторым интервалам признаков при принятии решения об отнесении распознаваемого экземпляра к кластерам. Это достигается путем изменения весов:

$$w_j^{(\eta,i)} = \begin{cases} 1, \eta = 2, B_{i,p} \neq A_{i,p}, A_{i,p} \neq 0, i = 1, 2, \dots, Q^*, \\ p = 1, 2, \dots, N, j = 1, 2, \dots, z; \\ 0, \eta = 2, B_{i,p} = A_{i,p}, A_{i,p} \neq 0, i = 1, 2, \dots, Q^*, \\ p = 1, 2, \dots, N, j = 1, 2, \dots, z. \end{cases}$$

где

$$A_{q,j} = \begin{cases} p, B_{q,j} = p, \neg \exists v = 1, 2, \dots, Q^*, v \neq q: B_{v,j} = p; \\ \neg \exists u = 1, 2, \dots, Q^*, u \neq q: \\ (l_{u,j} < l_{q,j} \wedge r_{q,j} < r_{u,j}) \vee (r_{u,j} > l_{q,j} \wedge \\ \wedge (r_{u,j} < r_{q,j} \vee l_{u,j} < r_{q,j})); \\ \neg \exists A = Q^* + 1, \dots, Q, A \neq q: \\ l_{q,j} < l_{c,j}, r_{q,j} > r_{c,j}; \\ 0, \text{ в противном случае.} \end{cases}$$

Выводы

В работе решена актуальная задача разработки математического обеспечения для построения диагностических нейро-нечетких моделей по прецедентам.

Впервые предложен метод структурно-параметрической идентификации нейро-нечетких моделей, позволяющий в неитеративном режиме определить значения весовых коэффициентов, функции постсинаптического потенциала и активационные функции нейро-нечеткой сети на основе предварительно сформированного разбиения признакового пространства, что позволяет исключить необходимость вычисления производных целевой функции по весам сети и избежать проблемы выбора начальной точки поиска, что, в свою очередь, позволяет ускорить процесс получения нейро-нечеткой модели по прецедентам.

Получил дальнейшее развитие не рекуррентный метод потенциальных функций, который модифицирован для расчета потенциала, наводимого блоками с известной классификацией на блок-кластер с неизвестной классификацией, что позволило существенно расширить аппроксимационные и обобщающие свойства разбиения признакового пространства.

Работа выполнена в рамках госбюджетной темы “Научно-методические основы и математическое обеспечение для автоматизации и моделирования процессов управления и поддержки принятия решений на основе процедур распознавания и эволюционной оптимизации в нейросетевом и нечеткологическом базисах” (номер гос. регистрации 0106U008621).

Список литературы: 1. Суботин, С. О. Подання й обробка знань у системах штучного інтелекту та підтримки прийняття рішень [текст] : Навчальний посібник / С.О. Суботин. — Запоріжжя: ЗНТУ, 2008. — 341 с. 2. Леоненков, А.В. Нечеткое моделирование в среде MATLAB и fuzzyTECH [текст] / А. В. Леоненков. — СПб.: БХВ-Петербург, 2003. — 736 с. 3. Борисов, В.В. Нечеткие модели и сети [текст] / В. В. Борисов, В. В. Круглов, А. С. Федулов. — М.: Горячая линия-Телеком, 2007. — 284 с. 4. Субботин, С. А. Метод синтеза классифицирующих нейро-нечетких сетей с учетом значимости термов признаков [текст] / С. А. Субботин // Информационные технологии. — 2008. — № 7. — С. 31–33. 5. Субботин, С. А. Неитеративный синтез и редукция нейро-нечетких моделей [текст] / С. А. Субботин // Искусственный интеллект. — 2006. — № 3. — С. 323–330. 6. Айзерман, М. А. Метод потенциальных функций в теории обучения машин [текст] / М. А. Айзерман, Э. М. Браверман, Л. И. Розоноэр. — М.: Наука, 1970. — 384 с. 7. Фор, А. Восприятие и распознавание образов [текст] / Пер. с фр. А. В. Серединского / под ред. Г. П. Катуса. — М.: Машиностроение, 1989. — 272 с.

Поступила в редколлегию 30.10.2009

УДК 004.93

Структурно-параметрична ідентифікація нейро-нечітких моделей для діагностики / С.О. Субботін // Біоніка інтелекту: наук.-техн. журнал. — 2009. — № 2 (71). — С. 118-122.

Розроблено методи синтезу структури, донавчання та спрощення нейро-нечітких діагностичних моделей за прецедентами, що дозволяють у неітеративному режимі встановлювати значення ваг нейро-нечітких мереж, не потребують обчислення похідних і вибору початкової точки пошуку, що дозволяє прискорити процес одержання нейро-нечіткої моделі за прецедентами.

Лл. 1. Бібліогр.: 7 найм.

UDC 004.93

The neuro-fuzzy model structural-parametrical identification for diagnostics / Subbotin S.A. // Bionics of Intelligence: Sci. Mag. — 2009. — № 2 (71). — P. 118-122.

The methods of synthesis of a structure, retraining and simplification of neuro-fuzzy diagnostic model on precedents are developed. They permitting in a uniterated mode to evaluate a weights values of neuro-fuzzy networks, do not require an evaluation of derivatives and choice of a starting point of searching. This allows to speed up the neuro-fuzzy model building on precedents.

Fig.1. Ref.: 7 items.

УДК 681.32



СИНТЕЗ ПРОВЕРЯЮЩИХ ТЕСТОВ ДЛЯ ОДНОРОДНЫХ СХЕМ

Г.Г. Четвериков¹, Я.Ю. Королева², М.А. Бережная³

¹ ХНУРЭ, г. Харьков, Украина, chetvergg@kture.kharkov.ua;

² ХНУРЭ, г. Харьков, Украина;

³ ХНУРЭ, г. Харьков, Украина

Проведен анализ методов тестового диагностирования однородной структуры (ОС), определены множества отличительных и характеристических символов автоматных диаграмм ОС. Предложен способ нахождения по характеристическому дереву – модели ячейки сети, определен класс циклических отличительных последовательностей ОС и необходимые условия существования в данной ОС циклических отличительных последовательностей.

ОДНОРОДНАЯ СТРУКТУРА, ТЕСТОВОЕ ДИАГНОСТИРОВАНИЕ, ОТЛИЧИТЕЛЬНАЯ ПОСЛЕДОВАТЕЛЬНОСТЬ

Введение

Микропроцессорные СБИС, программируемые логические интегральные схемы (ПЛИС) типа FPGA и CPLD широко используются в разработках современных вычислительных и управляющих устройств, являются топологической основой для создания высокопроизводительных однородных структур или сетей (ОС) систолической обработки информации. Существует ряд работ, рассматривающих различные аспекты проблемы синтеза ОС по заданному алгоритму обработки информации [1–4]. Отличительной чертой такой обработки является однотипность структуры ячеек-модулей сети, выполняемых ею операций, связей между ячейками, рекурсивность вычислений.

Второй проблемой, возникающей при синтезе ОС, является сложность процедур верификации и тестового диагностирования сети [4–6]. На рис. 1 представлена структура одномерной ОС, в которой левый вход $z(1)$ и все вертикальные выходы $x(i)$, $i = \overline{1, p}$ являются управляемыми, а выходы $\hat{x}(i)$ и $\hat{z}(p)$ – наблюдаемыми.

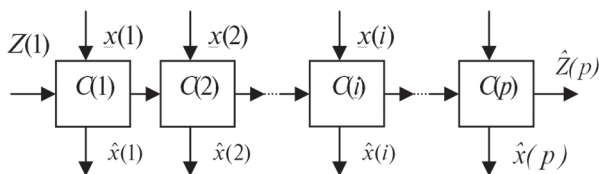


Рис. 1. Одномерная ОС с наблюдаемыми выходами \hat{x}

Различают два типа одномерных сетей: с наблюдаемыми выходами $\hat{x}(i)$ и без них.

В общем случае отсутствие наблюдаемых выходов \hat{x} в сети усложняет процедуру проверки ее исправности ввиду необходимости транспортировки множества неисправностей каждой ячейки сети на крайний правый выход сети.

Принципы построения проверяющих тестов на функциональном уровне исследованы в работах [3, 4], в которых ячейка ОС рассматривается как со-

вокупность четырех взаимодействующих каналов. Задача проверки исправности решается применением совокупности совместимых, сопряженных и самосопряженных тестов-наборов в некотором i -ом направлении распространения сигналов, где « i » изменяется по всем выходам ячейки сети. Получение требуемых тестовых наборов на входах проверяемой ячейки осуществляется путем настройки соответствующих каналов ОС. К сожалению, в этих работах не освещаются вопросы полноты проверяющего эксперимента, класс обнаруживаемых неисправностей и влияние структуры ячейки сети или ее функциональных характеристик на сложность построения и реализации диагностического эксперимента.

На уровне сети рассматриваются две модели неисправностей: 1) модель одиночной неисправности сети (допускается неисправной одна ячейка сети); 2) модель кратной неисправности сети (допускается неисправным произвольное множество ячеек сети). Первая модель представляет класс неисправностей F_1 , которые изменяют (искажают) таблицу переходов-выходов автоматной модели ячейки сети при ограничении: неисправность не изменяет числа состояний ячейки, является устойчивой на время прохождения проверяющего теста и допускается неисправной в момент проверки лишь одна произвольная ячейка сети. Класс неисправностей F_1 включает полное множество константных неисправностей ячейки, подкласс перемычек и коротких замыканий, перепутываний и инверсий, не увеличивающих числа состояний ячейки.

Вторая модель кратной неисправности ячеек сети представляет класс неисправностей F_k , когда при тех же ограничениях на изменения автоматной диаграммы ячейки сети, которые определены для класса F_1 , допускается неисправным произвольное множество ячеек сети.

Проблема тестового диагностирования одномерных ОС была исследована в работе [7], в ко-

торой были предложены методы тестирования неисправностей класса F_1 в ОС. В [8] определены необходимые и достаточные условия тестируемости одномерной однородной сети без наблюдаемых выходов \hat{x} относительно класса одиночных неисправностей ячеек сети. Показано, что сеть является тестируемой относительно класса неисправностей F_1 , если в таблице переходов ячейки множество последующих состояний содержит все состояния, и автоматная модель ячейки является минимальным автоматом Мура, то есть в таблице переходов нет двух одинаковых строк. Для нахождения проверяющих тестов сети в [8] был предложен подход, основанный на построении тестового графа и выделении множества фундаментальных циклов в тестовом графе сети. Однако предложенный подход не доведен до уровня алгоритмической завершенности и отсутствует оценка трудоемкости процедуры синтеза проверяющих тестов.

Недостатки существующих методов синтеза проверяющих тестов для ОС без наблюдаемых выходов \hat{x} можно исключить, если воспользоваться подходом, основанном на использовании характерных последовательностей автоматной модели ячейки сети и построении проверяющих тестов на основе циклических отличительных последовательностей ячейки ОС.

1. Отличительные последовательности в ОС

На функциональном уровне описания ячейки одномерной ОС без наблюдаемых выходов \hat{x} будем рассматривать ее таблицу истинности как таблицу переходов-выходов (ТПВ) автомата Мура, задаваемого тройкой (X, Z, δ) , у которого функции переходов и выходов совпадают, то есть $\delta(z_i, x_a) = \lambda(z_i, x_a) = \hat{z}_i, \forall z_i, z_j \in Z, \forall x_a \in X$.

Пусть $Z = \{z_1, z_2, \dots, z_i, \dots, z_n\}$ – множество состояний автоматной модели ячейки сети. Будем обозначать $Z/z_i = \{z_1, z_2, \dots, z_{i-1}, z_{i+1}, \dots, z_n\} = \{Z - z_i\}$ – множество состояний ячейки сети, из которого исключен элемент z_i . Как было показано в [8, 9], при синтезе проверяющих тестов сложность решения задачи анализа полноты полученных тестов и их расширения связана с наличием в автоматной модели ячейки сети пар x_α -совместимых состояний.

Определение 1. Если столбец x_α таблицы переходов ячейки сети содержит пару состояний (z_a, z_b) , для которых состояния-преемники эквивалентны, то есть $\delta(z_a, x_\alpha) = z_k$ и $\delta(z_b, x_\alpha) = z_k$, то состояния z_a и z_b являются x_α совместимыми или неразличимыми для входного символа x_α .

В общем случае может быть две альтернативы. Пара состояний (z_a, z_b) различается по меньшей мере одним входным символом x_β , либо пара $(z_a, z_b) - x_i$ -совместима для каждого входного символа $x_i \in X$. В первом случае состояния z_a и z_b можно идентифицировать только приложением ко входу проверяемой ячейки входного символа x_β .

Из табл. 1 переходов ячейки легко можно определить множества $z_c^{x_i} x_i$ – совместимых состояний для каждого входного символа $x_i \in X, i = \overline{1, m}$. Образуем множество $Z_c = \{Z_c^{x_1} \cap Z_c^{x_2} \cap \dots \cap Z_c^{x_m}\}$, элементы которого представляют собой состояния, x_α -совместимые для всех входных символов множества X .

Таблица 1

Таблица переходов ячейки

$z(t)$	$z(t+1)$		
	x_1	x_2	x_3
z_1	z_1	z_2	z_2
z_2	z_1	z_3	z_2
z_3	z_1	z_4	z_4
z_4	z_4	z_4	z_2

Определение 2. Пусть ячейка одномерной ОС без наблюдаемых выходов \hat{x} представлена моделью автомата Мура (X, Z, δ) , в котором x_α -преемником состояния $\forall z_i \in Z$ является состояние z_k , не обязательно отличное от z_i . Входной символ $x_\alpha \in X$ будем называть отличительным символом состояния z_i тогда и только тогда, когда z_k не является x_α -преемником для множества Z/z_i начальных состояний автомата и $z_k \notin z_c$.

Для различения состояний (z_a, z_b) можно воспользоваться множеством характеристических входных символов подобно тому, как в [10] использовались характеристические последовательности при построении диагностических экспериментов для автоматов, не имеющих отличительных последовательностей.

Определение 3. Пусть ячейка одномерной сети представлена моделью автомата Мура (X, Z, δ) . Множество входных символов $X_c = \{x_1, x_2, \dots, x_r\}$ называется множеством входных характеристических символов тогда и только тогда, когда для любой пары состояний $(z_a, z_b) \in Z$ автомата $\delta(z_a, x_1)\delta(z_a, x_2) \dots \delta(z_a, x_r) \neq \delta(z_b, x_1)\delta(z_b, x_2) \dots \delta(z_b, x_r)$.

Множество отличительных и характеристических символов может быть найдено из характеристического дерева автомата ячейки сети, которое строится по правилам, приведенным в [10].

Характеристическое дерево ячейки сети, представленной автоматной диаграммой в табл. 1, приведено на рис. 2, из которого можно найти множество отличительных и характеристических символов.

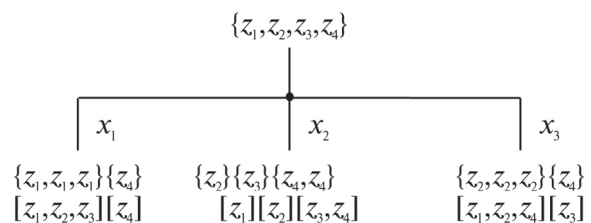


Рис. 2. Характеристическое дерево ячейки

Так как произведение разбиений

$$\pi(x_1) = (\overline{z_1 z_2 z_3}, \overline{z_4}) \text{ и } \pi(x_2) = (\overline{z_1}, \overline{z_2}, \overline{z_3 z_4})$$

равно $\pi(0)$, то множество $X_c = \{x_1, x_2\}$ является множеством входных характеристических символов. Простые σ -множества для каждой вершины характеристического дерева определяют множество состояний, для которых метка вершины дерева является отличительным входным символом. Исключения составляют лишь те простые σ -множества, которые являются состояниями преемниками, принадлежащими множеству Z_c , то есть множеству состояний, неразличимых для всех входных символов.

Например из характеристического дерева рис. 2 ячейки сети (табл. 1) следует, что $z_c^{x_1} = \{z_1, z_2, z_3\}$, $z_c^{x_2} = \{z_3, z_4\}$, $z_c^{x_3} = \{z_1, z_2, z_4\}$. Множества x_α -совместимых состояний ячейки позволяют определить множество Z_c в виде: $Z_c = Z_c^{x_1} \cap Z_c^{x_2} \cap Z_c^{x_3} = \emptyset$.

Определив множество Z_c , из характеристического дерева можно легко найти множество отличительных символов и состояний ячейки, которые различаются этими символами в соответствии с определением 2. Для рассматриваемого примера состояние z_4 различается символом E_1 , состояния z_1 и z_2 – символом E_2 , состояние z_3 – символом E_3 .

Определение 4. Пусть состояние z_j правого выхода ячейки $C(k)$ одномерной ОС транспортируется на крайний правый выход сети приложением входного набора X_T ко входам ячеек $C(k+1), C(k+2), \dots, C(p)$. Входной набор X_T является проверяющим тестом, идентифицирующим состояние z_j ячейки $C(k)$, если состояние наблюдаемых выходов сети различно, когда z_j и каждое из состояний множества Z/z_j приложено к левому входу ячейки $C(k+1)$, то есть $\delta(z_j, X_T) \neq \delta(z_i, X_T)$, $\forall z_i \in Z/z_j$.

Если для каждого состояния ячейки сети существует проверяющий входной набор X_T , то проверяющий тест всей ОС можно построить путем проверки правильности всех $m \cdot n$ переходов автоматной модели каждой ячейки сети, используя проверяющие тестовые наборы X_T для идентификации правильности переходов.

Нижеследующая теорема определяет необходимые и достаточные условия существования в одномерной сети проверяющих тестовых наборов X_T .

Теорема 1. Пусть правый выход ячейки $C(k)$ одномерной однородной сети находится в состоянии z_j , икверхним входам ячеек $C(k+1), C(k+2), \dots, C(p)$ приложен входной набор $X_k = (E_{k+1}, E_{k+2}, \dots, E_p)$, вызывающий последовательность переходов состояний ячеек сети в виде $z_j \xrightarrow{E_{k+1}} z_{k+1} \xrightarrow{E_{k+2}} z_{k+2} \xrightarrow{E_{k+3}} \dots \xrightarrow{E_{p-1}} z_p$, где состояние слева от входного символа $E_j \in X_k$, $\alpha = (k+1)(k+2) \dots$ является предшеству-

ющим, а состояние справа от x_α – последующим. Если существует последовательность состояний ячеек $C(k+1), C(k+2), \dots, C(p)$, порождаемая приложением входного набора X_k , в котором каждый символ $E_\alpha \in X_k$ является отличительным символом предшествующего состояния, то последовательность этих символов образует проверяющий тестовый набор состояния z_j ячейки $C(k)$ сети.

Доказательство. Необходимость. По условию теоремы символ x_{k+1} является отличительным для состояния z_j , x_{k+2} – для z_{j+1} и так далее. Пусть в результате воздействия неисправности на правом выходе ячейки $C(k)$ вместо состояния z_j появилось состояние z_i . Так как X_{k+1} является отличительным символом состояния z_j , то в соответствии с определением 2 для каждого $z_i \in Z/z_j$. Следовательно, при наличии неисправности $z_j \rightarrow z_i$ на правом выходе ячейки $C(k+1)$ появится состояние $z_{k+1}^* \neq z_{k+1}$. В свою очередь, так как x_{k+2} – отличительный символ состояния z_{k+1} , то $\delta(z_{k+1}, x_{k+2}) \neq \delta(z_{k+1}^*, x_{k+2})$ и так далее для каждой ячейки сети. В результате на наблюдаемых выходах ячейки $C(p)$ имеем $\delta(z_{p-1}^*, x_p) \neq \delta(z_{p-1}, x_p)$. Следовательно, состояние наблюдаемого выхода сети отличает состояние z_j ячейки $C(k)$ от любого другого состояния, и в соответствии с определением 4 входной набор $z_i \in Z/z_j$ является тестовым набором, обнаруживающим неисправность типа $z_j \rightarrow Z/z_j$ в ячейке $C(k)$.

Достаточность. Пусть входной набор x_k , приложенный ко входам ячеек $C(k+1), C(k+2), \dots, C(p)$ сети является проверяющим тестовым набором, обнаруживающим любую неисправность ячейки $C(k)$ вида $z_j \rightarrow Z/z_j$. Предположим, что входной символ x_{k+1} , приложенный ко входу ячейки $C(k+1)$, не является отличительным символом предшествующего состояния z_j ячейки $C(k)$. В соответствии с определением 3 существует по меньшей мере одно состояние z_α , которое совместимо с z_j , то есть $\delta(z_j, x_{k+1}) = \delta(z_\alpha, x_{k+1}) = z_{k+1}$. Поэтому неисправность типа $z_j \rightarrow z_i$ не будет транспортироваться на наблюдаемый выход сети. Следовательно, символ X_{k+1} должен быть отличительным символом состояния z_j . Рассуждая аналогично, нетрудно убедиться в том, что этому условию должен удовлетворять каждый символ $x_\alpha \notin x_k$.

2. Синтез тестов на основе циклических отличительных последовательностей

Задача построения проверяющих входных наборов для заданной одномерной ОС значительно упрощается, если существуют циклические переводящие последовательности, образованные из отличительных входных символов. При построении диагностических экспериментов по автоматным моделям дискретных устройств (ДУ) возникает задача нахождения множества переводящих после-

довательностей $T(z_i, z_j)$, с помощью которых ДУ из начального состояния z_i переводится в состояние z_j . Известно, что эта задача тривиальна, если ДУ задано графом или таблицей переходов [10], переводящая последовательность $T(z_i, z_j)$ определяется путем построения фрагментов дерева преемников состояния z_i , содержащего путь или множество путей, оканчивающихся состоянием z_j .

При построении проверяющих тестов для одномерных ОС возникает аналогичная задача нахождения множества переводящих последовательностей по автоматной модели ячейки сети. Тестопригодность сети, число ячеек которой превышает число состояний автоматной модели ячейки сети, определяется наличием в ней циклических переводящих последовательностей $T(z_i, z_j)$, которые так же, как и любые другие переводящие последовательности, можно легко найти из автоматной модели ячейки сети.

Определение 5. Циклическую переводящую последовательность, образованную из входных отличительных символов предшествующих состояний ячеек сети, будем называть циклической отличительной последовательностью (ЦОП).

В соответствии с теоремой 1 циклическая переводящая последовательность $T(z_i, z_j)$, которая образована из отличительных входных символов, является одновременно проверяющим тестовым набором, который транспортирует состояние проверяемой ячейки сети на наблюдаемый выход.

Рассмотрим класс однородных одномерных сетей без наблюдаемых выходов \hat{x} , состоящих из p ячеек, каждая из которых имеет m входных символов, n состояний и каждое состояние имеет по меньшей мере один отличительный символ. Для сетей такого типа справедливо следующее утверждение.

Теорема 2. Если в ячейке ОС с n состояниями каждое состояние имеет по крайней мере один отличительный символ, то для такой сети существует по меньшей мере одна циклическая отличительная последовательность.

Доказательство. Циклическая последовательность переходов состояний автомата отсутствует, когда состояния-преемники не совпадают с состояниями-предшественниками, то есть когда выполняется условие $\delta(z_i, x_i) = z_j, z_i \neq z_j, \forall x_i \in X (i = 1, m), \forall z_i, z_j \in Z (i, j = 1, n)$. Не нарушая общности рассуждений, представим последовательность переходов состояний, удовлетворяющую этому условию, в следующем виде:

$$z_1 \xrightarrow{E_1} z_2 \xrightarrow{E_2} z_3 \dots z_i \xrightarrow{E_i} z_j \dots \xrightarrow{E_{n-1}} z_n.$$

В этой последовательности нет циклов. Однако по условию теоремы каждое состояние имеет от-

личительный символ, в том числе и последнее состояние цепи z_n . Пусть x_n — отличительный символ z_n . Так как рассмотренная цепь включает все n состояний, то $\delta(z_n, x_n) = z_i$, где индекс « i » может принимать любое значение от 1 до n . Следовательно, существует переход в одно из предшествующих вершин цепи, а значит, существует по меньшей мере один цикл, образованный отличительными символами.

В качестве примера рассмотрим автомат, заданный табл. 1. Из характеристического дерева автомата (рис. 2) находим отличительные символы состояний: $z_1 - z_2; z_2 - z_2; z_3 - z_3; z_4 - z_1$.

Теорема 2 определяет необходимые условия существования в ячейке ОС циклической отличительной последовательности. Существование ЦОП для заданной однородной сети упрощает процедуру построения проверяющего эксперимента для этой сети так же, как и наличие отличительных последовательностей для автоматных моделей произвольных ДУ с элементами памяти. Такие ОС являются легко тестируемыми.

Выводы

На основе анализа методов тестового диагностирования ОС предложен единый методологический подход к решению этой проблемы, основанный на функциональном принципе описания поведения ОС, функциональных моделях неисправностей и использовании методов теории экспериментов с автоматами. Определены множества отличительных и характеристических символов автоматных диаграмм ОС, предложен способ их нахождения по характеристическому дереву — модели ячейки сети. ОС, ячейка которой для каждого состояния имеет отличительный символ, является легко тестируемой. Определены необходимые и достаточные условия существования в однородной сети проверяющих тестовых наборов, которые зависят от свойств автоматной модели ячейки сети и существовании последовательностей состояний, имеющих отличительные символы. Определен класс циклических отличительных последовательностей ОС и необходимые условия существования в данной ОС циклических отличительных последовательностей, формируемых из отличительных символов предшествующих состояний ячеек сети, которые позволяют формализовать и упростить процедуру и построения проверяющего эксперимента ОС.

Список литературы: 1. Тоффоли Т. Машины клеточных автоматов [Текст] / Т. Тоффоли, Н. Марголус. — М.: Мир, 1991. — 280 с. 2. Варшавский В.И. Однородные структуры. Анализ. Синтез. Поведение. [Текст] / В.И. Варшавский, В.Б. Мараховский. — М.: Энергия, 1973. — 152 с. 3. Синтез

управляющих устройств в однородных средах [Текст] / И.С. Визирев, В.Ф. Гузик и др. — М.: Наука, 1984. — 166 с.

4. Евренков Э.В. Цифровые автоматы с настраиваемой структурой [Текст] / Э.В. Евренков, И.В. Прангишвили. — М.: Энергия, 1974. — 240 с.

5. Чараев В.Г. Контроль исправности и диагностика неисправностей однородной двумерной структуры [Текст] / В.Г. Чараев // Автоматика и телемеханика. — 1968. — №7. — С. 45-52.

6. Cheng W.T. Multiple fault-detection in iterative logic arrays / W.T. Cheng, J.N. Patel // Proc. 1985 Int. Test Conference. — Nov., 1985. — P.493-499.

7. Kautz, W.H. Testing for faults in cellular logic arrays / W.H. Kautz // Proc. 8-th Annual Symp. Switching and Automata Theory. — 1967. — P. 161-174.

8. Friedman, A.D. Fault detection in digital circuits / A.D. Friedman, P.R. Menon — New Jersey: Prentice Hall, 1971. — P.220.

9. Фридман А. Теория и проектирование переключательных схем [Текст] / А. Фридман, Д. Менон. — М.: Мир, 1978. — 589 с.

10. Тоценко В.Г. Алгоритмы технического диагностирования дискретных устройств [Текст] / В.Г. Тоценко. — М.: Радио и связь, 1983. — 240 с.

Поступила в редколлегию 14.10.2009

УДК 681.32

Синтез комбінацийних сем у базисі поліноміальних форм / Г.Г. Четверіков, Я.Ю. Корольова, М.А. Бережна // Біоніка інтелекту: наук.-техн. журнал. — 2009. — № 2 (71). — С. 123-127.

Представлено метод синтезу перевіряючих тестів для класу одномірних ОС, у яких відсутні виходи \hat{x} , оснований на використанні автоматних моделей чарунок ОС і функціональних моделях пошкоджень чарунок мережі. Запропоновано і обґрунтовано використання циклічних і відрізняючих послідовностей чарунки мережі для побудови діагностичного експеримента в ОС.

Л. 2. Бібліогр.: 10 найм.

UDK 681.32

Synthesis of combinational circuits in polynomial forms BASIS / G.G. Chetverikov, Y.U. Koroleva, M.A. Beregnaya // Bionics of Intelligence: Sci. Mag. — 2009. — № 2 (71). — P. 123-127.

The test method of one-dimensional logic arrays (ILAs), composed of identical cells, are considered. The fault model assumed is that faults in single cell can change a cell behavior in any arbitrary way. The method is based on finding cyclic distinguishing sequences of automation model of the ILA cell.

Fig. 2. Ref.: 10 items.

ОБ АВТОРАХ

Бережная Марина Анатольевна	123	доцент кафедры технологий и автоматизации производства Харьковского национального университета радиоэлектроники
Бержанский Владимир Наумович	48	д-р физ.-мат. наук, профессор кафедры экспериментальной физики, проректор по научной работе Таврического национального университета им. В.И.Вернадского г. Симферополь
Бондаренко Михаил Федорович	5, 13, 24	член-корреспондент НАН Украины, д-р техн. наук, профессор, ректор Харьковского национального университета радиоэлектроники
Вишняк Михаил Юрьевич	93	канд. техн. наук, доцент кафедры системотехники Харьковского национального университета радиоэлектроники
Вечирская Ирина Дмитриевна	31	канд. техн. наук, старший научный сотрудник кафедры программного обеспечения ЭВМ Харьковского национального университета радиоэлектроники
Дикарева Светлана Самуиловна	48	канд. филол. наук, доцент кафедры межъязыковых коммуникаций Таврического национального университета им. В.И.Вернадского, г. Симферополь
Довгань Светлана Сергеевна	93	аспирант кафедры системотехники Харьковского национального университета радиоэлектроники
Емцев Евгений Дмитриевич	70	студент факультета компьютерных наук Харьковского национального университета радиоэлектроники
Ерохин Андрей Леонидович	66	д-р техн. наук, профессор, начальник кафедры информатики Харьковского национального университета внутренних дел
Заболотняя Татьяна Николаевна	54	канд. техн. наук, старший преподаватель кафедры программного обеспечения компьютерных систем Национального технического университета Украины «КПИ», г. Киев
Каменева Ирина Витальевна	75	аспирант кафедры программного обеспечения ЭВМ Харьковского национального университета радиоэлектроники, инженер отдела практики «Центр-Карьера»
Келеберда Игорь Николаевич	70, 79	канд. техн. наук, доцент кафедры программного обеспечения ЭВМ Харьковского национального университета радиоэлектроники
Кириченко Людмила Олеговна	111	канд. техн. наук, доцент кафедры прикладной математики Харьковского национального университета радиоэлектроники
Клименко Алексей Викторович	99	канд. физ.-мат. наук, старший научный сотрудник, докторант Харьковского национального университета радиоэлектроники
Кольченко Алексей Вячеславович	66	аспирант кафедры программного обеспечения ЭВМ Харьковского национального университета радиоэлектроники
Королева Яна Юрьевна	123	ассистент кафедры технологий и автоматизации производства Харьковского национального университета радиоэлектроники

Кротких Семен Сергеевич	111	аспирант кафедры прикладной математики Харьковского национального университета радиоэлектроники
Купин Олег Николаевич	79	студент факультета компьютерных наук Харьковского национального университета радиоэлектроники
Лесная Наталия Советовна	70, 79	д-р техн. наук, профессор кафедры программного обеспечения ЭВМ Харьковского национального университета радиоэлектроники
Марченко Михаил Михайлович	75	студент факультета компьютерных наук Харьковского национального университета радиоэлектроники
Олейник Александр Игоревич	106	канд. техн. наук, старший научный сотрудник, докторант Харьковского национального университета радиоэлектроники
Олейник Алексей Александрович	85	ассистент кафедры программных средств Запорожского национального технического университета
Остапова Ирина Вадимовна	40	научный сотрудник Украинского языково-информационного фонда НАН Украины, г. Киев
Полулях Сергей Николаевич	48	д-р физ.-мат. наук, доцент кафедры экспериментальной физики Таврического национального университета им. В.И.Вернадского, г. Симферополь
Потапова Елена Владимировна	48	аспирант Украинского языково-информационного фонда НАН Украины, младший научный сотрудник научно-исследовательской части Таврического национального университета им. В.И.Вернадского, г. Симферополь
Репка Виктория Борисовна	75	канд. техн. наук, доцент кафедры программного обеспечения ЭВМ Харьковского национального университета радиоэлектроники
Субботин Сергей Александрович	85	канд. техн. наук, доцент кафедры программных средств Запорожского национального технического университета
Хайрова Нина Феликсовна	36	канд. техн. наук, доцент кафедры информационных технологий и математики Харьковского гуманитарного университета “Народная украинская академия”
Четвериков Григорий Григорьевич	123	д-р техн. наук, профессор кафедры программного обеспечения ЭВМ Харьковского национального университета радиоэлектроники
Шабанов-Кушнарченко Сергей Юрьевич	5, 13, 24	д-р техн. наук, профессор кафедры программного обеспечения ЭВМ Харьковского национального университета радиоэлектроники
Шабанов-Кушнарченко Юрий Петрович	5, 13, 24	д-р техн. наук, профессор кафедры программного обеспечения ЭВМ Харьковского национального университета радиоэлектроники
Шарий Тимофей Вячеславович	61	аспирант кафедры компьютерных технологий Донецкого национального университета
Шатовская Татьяна Борисовна	75	канд. техн. наук, доцент кафедры программного обеспечения ЭВМ Харьковского национального университета радиоэлектроники, начальник отдела практики «Центр-Карьера»

ПРАВИЛА ОФОРМЛЕНИЯ РУКОПИСЕЙ ДЛЯ АВТОРОВ НАУЧНО-ТЕХНИЧЕСКОГО ЖУРНАЛА «БИОНИКА ИНТЕЛЛЕКТА»

Научно-технический журнал «Бионика интеллекта» принимает для опубликования написанные специально для него оригинальные рукописи, которые нигде раньше не издавались. Структура рукописи должна быть следующей: индекс УДК, заглавие, сведения об авторах, аннотация, ключевые слова, введение, основной текст статьи, выводы, список использованной литературы.

Согласно Постановлению ВАК Украины от 15.01.2003 №7-05/1 (Бюллетень ВАК, №1, 2003, с. 2) статья должна иметь следующие необходимые элементы: постановка проблемы в общем виде и ее связь с важными научными или практическими заданиями; анализ последних исследований, публикаций и выделение нерешенных ранее частей общей проблемы в данной области; формулирование целей и задач исследования; изложение основного материала исследований с полным обоснованием полученных научных результатов; выводы по данному исследованию и перспективы дальнейших исследований в данном направлении.

Статьи должны быть выполнены в редакторе Microsoft Word. Формат страницы — А4 (210 x 297 мм), поля: верхнее — 25 мм, нижнее — 20 мм, левое, правое — 17 мм. Количество колонок — 2, с интервалом между ними 5 мм, основной шрифт Times New Roman, кегль основного текста — 10 пунктов, междустрочный интервал — множитель (1,1), абзацный отступ — 6 мм. Объем рукописи — от 4 до 12 страниц (языки: русский, украинский, английский).

УДК печатается с первой строки, без отступа, выравнивание по левому краю.

Название рукописи печатается прописными буквами; шрифт прямой, полужирный, кегль 12.

Название разделов нумеруют арабскими цифрами, выделяют жирным шрифтом. Отступы для названия рукописи, инициалов и фамилий авторов, сведений об авторах, названий разделов, введения и выводов, списка литературы сверху — 6 пт, снизу — 3 пт.

Аннотация (на языке рукописи, абзац 4–10 строк, кегль 9) помещается в начале статьи и содержит информацию о результатах описанных исследований.

Ключевые слова (4–10 слов из текста статьи, которые с точки зрения информационного поиска несут смысловую нагрузку, на языке рукописи, через запятую в именительном падеже, кегль 9).

Рисунки и таблицы (черно-белые, контрастные) помещаются в текст после первой ссылки в виде встроенных объектов и раздельно нумеруются арабскими цифрами сквозной нумерацией, при наличии более одного объекта. Простые схемы, состоящие из 3–4 элементов, рисуют, используя вставку объекта Рисунок Microsoft Word. Более сложные выполняют в графических редакторах в виде черно-белых графических файлов форматов .tiff, .jpg, .wmf, .cdr с разрешением 300 dpi. Рисунки должны быть включены в текстовый файл, а также обязательно представлены отдельным файлом с соответствующим названием (например, ris_1.cdr).

Все элементы рисунка, включая надписи, должны быть сгруппированы. Все надписи в рисунках и таблицах должны быть выполнены шрифтом Times New Roman, кегль в рисунках — 10, в таблицах — 9. Табличный заголовок располагается справа над таблицей (кегель 9 пт).

Рисунок содержит подрисовочную центрированную подпись (вне рисунка), кегль 9, по центру, отступы сверху и снизу по 6 пт. Ширина рисунка должна соответствовать ширине колонки (или ширине страницы).

Формулы, символы, переменные должны быть набраны в редакторе формул MathType или Microsoft Equation. Формулы располагаются по центру и нумеруются при наличии ссылки на них в рукописи. Шрифт — Times New Roman. Высота переменной — 10 пунктов, индексов и подиндексов — 8 пт, основной математический символ — 12 (10) пт. Переменные, обозначенные латинскими буквами, набирают курсивом, греческие буквы, сокращения русских слов и цифры — прямым начертанием. Переменные, содержащиеся в тексте, также набирают в редакторе формул.

Небольшие формулы можно размещать не в отдельной строке, а прямо по тексту.

Список литературы включает опубликованные источники, на которые имеются ссылки в тексте, заключенные в квадратные скобки, печатается без абзацного отступа, кегль 9 пунктов, отступ сверху — 6 пт.

После списка литературы с отступами сверху — 6 пт указывается дата поступления статьи в редколлегию. Число и месяц задаются двузначными числами через точку. Размер шрифта — 9 пт, курсив, выравнивание по правому краю.

Рефераты (Times New Roman, кегль — 9 пунктов, 3–4 предложения) на украинском и английском языках. Реферат не должен дублировать текст аннотации.

Вместе с рукописью (на листах белой бумаги формата А4 плотностью 80–90 г/м², напечатанной на лазерном принтере, в 2-х экземплярах) необходимо подать следующие документы:

1. Заявку утвержденного образца, подписанную всеми авторами:

«Прошу принять статью ... (указать фамилии авторов, название статьи) объемом ... страниц (указать количество страниц) для опубликования в научно-техническом журнале «Бионика интеллекта». Оплату гарантируем.

Сведения об авторах: (указать фамилию, имя, отчество каждого автора, место работы, ученую степень, научное звание, контактный телефон, почтовый и электронный адреса).

Подписи авторов».

2. Акт экспертизы о возможности опубликования материалов в открытой печати.

3. Рецензию, подписанную доктором наук.

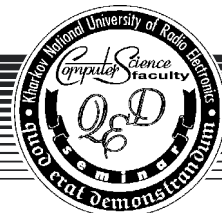
4. Сведения об авторах.

5. Электронный вариант рукописи, реферата и сведений об авторах.

6. Оплату за публикацию.

ИНФОРМАЦИОННОЕ СООБЩЕНИЕ

Факультет компьютерных наук ХНУРЭ
Ежемесячный научный семинар Q.E.D.



Если в рамках научных направлений факультета КН Вы

хотите обсудить в «полновесном» формате (до полутора часов):

- «Новый результат в ...»
- «Состояние и перспективы развития ...»
- «Сравнительный анализ подходов (методов...) ...»

сообщить в формате «лягушатника» (до 20 минут):

- обоснование темы диссертации
- цели и задачи диссертационных исследований
- теоретическую значимость и практическую ценность результатов
- полученные теоретические результаты и их новизну
- представить монографию, статью, доклад ...

приглашаем принять участие в работе семинара Q.E.D.

Постоянные члены семинара:

проф. Авраменко В.П.
проф. Бескорвайный В.В.
проф. Бодянский Е.В.
проф. Бондаренко М.Ф.
проф. Борячок М.Д.
проф. Гребенник И.В.
проф. Дударь З.В.
проф. Кучеренко Е.И.
проф. Левыкин В.М.
проф. Лесная Н.С.
проф. Машгалир В.П. (зам. председателя)
проф. Петров Э.Г. (председатель)
доц. Рябова Н.В.
проф. Филатов В.А.
доц. Чалая Л.Э. (ученый секретарь)
проф. Четвериков Г.Г.
проф. Шабанов-Кушнаренко Ю.П.

*Семинар проводится
каждый второй четверг
в 15⁰⁵, ауд. 259*

СЛЕДИТЕ ЗА ОБЪЯВЛЕНИЯМИ!

Запись для выступлений на семинаре
у секретаря (комн. 370)

ПРИГЛАШАЮТСЯ ВСЕ ЖЕЛАЮЩИЕ

Харьковский национальный университет
радиоэлектроники, г. Харьков, просп. Ленина, 14

Наукове видання

БІОНІКА ІНТЕЛЕКТУ
інформація, мова, інтелект

Науково-технічний журнал

№ 2 (71)

2009

Головний редактор — *М. Ф. БОНДАРЕНКО*

Відповідальний редактор — *Ю. П. Шабанов-Кушнарєнко*

Заступник відповідального редактора — *Г. Г. Четвериков*

Відповідальний секретар — *І. Д. Вечірська*

Коректор — *Л. М. Денісова*

Комп'ютерна верстка — *О. Б. Ісаєва*

Рекомендовано Вченою Радою
Харківського національного університету радіоелектроніки
(протокол № 56 від 30.10.2009)

Адреса редакції:

Україна, 61166, Харків-166, просп. Леніна, 14,
Харківський національний університет радіоелектроніки, к. 127, 285
тел. 702-14-77, факс 702-10-13,
e-mail: bionics@kture.kharkov.ua

Підписано до друку 30.10.2009. Формат 60 × 84 ¹/₈. Друк ризографічний.
Папір офсетний. Гарнітура Newton. Умов. друк. арк. 15,4. Обл.-вид. арк. 15,0.
Тираж 300 прим. Зам. № .

Надруковано в навчально-науковому видавничо-поліграфічному центрі ХНУРЕ
61166, Харків-166, просп. Леніна, 14