МИНИСТЕРСТВО ОБРАЗОВАНИЯ УКРАИНЫ

ХАРЬКОВСКИЙ ГОСУДАРСТВЕННЫЙ
ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ РАДИОЭЛЕКТРОНИКИ

ПРОБЛЕМЫ БИОНИКИ

Всеукраинский межведомственный
научно-технический сборник

Основан в 1968 г.

ВЫПУСК 49

Харків
Харківський державний технічний
університет радіоелектроніки

1998

УДК 519.7

В сборнике рассмотрен широкий круг проблем бионики интеллекта. Представлены результаты моделирования процесса обработки словоформ и морфологических отношений, формализации нечетких и противоречивых знаний. Описаны различные алгебры предикатных операций. Сопоставлены системы переработки информации компьютером и человеком.

Для преподавателей вузов, научных работников и специалистов.

У збірнику розглянуто широке коло проблем біоніки інтелекту. Подано результати моделювання процесу обробки словоформ та морфологічних відношень, формалізації нечітких та суперечливих знань. Описано різні алгебри предикатних операцій. Зіставлено системи переробки інформації комп'ютером і людиною.

Для викладачів вищих закладів освіти, науковців і фахівців.

A wide range of problems concerning the intelligence bionics is considered in this issue. Results of the modelling of processing word-forms and morphological relations, of formalizing fuzzy and contradictory knowledge. Systems of processing information by man and computer are compared.

The issue is intended for university teachers, research workers and specialists.

Ответственный за выпуск канд. техн. наук *В.А. Чикина*

Сборник включен в список специальных изданий ВАК Украины по физико-математическим и техническим наукам

Адрес редакционной коллегии: Украина, 310726, Харьков-726, просп. Ленина, 14, Харьковский государственный технический университет радиоэлектроники (ХТУРЭ), тел. 40-94-46

Проблемы бионики
1998. Вып. 49. С. 3—13

ХТУРЭ

УДК 519.7

З.В. ДУДАРЬ, Н.С. КРАВЕЦ, Ю.П. ШАБАНОВ-КУШНАРЕНКО

О ФУНДАМЕНТАЛЬНОЙ АЛГЕБРЕ ПРЕДИКАТНЫХ ОПЕРАЦИЙ*

Предикатные операции используются в тех случаях, когда возникает необходимость производить действия над предикатами [1] или описывать связи между ними [2]. Таблицы, содержащиеся в базах данных, удобно формально понимать как некоторые предикаты, а действия над ними - как предикатные операции. Связи между таблицами можно понимать как предикаты второго порядка, т. е. как специальные предикатные операции, значениями которых служат предикаты 0 и 1. Алгебры предикатных операций полезно использовать при проектировании систем обработки информации, различных информационных структур и их электронных схем. При изучении механизмов интеллекта человека алгебры предикатных операций целесообразно использовать для формульной записи свойств (законов) интеллектуального поведения испытуемого. Само же это поведение удачно описывается на языке алгебры предикатов. С помощью формул алгебр предикатных операций изящно и просто выражается смысловая структура предложений и текстов естественного языка [3]. Мысли можно формально представлять в виде предикатов, а для выражения действий над ними использовать предикатные операции.

Учение о предикатных операциях соотносится с учением о предикатах примерно так же, как математический анализ - со школьной алгеброй. В школьной алгебре изучаются числовые функции, а в математическом анализе - операции над ними. Алгебра предикатных операций представляет собой как бы второй этаж, надстройку над алгеброй предикатов, ее расширение. Алгебра предикатов охватывается алгеброй предикатных операций. Школьная алгебра и математический анализ относятся к области числовой математики, которая развивалась преимущественно применительно к потребностям описания объектов и процессов, наблюдаемых во внешнем (объективном, физическом) мире. Алгебра же предикатов и алгебра предикатных операций относятся к области *логической математики*, которая возникла в связи с необходимостью формального описания объектов и процессов, наблюдаемых во внутреннем (субъективном, психологическом) мире

* Статья публикуется в авторской редакции.

человека и машины. Основная сфера приложения логической математики - формальное описание информационных объектов и процессов. Мы ожидаем, что со временем учение о предикатных операциях станет главным математическим инструментом информатики.

Пусть U - универсум предметов; $x_1, x_2, \dots, x_m$ - предметные переменные; М - множество всех предикатов $P(x_1, x_2, \dots, x_m)$ на предметном пространстве U^m [1]. Множество М называется *универсумом предикатов*. Переменные $X_1, X_2, \dots, X_m$, определенные на множестве М, называются *предикатными переменными*. Их значениями служат предикаты, заданные на U^m. Множество M^n называется *предикатным пространством размерности n* над предметным пространством U^m. Элементы множества M^n (наборы предикатов) называются *предикатными векторами*. Предикатное пространство представляет собой двухэтажную конструкцию: на первом этаже находятся предметы, на втором - предикаты. Любая функция $F(X_1, X_2, \dots, X_n)=Y$, отображающая множество M^n в множество М, называется *предикатной операцией*. Образуем множество R всех предикатных операций. *Алгеброй предикатных операций* над R называется любая алгебра, заданная на носителе R. В классическом математическом анализе предметам алгебры предикатных операций соответствуют числа, предикатам - функции, предикатным операциям - операторы, отношениям второго порядка, связывающим предикаты, - функциональные уравнения (в частности, дифференциальные и интегральные).

Пусть $F(X_1, X_2, \dots, X_n)=Y$ - предикатная операция, отображающая множество M^n в множество М. Здесь $X_1, X_2, \dots, X_n$ - предикатные переменные, выступающие в роли аргументов операции F; Y - предикатная переменная, являющаяся значением операции F. *Отрицанием* $\neg F = \bar{F}$ *предикатной операции* F называется такая предикатная операция, значения которой определяются по правилу

$$(\neg F)(X_1, X_2, \dots, X_n)=\neg F(X_1, X_2, \dots, X_n) \qquad (1)$$

для любых $X_1, X_2, \dots, X_n \in M$. Пусть F и G - предикатные операции, отображающие M^n в М. *Дизъюнкцией* $F \vee G$ *предикатных операций* F и G называется предикатная операция, значения которой определяются по правилу

$$F \vee G)(X_1, X_2, \dots, X_n)=F(X_1, X_2, \dots, X_n) \vee G(X_1, X_2, \dots, X_n) \qquad (2)$$

для любых $X_1, X_2, \dots, X_n \in M$. *Конъюнкцией* $F \wedge G$ *предикатных операций* F и G называется предикатная операция, значения которой определяются по правилу

$$(F \wedge G)(X_1, X_2, \dots, X_n)=F(X_1, X_2, \dots, X_n) \wedge G(X_1, X_2, \dots, X_n) \qquad (3)$$

для любых $X_1, X_2, ..., X_n \in$ M. В равенствах (1)-(3) слева от знака равенства фигурируют операции $\neg$, $\vee$, $\wedge$ над предикатными операциями, справа знаки $\neg$, $\vee$, $\wedge$ обозначают операции над предикатами.

Булевой алгеброй предикатных операций называется любая алгебра предикатных операций с базисом операций, состоящим из отрицания, конъюнкции и дизъюнкции. Обратим внимание на то, что базисные операции булевой алгебры предикатных операций не являются предикатными операциями, они представляют собой операции над предикатными операциями, т.е. операции следующего, более высокого уровня. Аналогом этих операций в числовой математике может служить, к примеру, операция сложения линейных операторов, рассматриваемая в курсе функционального анализа. Имеется много различных булевых алгебр предикатных операций. Это обусловлено тем, что в определении данного понятия не указываются конкретные базисные элементы, которые в булевой алгебре могут быть выбраны по-разному. Их выбор и определяет конкретную булеву алгебру предикатных операций. В роли базисных элементов в булевой алгебре предикатных операций выступают некоторые фиксированные предикатные операции.

Тождественной предикатной операцией по переменной X_i ($i = \overline{1, n}$) называется операция со значениями

$$\mathrm{F}(X_1, X_2, ..., X_i, ..., X_n)=X_i \tag{4}$$

при любых $X_1, X_2, ..., X_i, ..., X_n \in$ M. Будем обозначать ее символом X_i. Каждой предикатной переменной взаимно однозначно соответствует своя тождественная предикатная операция. Тем не менее тождественные предикатные операции следует отличать от соответствующих предикатных переменных. Всего имеется n различных тождественных предикатных операций. *Константной предикатной операцией* называется любая операция со значением

$$\mathrm{F}(X_1, X_2, ..., X_n)=P \tag{5}$$

при любых $X_1, X_2, ..., X_n \in$ M. Здесь P - фиксированный предикат из M. Обозначим эту операцию символом P. Каждому предикату $P \in$ M соответствует своя константная предикатная операция P. Всего имеется $|\mathrm{M}|$ константных предикатных операций, где $|\mathrm{M}|$ - мощность множества M.

Алгеброй предикатных операций с константами и переменными называется булева алгебра предикатных операций, у которой базисными элементами являются всевозможные тождественные и константные предикатные операции. Тождественные предикатные операции называются *переменными алгебры предикатных операций*. Предикатные же переменные - это просто необходимый строительный материал, используемый для получения тех или иных предикатных операций, в частности, - предикатных операций вида X_i и X_i^P ($i=\overline{1,n}$, $P\in$M) (об операциях вида X_i^P речь пойдет ниже). Константные предикатные операции называются *константами алгебры предикатных операций*. Константные же предикаты - это, строго говоря, нечто иное, они используются в алгебре предикатов.

Приведем пример формулы алгебры предикатных операций с константами и переменными, имеющей предметные переменные x, y, предикатные переменные X, Y и универсум предметов $U=\{a, b, c\}$:

$$(x^a \vee x^b)X \vee \overline{x^a y^c}\,Y. \tag{a}$$

Фигурирующие в ней подформулы $x^a \vee y^b$ и $x^a y^c$ выражают константы данной алгебры. Формула (а), взятая в целом, выражает некоторую предикатную операцию F(X, Y)=Z, преобразующую произвольные предикаты X и Y в предикат Z. Возьмем, к примеру, в роли X - предикат $x^a y^b$, а в роли Y - предикат $x^a \vee y^b$ и отыщем предикат Z=F(X, Y). По формуле (а) находим: $Z=(x^a \vee x^b)x^a x^b \vee \overline{x^a y^c}(x^a \vee y^b) = x^a y^b \vee \overline{x^a y^c} =$ $=x^a y^b \vee \overline{x^a} \vee y^c = x^a y^b \vee y^b \vee y^c \vee y^a \vee y^b = x^b \vee x^c \vee y^a \vee y^b = x^a \vee y^c = \overline{x^a y^c}$. Итак, операция F($X$, Y)=Z, характеризуемая формулой (а), ставит в соответствие предикатам $X=x^a y^b$ и $Y=x^a \vee y^b$ предикат $Z=\overline{x^a y^c}$.

Теорема о неполноте алгебры предикатных операций с константами и переменными. *При любых m и n алгебра предикатных операций с константами и переменными, в универсуме предметов U которой содержится более одного элемента, неполна.*

Этой теоремой устанавливается удивительный факт. Казалось бы, алгебра предикатных операций с константами и переменными построена самым естественным способом, который обычно используется в классической математике: ввели константы, переменные, а также все необходимые операции, и тем не менее алгебра получилась неполноценная. Это наводит на мысль, что при конструировании полноценной алгебры предикатных операций надо действовать каким-то иным, нетрадиционным способом.

Доказательство. Рассмотрим предикатную операцию

$$\mathrm{F}(X_1)=\begin{cases}1, \text{ если } X_1=0,\\ 0, \text{ если } X_1\neq 0.\end{cases} \tag{б}$$

Предположим, что операция F выражается формулой алгебры, указанной в условии теоремы (обозначим для краткости эту алгебру символом А). В этом случае операция F выражается формулой с единственной переменной X_1. Поскольку $X_1 \wedge X_1 = X_1$ и $X_1 \wedge \overline{X_1} = 0$, то любая формула алгебры А с одной переменной X_1 после раскрытия скобок и упрощений приобретает вид $AX_1 \vee B\overline{X_1} \vee C$, где A, B, C - предикатные константы. Следовательно найдутся такие A, B и C, что $F(X_1)=AX_1 \vee B\overline{X_1} \vee C$ (α). Подставляя в (α) вместо X_1 нулевой и единичный предикаты, получаем: $F(0)=A0 \vee B\overline{0} \vee C = B \vee C$ (β), $F(1)=A1 \vee B\overline{1} \vee C = A \vee C$ (γ). Согласно (6), $F(0)=1$ (δ), $F(1)=0$ (ε). Используя (α)-(ε), получаем: $F(X_1)=AX_1 \vee B\overline{X_1} \vee C(X_1 \vee \overline{X_1}) = (A \vee C)X_1 \vee (B \vee C)\overline{X_1} = F(1)X_1 \vee F(0) \wedge \overline{X_1} = 0X_1 \vee 1\overline{X_1} = \overline{X_1}$. Итак, $F(X_1)=\overline{X_1}$. Этот результат противоречит определению (6), т. к. при $|U|>1$ множество всех предикатов содержит не только предикаты 0 и 1. Поэтому найденная операция $F(X_1)=\overline{X_1}$, вопреки определению, принимает не два, а большее число значений. Следовательно, предикатная операция F не выражается формулами алгебры А, иначе говоря, алгебра А неполна. Теорема доказана.

Универсум предикатов M состоит из двух предикатов 0 и 1 только в том случае, если универсум предметов U содержит всего лишь один элемент. Действительно, пусть $U=\{a\}$, тогда имеется всего две возможности: либо $P(a, a, \ldots, a)=0$, либо $P(a, a, \ldots, a)=1$ (в наборе элемент a встречается m раз). В случае, когда $M=\{0, 1\}$, алгебра предикатных операций с константами и переменными называется *алгеброй n-местных булевых функций*. Каждая ее функция имеет вид $F(X_1, X_2, \ldots, X_n)=Y$, где $X_1, X_2, \ldots, X_n$, Y - двоичные переменные, определенные на множестве $\{0,1\}$. В этой алгебре имеется n переменных $X_1, X_2, \ldots, X_n$ и всего две константы - предикаты 0 и 1.

Теорема о полноте алгебры булевых функций. *Алгебра булевых функций при любом n полна. Любая ее функция может быть выражена формулой*

$$F(X_1, X_2, \ldots, X_n) = \bigvee_{\sigma_1, \sigma_2, \ldots, \sigma_n \in \{0,1\}} F(\sigma_1, \sigma_2, \ldots, \sigma_n)\, X_1^{\sigma_1} X_2^{\sigma_2} \ldots X_n^{\sigma_n} \qquad (7)$$

называемой *совершенной дизъюнктивной нормальной формой (СДНФ)* булевой функции. Здесь $X_i^1 = X_i$, $X_i^0 = \overline{X_i}$.

Доказательство очевидно. Если эту теорему вырвать из контекста данного параграфа, то ее вполне можно было бы отождествить с известной теоремой о полноте алгебры булевых функций. На самом же деле отличие имеется, и оно состоит в том, что теперь речь идет не об алгебре двоичных знаков, рассматриваемой изолированно от других алгебр, а об особом случае алгебры предикатных операций с константами и переменными. Только в этом случае имеет место полнота алгебры. Именно наличием свойства полноты мы можем объяснить тот факт, что из всевозможных алгебр предикатных операций с константами и переменными только алгебра булевых функций получила широкое применение на практике.

Мы попытались построить полноценную алгебру предикатных операций, используя тождественные предикатные операции в качестве ее базисных элементов, но из этого пока ничего путного не вышло. Теперь мы пойдем другим, нестандартным, путем и попытаемся построить алгебру предикатных операций по образу и подобию дизъюнктивно-конъюнктивной алгебры предикатов [1]. *Предикатной операцией узнавания предиката P* по переменной X_i ($i = \overline{1,n}$) называется операция

$$F(X_1, X_2, ..., X_i, ..., X_n)=\begin{cases}1, \text{ если } X_i = P,\\ 0, \text{ если } X_i \neq P.\end{cases} \quad (8)$$

Зависимостью (8) мы ввели как раз те предикатные операции (среди них находится и операция (6)), которых нам не хватало в базисе алгебры предикатных операций с константами и переменными для обретения ею свойства полноты. Будем обозначать операцию (8) символом X_i^P. Все аргументы этой операции, кроме X_i, несущественны. Каждой паре - предикату $P\in$M и переменной X_i ($i = \overline{1,n}$) взаимно однозначно соответствует своя операция: предикат узнавания предиката X_i^P. *Дизъюнктивно-коньюктивной алгеброй предикатных операций* называется такая алгебра предикатных операций, у которой базисными операциями служат дизъюнкция и коньюнкция, а базисными элементами всевозможные константы $P\in$M и предикаты узнавания предиката X_i^P ($i = \overline{1,n}$, $P\in$M).

Важно подчеркнуть, что в числе базисных элементов дизъюнктивно-коньюнктивной алгебры предикатных операций нет тождественных предикатных операций, несмотря на то, что символ X_i фигурирует в выражении X_i^P. Этот символ является одним из двух элементов (наряду с символом P), образующих имя X_i^P базисного элемента алгебры предикатных операций. В составе предикатной операции X_i^P переменная X_i представляет собой лишь предикатную переменную, но не тождественную предикатную операцию. Поэтому дизъюнктивно-коньюнктивная алгебра предикатных операций не может быть отнесена к классу алгебр, которые строятся традиционным способом с использованием переменных алгебры в качестве ее базисных элементов.

Если бы операции X_i^P ($i = \overline{1,n}$, $P \in M$) были переведены из разряда предикатных операций в разряд базисных операций алгебры, тогда появилась бы возможность рассматривать переменные X_i как тождественные предикатные операции. Но мы этим (третьим) путем сейчас не можем пойти, поскольку нашей целью, в данном случае, является получение такой алгебры предикатных операций, которая могла бы служить надстройкой (расширением) дизъюнктивно-коньюктивной алгебры предикатов [1}, а этого можно достичь, только двигаясь по второму пути (т.е. используя операции X_i^P не в роли базисных операций, а только в роли ее базисных элементов).

Приведем пример формулы дизъюнктивно-коньюктивной алгебры предикатных операций:

$$X^{x^a} y^b \vee x^a, \tag{б}$$

полагая, что $U=\{a, b\}$, $m=n=1$. Выражение x^a, входящее в состав имени предиката узнавания предиката X^{x^a}, не является константой алгебры, оно выражает фиксированный предикат, выполняющий роль показателя в имени базисной предикатной операции. Этот предикат мы будем записывать в виде формулы дизъюнктивно-коньюктивной алгебры предикатов. Напротив, входящие далее в формулу (б) подформулы y^b и x^a представляют из себя константы алгебры, выражающие некоторые из базисных предикатных операций. Эти константы также записываются формулами дизъюнктивно-коньюктивной алгебры предикатов. Формула (б) выражает вполне определенную предикатную операцию $F(X)=Y$, преобразующую произвольно взятый предикат X в предикат Y. Возьмем, к примеру, в роли X предикат x^b и отыщем предикат $Y=F(X)$. По формуле (б) находим: $Y=(x^b)^{(x^a)} y^b \vee x^a = 0y^b \vee x^a = x^a$. Итак , операция $F(X)=Y$, характеризуемая формулой (б), ставит в соответствие предикату $X=x^b$ предикат $Y=x^a$. По результатам подобных вычислений построена таблица, характеризующая предикатную операцию $Y=F(X)$.

X	0	x^a	x^b	1
Y	x^a	1	x^a	x^a

Заметим, что табличное представление предикатных операций практически возможно только в очень простых случаях, подобных тому, который рассмотрен здесь. В более сложных случаях таблица предикатных операций становится необозримо большой, поэтому остается лишь один эффективный способ представления предикатных операций - формулами.

Теорема о полноте дизъюнктивно-конъюнктивной алгебры предикатных операций. *При любых U, m и n дизъюнктивно-конъюнктивная алгебра предикатных операций полна. Любая предикатная операция в ней выражается формулой*

$$\mathrm{F}(X_1, X_2, \ldots, X_n) = \bigvee_{P_1, P_2, \ldots, P_n \in M} \mathrm{F}(P_1, P_2, \ldots, P_n) X_1^{P_1} X_2^{P_2}, \ldots, X_n^{P_n}, \qquad (9)$$

называемой *СДНФ* предикатной операции F.

Доказательство очевидно. Логическое суммирование в СДНФ ведется по всем предикатным векторам $(P_1, P_2, \ldots, P_n) \in \mathrm{M}^n$. Символом $\mathrm{F}(P_1, P_2, \ldots, P_n)$ обозначена константа, являющаяся значением функции F от вектора $(P_1, P_2, \ldots, P_n)$. Обратим внимание на то, что в (9) символы $X_1, X_2, \ldots, X_n$ употреблены лишь как предикатные переменные, но не как предикатные операции, а символ F обозначает фиксированную предикатную операцию. Запишем, в виде примера, СДНФ предикатной операции, представленной таблицей:

$$x^a X^0 \vee 1\, X^{x^a} \vee x^a X^{x^b} \vee x^a X^1. \qquad (в)$$

В дизъюнктивно-конъюнктивной алгебре предикатных операций не обязательно иметь в качестве базисных элементов все константы. В базисе элементов достаточно сохранить из числа констант лишь предикат 0 и все предикаты узнавания предмета x_i^a ($i = \overline{1, m}$, $a \in U$). Все остальные предикаты выражаются средствами дизъюнктивно-конъюнктивной алгебры предикатов [1]. Предикаты 0 и x_i^a здесь рассматриваются в роли константных предикатных операций, а операции дизъюнкции и конъюнкции - в роли операций, действующих на предикатные операции. *Фундаментальной алгеброй* называется алгебра предикатных операций, у которой базисными операциями служат дизъюнкция и конъюнкция, а базисными элементами - предикат 0 и всевозможные предикаты узнавания предметов x_i^a ($i = \overline{1, m}$, $a \in U$) и предикаты узнавания предикатов X_j^P ($j = \overline{1, n}$, $P \in \mathrm{M}$). Фундаментальную алгебру можно рассматривать как сокращенный (но тем не менее полный) вариант дизъюнктивно-конъюнктивной алгебры предикатных операций.

Теорема о полноте фундаментальной алгебры. *Фундаментальная алгебра при любых U, m и n полна.*

Доказательство очевидно. В качестве примера формулы фундаментальной алгебры может служить формула (б).

Любое подмножество *B* носителя *A* алгебры A называется *замкнутым* в алгебре A, если результат каждой базисной операции алгебры

А, выполненной над произвольными элементами множества B, снова принадлежит множеству B. В этом случае будем говорить, что множество *B замкнуто* относительно каждой из базисных операций алгебры А. Если множество B замкнуто, то оно, вместе с базисными операциями (действие которых теперь ограничено только множеством B) и базисными элементами (только теми, которые содержатся в множестве B) алгебры А, тоже образуют алгебру. Обозначим эту алгебру буквой В. Так построенная алгебра В называется *подалгеброй* алгебры А. Выделим в множестве A всех предикатных операций подмножество B всех константных предикатных операций. Все их можно рассматривать просто как предикаты. Из базисных элементов фундаментальной алгебры в множестве B остались только предикаты узнавания предмета x_i^a ($i=\overline{1,m}$, $a\in U$). Сохраняем операции $\vee$ и $\wedge$. Ясно , что результатом этих операций, примененных к предикатам, снова будут предикаты. Получили дизъюнктивно-конъюнктивную алгебру предикатов. Следовательно, дизъюнктивно-конъюнктивная алгебра предикатов является подалгеброй фундаментальной алгебры. Дизъюнктивно-конъюнктивную алгебру предикатов получаем из фундаментальной алгебры, если из ее базиса исключим все предикаты узнавания предикатов X_j^P ($j=\overline{1,n}$, $P\in$ M).

Дизъюнктивно-конъюнктивная алгебра предикатов, рассматриваемая как алгебра предикатных операций, неполна. На ее языке можно выразить только все константные предикатные операции. С этой точки зрения дизъюнктивно-конъюнктивную алгебру предикатов можно рассматривать как фрагмент фундаментальной алгебры. Любопытно, что возможна и другая точка зрения, при которой фундаментальную алгебру можно рассматривать как *усеченный вариант* дизъюнктивно-конъюнктивной алгебры предикатов [4], определенной на расширенном универсуме предметов. В этом случае предикатные переменные переводятся в ранг предметных переменных, определенных на специальных множествах элементов с булевой структурой. В роли значений этих предметных переменных теперь выступают имена предикатов. При таком подходе алгебра предикатных операций *редуцируется* в алгебру предикатов. Проблема редукции алгебры предикатных операций в алгебру предикатов порождает много интересных задач, еще ждущих своего решения.

Консервативно расширим фундаментальную алгебру, вводя в ней базисную операцию отрицания и базисный элемент 1. Перечисляемые ниже законы называются *основными тождествами фундаментальной алгебры: законы идемпотентности* $X\vee X=X$, $X\wedge X=X$; *коммутативности* $X\vee Y=$ $=Y\vee X$, $X\wedge Y=Y\wedge X$; *ассоциативности* $(X\vee Y)\vee Z=X\vee(Y\vee Z)$, $(XY)Z=X(YZ)$; *дистрибутивности* $(X\vee Y)Z=$

$=XZ \vee YZ$, $XY \vee Z = (X \vee Z)(Y \vee Z)$; *элиминации* $X \vee XY = X$, $X(X \vee Y) = X$; *свертывания* $X \vee Y\overline{Y} = X$, $X(Y \vee \overline{Y}) = X$; *Моргана* $\overline{X \vee Y} = \overline{X}\,\overline{Y}$, $\overline{XY} = \overline{X} \vee \overline{Y}$; *закон двойного отрицания* $\overline{\overline{X}} = X$; *противоречия* $X\overline{X} = 0$; *исключенного третьего* $X \vee \overline{X} = 1$; *законы для предикатных операций* 0 *и* 1 $X \vee 0 = X$, $X0 = 0$, $X \vee 1 = 1$, $X1 = X$. Символами X, Y, Z здесь обозначены произвольные предикатные операции на множестве R.

Нижеследующие законы связывают предметные и предикатные переменные: *законы отрицания для предиката узнавания предмета*

$$\overline{x_i^a} = \bigvee_{\substack{b \in U \\ b \neq a}} x_i^b, \ (i = \overline{1, m}); \tag{10}$$

для предикатов узнавания предиката

$$\overline{X_j^P} = \bigvee_{\substack{Q \in M \\ Q \neq P}} X_j^Q, \ (j = \overline{1, n}) \tag{11}$$

законы истинности для предикатов узнавания предмета

$$\bigvee_{a \in U} x_i^a = 1, \ (i = \overline{1, m}); \tag{12}$$

для предикатов узнавания предиката

$$\bigvee_{P \in M} \overline{X_j^P} = 1, \ (j = \overline{1, n}); \tag{13}$$

законы ложности для предикатов узнавания предмета

$$x_i^a x_i^b = 0, \tag{14}$$

если $a \neq b$ (a, $b \in U$); *для предикатов узнавания предиката*

$$X_j^P X_j^Q = 0, \tag{15}$$

если $P \neq Q$ (P, $Q \in$ M). Базисную операцию отрицания можно выразить явно (т.е. с помощью прямого определения) в базисе фундаментальной алгебры с помощью законов отрицания и Моргана [1], а базисный элемент 1 - с помощью закона истинности. Операцию отрицания можно, кроме того, задать неявно (т.е. с помощью косвенного определения) законами противоречия и исключенного третьего.

Теорема о полноте системы основных тождеств фундаментальной алгебры. *Система основных тождеств фундаментальной алгебры полна при любых U, m и n.*

Доказательство аналогично доказательству теоремы о полноте системы основных тождеств дизъюнктивно-конъюнктивной алгебры предикатов [4]. Система основных тождеств фундаментальной алгебры сократима: более половины законов из нее можно исключить без потери ею свойства полноты. Установим, для примера, тождественность формул (б) и (в) фундаментальной алгебры с универсумом $U=\{a, b\}$. Согласно законам истинности для предикатов узнавания предмета и предиката имеем:

$$x^a X^0 \vee (x^a \vee x^b)\, X^{x^a} \vee x^a\, X^{x^b} \vee x^a X^1 = x^b\, X^{x^a} \vee x^a (x^0 \vee X^{x^a} \vee$$
$$\vee\, X^{x^b} \vee X^1) = x^b\, X^{x^a} \vee x^a\, 1 = x^b\, X^{x^a} \vee x^a.$$

Список литературы: 1. *Дударь З.В., Мельникова Р.В., Шабанов-Кушнаренко Ю.П.* Отношения как объекты формульного описания // АСУ и приборы автоматики. 1998. Вып. 107. С. 68-77. 2. *Шабанов-Кушнаренко Ю.П.* Теория интеллекта. Проблемы и перспективы. Х.: Выща шк. Изд-во при Харьк. ун-те, 1987. 159 с. 3. *Гвоздинская Н.А., Дударь З.В., Шабанов-Кушнаренко Ю.П.* О математическом описании смысла текстов естественного языка // Проблемы бионики. 1998. Вып 48. С. 141-149. 4. *Шабанов-Кушнаренко Ю.П.* Теория интеллекта. Математические средства. Х.: Выща шк. Изд-во при Харьк. ун-те, 1984. 143 с.

Поступила в редколлегию 10.03.98

Проблемы бионики
1998. Вып. 49. С. 14—22

ХТУРЭ

УДК 519.7

З.В. ДУДАРЬ, Н.С. КРАВЕЦ, Ю.П. ШАБАНОВ-КУШНАРЕНКО

О ПРИКЛАДНОЙ АЛГЕБРЕ ПРЕДИКАТНЫХ ОПЕРАЦИЙ*

Фундаментальная алгебра, описанная в работе [1], - это пока единственная из известных нам алгебр предикатных операций, для которой доказано, что она полна. Полнота означает, что с помощью формул фундаментальной алгебры можно выразить любую предикатную операцию, содержащуюся в ее носителе. Кроме того, только для фундаментальной алгебры известна полная система законов, позволяющая решить вопрос о том, выражают ли две произвольно взятые ее формулы одну и ту же предикатную операцию или нет. Эти качества фундаментальной алгебры делают ее незаменимой для теоретических изысканий в области логической математики. Вместе с тем, для практических применений фундаментальная алгебра не всегда удобна, поскольку ее формулы и тождества не обладают достаточной компактностью и изяществом. Поэтому продолжает оставаться актуальной задача разработки других алгебр предикатных операций, более удобных практически для тех или иных приложений, чем фундаментальная алгебра.

Вначале, опираясь на аппарат фундаментальной алгебры, изучим некоторые операции над предикатами, которые будут использованы ниже в качестве строительных блоков при построении более практичной алгебры предикатных операций, называемой нами алгеброй подстановочных операций, или прикладной алгеброй. Фундаментальная алгебра полна, следовательно, она универсальна в том смысле, что на ее языке можно выразить любые операции над предикатами. Ниже на языке фундаментальной алгебры выражаются операции замены и перестановки аргументов предиката и операции подстановки какого-либо предмета на место одного из аргументов предиката. Обычно эти операции воспринимаются как действия над формулами, однако ничто не мешает рассматривать их также и как операции над предикатами, выражаемыми этими формулами.

Заменой аргумента x_i на аргумент x_j называется операция

$$x_i / x_j(P)=Q, \tag{1}$$

---

* Статья публикуется в авторской редакции.

которая каждому предикату $P(x_1, x_2, ..., x_m)$ ставит в соответствие предикат $Q(x_1, x_2, ..., x_m)$ по следующему правилу: для любых $a_1, a_2, ..., a_m \in U$

$$Q(a_1, a_2, ..., a_{i-1}, a_i, a_{i+1}, ..., a_{j-1}, a_j, a_{j+1}, ..., a_m)=$$
$$=P(a_1, a_2, ..., a_{i-1}, a_j, a_{i+1}, ..., a_{j-1}, a_j, a_{j+1}, ..., a_m). \qquad (2)$$

Предикат, получаемый из предиката $P(x_1, ..., x_i, ..., x_j, ..., x_m)$ заменой аргумента x_i на аргумент x_j, будем записывать в виде $P(x_1, ..., x_j, ..., x_j, ..., x_m)$. Например, результатом замены аргумента y на аргумент x в предикате $P(x, y)$ будет предикат $P(x, x)$. Здесь x и y обозначают некоторые из аргументов $x_1, x_2, ..., x_m$. Подразумевается, что все остальные аргументы предиката $P(x, y)$ несущественны. Выполним, например, операцию замены переменной x_1 на переменную x_2 для предиката P, заданного формулой:

$$P(x_1, x_2, x_3)=x_1^1x_2^2\vee x_2^3x_3^3\vee x_1^2x_3^1. \qquad (a)$$

В результате получаем: $x_1/x_2(P)=x_2^1x_2^2\vee x_2^3x_3^3\vee x_2^2x_3^1=x_2^3x_3^3\vee x_2^2x_3^1$.

Предикат $Q(x_1, x_2, ..., x_m)$ называется *обратным* предикату $P(x_1, x_2, ..., x_m)$ по аргументам x_i, x_j, если для любых $a_1, a_2, ..., a_m \in U$

$$Q(a_1, a_2, ..., a_i, ..., a_j, ..., a_m)=P(a_1, a_2, ..., a_j, ..., a_i, ..., a_m). \qquad (3)$$

Предикат, обратный предикату $P(x_1, ..., x_i, ..., x_j, ..., x_m)$ по аргументам x_i и x_j будем записывать в виде $P(x_1, ..., x_j, ..., x_i, ..., x_m)$. Например, предикатом, обратным предикату $P(x, y, z)$ по аргументам x и z, будет предикат $P(z, y, x)$. Операция

$$x_i|x_j(P)=Q \qquad (4)$$

получения предиката $Q(x_1, x_2, ..., x_m)=P(x_1, ..., x_j, ..., x_i, ..., x_m)$ из предиката $P(x_1, ..., x_i, ..., x_j, ..., x_m)$ называется *обращением предиката P по аргументам* x_i и x_j или *перестановкой* его аргументов x_i и x_j. Выполним, к примеру, операцию перестановки аргументов x_1 и x_2 для предиката P, заданного формулой (а). В результате получаем $x_1|x_2(P)=x_2^1x_1^2\vee x_1^3x_3^3\vee x_2^2x_3^1$.

Подстановкой

$$x_i/a(P)=Q \qquad (5)$$

значения a на место аргумента x_i называется операция, которая каждому предикату $P(x_1, x_2, ..., x_m)$ ставит в соответствие предикат $Q(x_1, x_2, ..., x_m)$ по следующему правилу: для любых $a_1, a_2, ..., a_m \in U$

$$Q(a_1, ..., a_i, ..., a_m)=P(a_1, ..., a, ..., a_m). \qquad (6)$$

Предикат, получаемый из предиката $P(x_1, x_2, ..., x_{i-1}, x_i, x_{i+1}, ..., x_m)$ подстановкой значения a на место аргумента x_i, будем записывать в виде $Q(x_1, x_2, ..., x_m)=P(x_1, x_2, ..., x_{i-1}, a, x_{i+1}, ..., x_m)$. Аргумент x_i у предиката Q - несущественный. Например, результатом подстановки значения a на место аргумента y в предикате $P(x, y)$ будет предикат $Q(x, y)=P(x, a)$. Произведем, для примера, подстановку предмета 1 на место аргумента y для предиката P, заданного формулой (а). В результате получаем: $x_1/1(P)=1^1x_2^2\vee x_2^3x_3^3\vee 1^2x_3^3= =x_2^2\vee x_2^3x_3^3$.

Замена аргумента предиката выражается на языке фундаментальной алгебры следующим образом:

$$x_i/x_j(X)=\bigvee_{P\in M} P(x_1, x_2, ..., x_{i-1}, x_j, x_{i+1}, ..., x_{j-1}, x_j, x_{j+1}, ...,x_m)\cdot$$
$$\cdot X^{P(x_1, x_2, ..., x_{i-1}, x_i, x_{i+1}, ..., x_{j-1}, x_j, x_{j+1}, ..., x_m)} \quad (7)$$

Под символом X понимается любая из переменных X_l ($l=\overline{1,n}$). Выражаем перестановку аргументов предиката:

$$x_i|x_j(X)=\bigvee_{P\in M} P(x_1, x_2, ..., x_{i-1}, x_j, x_{i+1}, ..., x_{j-1}, x_i, x_{j+1}, ..., x_m)\cdot$$
$$\cdot X^{P(x_1, x_2, ..., x_{i-1}, x_i, x_{i+1}, ..., x_{j-1}, x_j, x_{j+1}, ..., x_m)} \quad (8)$$

Выражаем операцию подстановки:

$$x_i/a(X)=\bigvee_{P\in M} P(x_1, x_2, ..., x_{i-1}, a, x_{i+1}, ...,x_m)\cdot$$
$$\cdot X^{P(x_1, x_2, ..., x_{i-1}, x_i, x_{i+1}, ..., x_m)} \quad (9)$$

Зависимости (7)-(9) получены «силовым» методом подбора. В качестве показателей предикатов узнавания предиката берутся значения аргумента X рассматриваемой операции (замены, перестановки или подстановки), а в роли множителей при них в каждом случае подбирается нужный результат операции. Неудивительно, что в результате получаются громоздкие формулы, малопригодные для практики. Однако в данном случае важен сам факт выразимости операций замены, перестановки и подстановки на языке фундаментальной алгебры.

Замену, перестановку и подстановку можно понимать не только как предикатные операции, но и как операции над предикатными операциями. При таком понимании они определяются следующим образом:

$$x_i/x_j(\mathrm{F}(X_1, X_2, \ldots, X_n)) = \mathrm{F}(x_i/x_j(X_1), x_i/x_j(X_2), \ldots, x_i/x_j(X_n)); \qquad (10)$$

$$x_i\,x_j(\mathrm{F}(X_1, X_2, \ldots, X_n)) = \mathrm{F}(x_i|x_j(X_1), x_i|x_j(X_2), \ldots, x_i|x_j(X_n)); \qquad (11)$$

$$x_i/a(F(X_1, X_2, \ldots, X_n)) = F(x_i/a(X_1), x_i/a(X_2), \ldots, x_i/a(X_n)). \qquad (12)$$

В левой части равенств (10)-(12) символ F обозначает предикатную операцию, в правой части тот же символ обозначает эту же операцию, которая, однако, возведена в ранг операции над предикатными операциями. Равенства (10)-(12) выражают следующее правило: операции замены (перестановки или подстановки) и операцию F в их суперпозиции можно менять местами.

Откуда берутся определения (10)-(12)? Они выведены нами из наблюдений над практическими действиями математика, выполняющего операции замены, перестановки и подстановки в формулах. Математическая практика свидетельствует о том, что операции замены, перестановки и подстановки приходится производить не только над предикатами, но также и над предикатными операциями. И делается это за счет упомянутой выше перемены местами операций в их суперпозиции. Например, операция замены переменной, выполняемая над дизъюнкцией предикатов, производится при помощи изменения очередности выполнения операций x/y в их суперпозиции: $x/y(P(x)\vee Q(x)) = x/y(P(x))\vee x/y(Q(x)) = P(y)\vee Q(y)$. Из этого и подобных ему других фактов индуктивно выводим общее правило (10). Аналогично выводятся правила (11), (12). Например, когда мы выполняем подстановку x/a в какой-нибудь формуле, то в каждой ее части ищем вхождение переменной x и везде заменяем его на предмет a. При этом операция подстановки, которая вначале была внешней по отношению к формуле, теперь проникает внутрь нее в соответствии с правилом (12). Обратим внимание на то, что символы X_1, X_2, ..., X_n в зависимостях (7)-(9) выступают в роли предикатных переменных алгебры, они еще не возведены в ранг предикатных операций. Операции X^P ($P\in M$), фигурирующие в зависимостях (7)-(9), - это предикатные операции, но не операции алгебры. В выражениях (7)-(9) и в правой части равенств (10)-(12) символы x_i/x_j, $x_i|x_j$ и x_i/a обозначают предикатные операции. В левой части равенств (10)-(12) те же символы обозначают операции над предикатными операциями.

Выразим, далее, на языке фундаментальной алгебры кванторы общности и существования, которые широко используются в классической математике в качестве логического инструментария. *Квантор общ-*

ности, как известно, определяется следующим образом: если $U=\{a_1, a_2, \ldots, a_k\}$, то $\forall xP(x)=P(a_i)\wedge P(a_2)\wedge\ldots\wedge P(a_k)$. В общем случае:

$$\forall x_i(X)=\bigwedge_{a\in U} x_i/a(X). \tag{13}$$

Аналогично определяется *квантор существования*: если $U=\{a_1, a_2, \ldots, a_k\}$, то $\exists xP(x)=P(a_1)\vee P(a_2)\vee\ldots\vee P(a_k)$. В общем случае:

$$\exists x_i(X)=\bigvee_{a\in U} x_i/a(X). \tag{14}$$

Зависимости (13) и (14) можно применять не только к конечному, но и к бесконечному универсуму U. Кванторы общности и существования можно понимать не только как предикатные операции, но и как операции над предикатными операциями. В этом случае они определяются следующими равенствами:

$$\forall x_i(\mathrm{X})=\bigwedge_{a\in U} x_i/a(\mathrm{X}); \tag{15}$$

$$\exists x_i(\mathrm{X})=\bigvee_{a\in U} x_i/a(\mathrm{X}). \tag{16}$$

О происхождении определений (15) и (16) можно высказать соображения, аналогичные тем, которые были приведены по поводу определений (10)-(12). Символ X обозначает произвольную предикатную операцию.

Кванторы общности и существования, когда значения предметной переменной ограничены множеством $A\subseteq U$, определяются следующими зависимостями:

$$\forall x\in A P(x)=\forall x(A(x)\supset P(x)); \tag{17}$$

$$\exists x\in A P(x)=\exists x(A(x)\wedge P(x)). \tag{18}$$

Определяем кванторы по предикатной переменной:

$$\forall P\ \mathrm{F}(P)=\bigwedge_{P\in \mathrm{M}} \mathrm{F}(P); \tag{19}$$

$$\exists P\ \mathrm{F}(P)=\bigvee_{P\in \mathrm{M}} \mathrm{F}(P). \tag{20}$$

Здесь F(P) - произвольно выбранная предикатная операция, действующая на предикат P; M - универсум предикатов [1].

Определяем кванторы по набору предметных переменных $\xi=(x_1, x_2, ..., x_m)$:

$$\forall\xi P(\xi)=\forall x_1\forall x_2...\forall x_m P(x_1, x_2, ..., x_m); \quad (21)$$

$$\exists\xi P(\xi)=\exists x_1\exists x_2...\exists x_m P(x_1, x_2, ..., x_m). \quad (22)$$

С их помощью можно выразить кванторы по предикатной переменной P, значения которой ограничены предикатами, включенными в фиксированный предикат M:

$$\forall P\subseteq M\ \mathrm{F}(P)= \bigwedge_{P\in \mathrm{M}} (\forall\xi(P(\xi)\supset M(\xi))\supset \mathrm{F}(P)); \quad (23)$$

$$\exists P\subseteq M\ \mathrm{F}(P)= \bigvee_{P\in \mathrm{M}} (\forall\xi(P(\xi)\supset M(\xi))\wedge \mathrm{F}(P)). \quad (24)$$

Зависимости (23) и (24) можно обобщить следующим образом:

$$\forall P\subseteq M\ \mathrm{F}(P, M)= \bigwedge_{P\in \mathrm{M}} (\forall\xi(P(\xi)\supset M(\xi))\supset \mathrm{F}(P, M)); \quad (25)$$

$$\exists P\subseteq M\ \mathrm{F}(P, M)= \bigvee_{P\in \mathrm{M}} (\forall\xi(P(\xi)\supset M(\xi))\wedge \mathrm{F}(P, M)). \quad (26)$$

Здесь F(P, M) - произвольно выбранная предикатная операция, действующая на предикаты P и M. Заметим, что выражения (19), (20) и (23)-(26) не выводят нас за пределы класса формул фундаментальной алгебры. Предикатные операции F(P) и F(P, M) выражаются на языке фундаментальной алгебры (так как это не переменные, а фиксированные операции). Вместе с тем, эти выражения нельзя причислить к формулам классического логического исчисления первого порядка, их нельзя также отнести и к формулам других широко известных логик [2, с. 49-54]. Фундаментальная алгебра предикатных операций, насколько мы можем судить на основании анализа литературных источников, представляет собой структуру, для которой нет аналога в современной математической логике.

Теперь мы полностью подготовлены к введению прикладной алгебры предикатных операций. *Прикладной алгеброй* называется алгебра предикатных операций с базисом операций, образованным из подстановок вида $x_i/a(\mathrm{X})$ ($i=\overline{1,m}$, $a\in U$), а также операций отрицания и дизъюнкции, и с базисом элементов, образованным из предикатов равенства вида $D(x_1, x_i)$ ($i=\overline{2,m}$) (здесь они выступают в роли константных предикатных операций), и предикатных переменных X_i ($i=\overline{1,n}$) (представляющих в данном случае тождественные предикатные операции). Число элементов в базисе прикладной алгебры гораздо меньше, чем в базисе фундаментальной алгебры, однако число базисных операций в ней резко возрастает. Алгебры называются *равносильными*, если на языке их формул описывается одно и то же множество объектов.

Теорема о равносильности фундаментальной и прикладной алгебр. *Фундаментальная и прикладная алгебры предикатных операций, имеющие один и тот же носитель, равносильны.*

Доказательство. Равносильность фундаментальной и прикладной алгебр будет доказана, если удастся базисные элементы и операции фундаментальной алгебры выразить через базисные элементы и операции прикладной алгебры, и наоборот. 1) Выражаем базис фундаментальной алгебры через базис прикладной. а) Выражаем базисные операции фундаментальной алгебры. Дизъюнкция уже имеется. Выражаем конъюнкцию через дизъюнкцию и отрицание: $X \wedge Y = \overline{\overline{X} \vee \overline{Y}}$. Отрицание имеется. б) Выражаем базисные элементы: $0=X_1 \wedge X_1$. Переменная X_1 имеется. Выражаем предикаты узнавания предметов: $x_1^a = D(x_1, a) = x_2/a(D(x_1, x_2))$; если же $i=\overline{2,m}$, то $x_i^a=D(a, x_i)=x_1/a(D(x_1, x_i))$ $(a \in U)$. Подстановочные операции и предикаты равенства имеются. Выражаем предикаты узнавания предиката:

$$X_j^P=\forall\xi(X_j(\xi)\sim P(\xi)), \tag{7}$$

где $j=\overline{1,n}$, $P \in$ M. Если предикатная переменная X_j имеется в фундаментальной алгебре, то она имеется и в прикладной, поскольку носители обеих алгебр совпадают. Любой фиксированный предикат P выражается формулой дизъюнктивно-конъюнктивной алгебры предикатов с помощью операций $\vee$ и $\wedge$, примененных к предикатам узнавания предмета x_i^a ($i=\overline{1,m}$, $a \in U$), которые уже выражены. Квантор общности по набору предметных переменных выражается через кванторы по одной переменной зависимостью (21). Кванторы общности по одной переменной выражаются через подстановочные операции и конъюнкцию по формуле (15). Весь базис фундаментальной алгебры выражен через базис прикладной алгебры. 2) Выражаем базис прикладной алгебры через базис фундаментальной. а) Выражаем базисные операции прикладной алгебры. Подстановочные операции выражаем по формулам (9) и (12). В силу полноты фундаментальной алгебры любая предикатная операция в ней выражается. Отрицание выражаем с помощью законов Моргана и законов отрицания для узнаваний предметов и предикатов. Дизъюнкция уже имеется. б) Выражаем базисные элементы. Предикат равенства уже выражен зависимостью (2) из работы [3]. Тождественную предикатную операцию выражаем формулой

$$X_i = \bigvee_{P \in M} P X_i^P. \tag{28}$$

В этом выражении справа от знака равенства символ X_i выражает только предикатную переменную, слева - тот же символ X_i уже возведен в ранг тождественной предикатной операции. Весь базис прикладной алгебры выражен через базис фундаментальной алгебры. Итак, фундаментальная и прикладная алгебры равносильны. Теорема доказана.

Из доказанной теоремы и теоремы о полноте фундаментальной алгебры [1] непосредственно следует

Теорема о полноте прикладной алгебры. *Прикладная алгебра полна при любых U, m и n.*

Прикладная алгебра еще слабо изучена. Требует решения вопрос о ее несократимости, об отыскании полной системы ее тождеств, ждут разработки методы решения ее уравнений и многие другие проблемы. Выразим все предикаты равенства в базисе прикладной алгебры. Равенства $D(x_1, x_i)$ ($i=\overline{2,n}$) уже содержатся в самом базисе прикладной алгебры. Если $i=\overline{2,n}$ и $j=\overline{2,n}$, то предикаты равенства выражаем следующим образом:

$$D(x_i, x_j) = \exists x_1 (D(x_1, x_i) \wedge D(x_1, x_j)). \tag{29}$$

Выражать кванторы на языке прикладной алгебры мы уже умеем. Для $i=1$ или $j=1$ выражение (29) не годится, т.к. возникает коллизия переменных. Квантор существования в (29) переменную x_1 исключает (делает ее несущественной). Справедливость равенства (29) следует из симметричности и транзитивности предиката равенства. Если $i=1$ и $j=\overline{2,n}$, то, полагая $D(x_1, x_j)$, приходим к базисным равенствам. Когда же $j=1$ и $i=\overline{2,n}$, то, полагая $D(x_i, x_j)=D(x_1, x_i)$, снова приходим к базисным равенствам. Для оставшегося случая $i=j=1$ полагаем $D(x_1, x_1)=1$, используя рефлексивность равенства. Единичный же предикат на языке прикладной алгебры выражается: $1=X_1 \vee \overline{X_1}$. Итак, все равенства выражены через базисные. Для примера, получим равенство $D(x_2, x_3)$ из базисных равенств прикладной алгебры с помощью зависимости (29). Пусть $U=\{a, b\}$, тогда $D(x_1, x_2)=x_1^a x_2^a \vee x_1^b x_2^b$; $D(x_1,x_3) = x_1^a x_3^a \vee x_1^b x_3^b$. Находим:

$D(x_2, x_3) = x_1/a\ ((x_1{}^a x_2{}^a \vee x_1{}^b x_2{}^b)(x_1{}^a x_3{}^a \vee x_1{}^b x_3{}^b)) \vee x_1/b \wedge ((x_1{}^a x_2{}^a \vee x_1{}^b x_2{}^b)(x_1{}^a x_3{}^a \vee x_1{}^b x_3{}^b)) = x_2{}^a \wedge x_3{}^a \vee x_2{}^b x_3{}^b$.

Операция замены аргументов предиката в прикладной алгебре выражается следующим образом:

$$x_i/x_j(P)=\exists\sigma(x_i/\sigma(D_{ij}\wedge P)),\ (i=\overline{1,m}). \tag{30}$$

Предикат равенства здесь действует обратно подстановке, он восстанавливает переменную, исчезнувшую в формуле в результате подстановки. Для примера, выразим замену аргументов предиката через подстановки по формуле (30). Пусть $U=\{a,b,c\}$, $P(x_1,x_2)= x_1{}^a x_2{}^b \vee x_1{}^b x_2{}^a$. Тогда $x_1/x_3(P) = \exists\sigma(x_1/\sigma(D_{13}\wedge P) = x_1/a(D_{13}\wedge P) \vee x_1/b\ (D_{13}\wedge P) \vee x_1/c\ (D_{13}\wedge P) = x_1/a(D_{13}) \wedge x_1/a(P) \vee x_1/b(D_{13}) \wedge x_1/b\ (P) \vee x_1/c\ (D_{13}) \wedge x_1/c(P)$. Далее: $x_1/a(D_{13}) = x_1/a(x_1{}^a x_3{}^a \vee x_1{}^b x_3{}^b \vee x_1{}^c x_3{}^c) = x_3{}^a$; $x_1/b(D_{13}) = x_3{}^b$; $x_1/c(D_{13})=x_3{}^c$; $x_1/a(P) = x_1/a(x_1{}^a x_2{}^b \vee x_1{}^b x_2{}^a) = x_2{}^b$; $x_1/b(P) = x_2{}^a$; $x_1/c(P) = 0$; $x_1/x_3(P) = x_3{}^a x_2{}^b \vee x_3{}^b x_2{}^a \vee x_3{}^c 0 = x_3{}^a x_2{}^b \vee x_3{}^b \wedge x_3{}^a$. Мы видим, что в результате применения правила (30) в формуле предиката $P(x_1, x_2)$ произошла замена переменной x_1 на x_3. Операция перестановки аргументов предиката выражается через замены аргументов предиката следующим образом:

$$x_i|x_j(P)=x_k/x_j\, x_j/x_i\, x_i/x_k(P). \tag{31}$$

Здесь в роли x_k используется какой-нибудь из несущественных аргументов предиката P. Для примера, выразим перестановку аргументов у конкретного предиката через замены аргументов по формуле (31). Пусть $U=\{a, b\}$; $P(x_1, x_2) = x_1{}^a x_2{}^b \vee x_1{}^b$. Тогда $x_1|x_2(P) = x_3/x_2 x_2/x_1 x_1/x_3\ (x_1{}^a x_2{}^b \vee x_1{}^b)$; $x_1/x_3\ (x_1{}^a x_2{}^b \vee x_1{}^b) = x_3{}^a x_2{}^b \vee x_3{}^b$; $x_2/x_1(x_3{}^a x_2{}^b \vee x_3{}^b) = x_3{}^a x_1{}^b \vee x_3{}^b$; $x_3/x_2(x_3{}^a x_1{}^b \vee x_3{}^b) = x_2{}^a x_1{}^b \vee x_2{}^b$. Получили перестановку аргументов x_1 и x_2.

Список литературы: 1. *Дударь З.В., Кравец Н.С., Шабанов-Кушнаренко Ю.П.* О фундаментальной алгебре предикатных операций – См. статью в настоящем сборнике. С. 3–13. 2. *Справочная* книга по математической логике. Теория моделей: В 4 ч.: Пер. с англ. / Под. Ред. Дж. Барвайса. М.: Наука, 1982. Ч.1. 391 с. 3. *Дударь З.В., Самуйлик И.Г., Шабанов-Кушнаренко Ю.П.* Отображения как объекты формульного описания // Радиоэлектроника и информатика. 1998. Вып. 2. С. 8–15.

Поступила в редколлегию 10.03.98

Проблемы бионики
1998. Вып. 49. С. 23—33

ХТУРЭ

УДК 681.513.7

Е.В. БОДЯНСКИЙ

ОБНАРУЖЕНИЕ РАЗЛАДОК В НЕЛИНЕЙНЫХ СТОХАСТИЧЕСКИХ ПОСЛЕДОВАТЕЛЬНОСТЯХ С ПОМОЩЬЮ РЕКУРРЕНТНЫХ ИСКУССТВЕННЫХ НЕЙРОННЫХ СЕТЕЙ

Задача обнаружения изменения свойств стохастических последовательностей тесно связана с проблемой технической и медицинской диагностики. Эта задача широко обсуждалась в научной литературе [1–5], а для ее решения предложено множество подходов, связанных большей частью с идеями математической статистики, теории случайных процессов, распознавания образов, кластер-анализа и т.п. Не вдаваясь в критику полученных результатов, заметим, что жесткие предположения о статистических свойствах рядов ограничивают возможности данных подходов.

Более универсальным представляется многомодельный подход [1; 5—7]. Его суть состоит в том, что диагностируемый сигнал проходит через множество моделей, каждая из которых базируется на определенной гипотезе относительно характера возможных изменений. Если какая-либо из гипотез действительно подтверждается, то сигналы обновлений на выходе соответствующей модели должны быть малыми. Таким образом, решающий механизм, по сути, основан на отыскании той модели, на выходе которой обновления минимальны, а вероятность соответствующей гипотезы максимальна. Достоинства изложенного подхода несомненны, однако реальная последовательность обычно настолько многообразна, что никакая из моделей (как правило, линейных) полностью не отражает ее изменяющихся свойств.

Последние годы характеризуются резким увеличением количества исследований в области теории и практики искусственных нейронных сетей, в том числе и для решения задач диагностики [8–14]. Предложенные диагностирующие нейронные сети реализуют в основном идеи теории классификации при наличии обучающей выборки, при этом появление не предусмотренных априори состояний процесса может быть не обнаружено сетью.

Предлагаются архитектура многослойной рекуррентной искусственной нейронной сети и алгоритмы настройки ее параметров, сочетающие в себе достоинства многомодельного подхода и аппроксимирующих свойств прогнозирующих нейронных сетей [15–18] с нелинейными функциями активации. Изменение свойств стохастической последовательности фиксируется с помощью диагностирующего вектора, элементы которого являются синаптическими весами выходного нейрона.

Предлагаемая нами архитектура диагностирующей рекуррентной нейронной сети показана на рис. 1 и представляет собой сеть элементарных нейронов. Последние отличаются друг от друга видом функций активации и алгоритмами обучения, являющимися в общем случае градиентными процедурами безусловной или условной оптимизации.

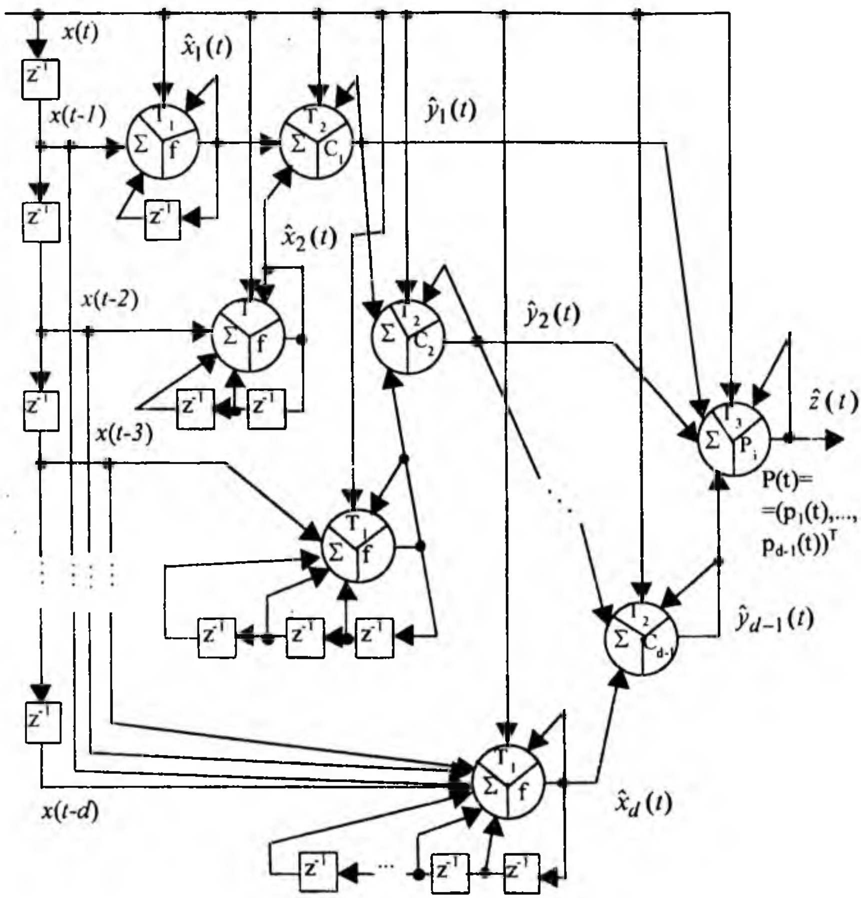


Рис. 1

На рис. 2 изображен элементарный искусственный нейрон, при этом сектор $\sum$ соответствует операции линейного суммирования входов $\sum w_j(t)x_j(t)$ с настраиваемыми синаптическими весами $w_j(t)$, сектор f соответствует нелинейному преобразованию с помощью функции активации $\hat{y}(t) = f(\sum w_j(t)x_j(t))$, сектор T соответствует алгоритму настройки нейрона на основе рекуррентных процедур типа

$$w(t+1) = w(t) + \psi(y(t), \hat{y}(t)),$$

где $y(t)$ – обучающий сигнал.

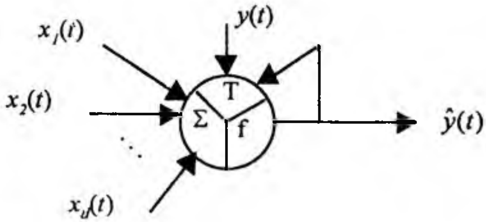


Рис. 2

Контролируемая стохастическая последовательность $\{x(t)\}$, t=1, 2, ... подается на входной слой сети, представляющий собой последовательную цепочку элементов чистого запаздывания $z^{-1}(z^{-1}x(t) = x(t-1))$, в результате чего на выходе этого слоя формируется набор задержанных значений временного ряда $x(t-1)$, $x(t-2)$, ..., $x(t-d)$. Чем больше значение d, тем более широкими диагностирующими возможностями обладает нейронная сеть.

Первый скрытый слой образован нейронами типа Мак-Каллоха — Питтса, на суммирующие входы которых подаются задержанные значения контролируемой последовательности $x(t)$ и по цепи обратной связи — задержанные значения прогноза $\hat{x}_j(t)$, $j = 1, 2, \ldots, d$. Входы нейронов T_1 соответствуют входам алгоритма настройки синаптических весов, а f описывает нелинейную функцию активации нейронов.

В результате обработки сигнала $x(t)$ нейронами первого слоя на их выходах появляются прогнозные оценки

$$\begin{cases} \hat{x}_1(t) = f(x(t-1),\ \hat{x}_1(t-1)); \\ \hat{x}_2(t) = f(x(t-1), x(t-2), \hat{x}_2(t-1), \hat{x}_2(t-2)); \\ \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \\ \hat{x}_d(t) = f(x(t-1),\dots,\ x(t-d),\ \hat{x}_d(t-1),\dots,\ \hat{x}_d(t-d)), \end{cases} \tag{1}$$

соответствующие нелинейному процессу авторегрессии – скользящего среднего (NARMA-процесс) [16; 17] порядка от 1 до d. Задачей, решаемой сетью, является определение текущего значения порядка NARMA-процесса и моментов возможного его изменения в реальном времени.

$j = 1, 2, \dots, d-1$:

Нейронами второго скрытого слоя T_2 производится попарное объединение выходов нейронов T_1 с целью получить оценки $\hat{y}_j(t)$, $j = 1, 2, \dots, d-1$:

$$\begin{cases} \hat{y}_1(t) = \varphi(\hat{x}_1(t), \hat{x}_2(t), c_1), \qquad \hat{x}_1(t) = \hat{y}_0(t); \\ \hat{y}_2(t) = \varphi(\hat{y}_1(t), \hat{x}_3(t), c_2, c_1); \\ \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \\ \hat{y}_{d-1}(t) = \varphi(\hat{y}_{d-2}(t), \hat{x}_d(t), c_{d-1}, c_{d-2}, \dots, c_1) \end{cases} \tag{2}$$

и весовых коэффициентов c_j, характеризующих точность объединяемых прогнозов $\hat{y}_{j-1}(t)$, $\hat{x}_{j+1}(t)$ и объединенного $\hat{y}_j(t)$. Вектор текущих весов $C(t) = (c_1(t),\ c_2(t),\ \dots,\ c_{d-1}(t))^T$ описывает качество прогнозирования, достигаемое во втором скрытом слое в каждый текущий момент времени, а изменение соотношений между его элементами уже само по себе свидетельствует об изменении структуры и параметров контролируемой последовательности $x(t)$.

Выходной слой сети образован одним нейроном T_3. Его выходом являются прогноз $\hat{z}(t)$ и вектор диагностических признаков $P(t) = (p_1(t), p_2(t), \dots, p_{d-1}(t))^T$, элементы которого $p_j(t)$ соответствуют вероятностям того, что "истинное" состояние процесса $x(t)$ наилучшим образом описывается оценкой $\hat{y}_j(t)$. Максимальное значение $p_j(t)$ определяет порядок диагностируемой NARMA-последовательности в текущий момент времени t, а непрерывное уточнение вектора $P(t)$ с помощью соответствующих алгоритмов настройки нейронов позволяет обнаруживать момент "разладки" процесса $x(t)$.

Выходной сигнал j-го нейрона первого скрытого слоя может быть представлен в виде

$$\hat{x}_j(t) = f_j\left(\sum_{i=1}^{j} w_{ji}(t)x(t-j) + \sum_{i=1}^{j}\omega_{ji}(t)\hat{x}_j(t-j) + w_{j0}(t)\right) = \\ = f_j(w_j^T(t)X_j(t)) = f_j(\tilde{X}_j(t)), \tag{3}$$

где $f_j(\bullet)$ – функция активации j-го нейрона; $w_j(t) = (w_{j0}(t), w_{j1}(t), \ldots, w_{jj}(t), \omega_{j1}(t), \ldots, \omega_{jj}(t))^T$ – $(2j+1)\times 1$ – вектор настраиваемых синаптических весов; $X_j(t) = (1, x(t-1), \ldots, x(t-j), \hat{x}(t-1), \ldots, \hat{x}(t-j))^T$ – вектор обобщенных входов; $\tilde{X}_j(t) = w_j^T(t)x_j(t)$; $j = 1, 2, \ldots, d$; $t = 1, 2, \ldots$ – текущее дискретное время. Выражение (3) представляет собой описание нелинейной стохастической последовательности авторегрессии – скользящего среднего j-го порядка, причем, как отмечалось в [16], именно определение значения j является наиболее сложной проблемой, в связи с чем для решения конкретных задач целесообразно принимать достаточно большие значения d.

Вводя в рассмотрение ошибку прогнозирования j-го нейрона

$$\varepsilon_j(t) = x(t) - \hat{x}_j(t) = x(t) - f_j(\tilde{X}_j(t)), \tag{4}$$

можно записать градиентную процедуру настройки синаптических весов в виде [19; 20]

$$w_j(t+1) = w_j(t) + \eta_j(t)\varepsilon_j(t)\nabla_{w_j} f_j(\tilde{X}_j(t)) = \\ = w_j(t) + \eta_j(t)\varepsilon_j(t)G_j(t). \tag{5}$$

Здесь $\eta_j(t)$ – параметр шага поиска, принимаемый чаще всего постоянным; $\nabla_{w_j} f_j(\tilde{X}_j(t)) = G_j(t)$ – градиент функции активации по синаптическим весам.

Заметим, что для наиболее распространенных в теории и практике многослойных нейронных сетей функций активации

$$\begin{cases} f_j'(\tilde{X}_j(t)) = \tanh(\gamma_j \tilde{X}_j(t)) = \dfrac{1 - e^{-2\gamma_j \tilde{X}_j(t)}}{1 + e^{-2\gamma_j \tilde{X}_j(t)}}; \\ f_j''(\tilde{X}_j(t)) = \dfrac{1}{1 + e^{-\gamma_j \tilde{X}_j(t)}} \end{cases} \tag{6}$$

градиенты имеют вид

$$\begin{cases} \nabla_{w_j} f_j'(\tilde{X}_j(t)) = G_j'(t) = \gamma_j (1-(f_j'(\tilde{X}_j(t)))^2) X_j(t) = \\ \qquad\qquad = \gamma_j (1-\hat{x}_j^2(t)) X_j(t); \\ \nabla_{w_j} f_j''(\tilde{X}_j(t)) = G_j''(t) = \gamma_j f_j''(\tilde{X}_j(t))(1-f_j''(\tilde{X}_j(t))) X_j(t) = \\ = \gamma_j \hat{x}_j(t)(1-\hat{x}_j(t)) X_j(t). \end{cases} \tag{7}$$

Сходимость градиентных процедур типа (5) обеспечивается в достаточно широком интервале варьирования параметра шага $\eta_j(t)$: для детерминированного случая данный параметр должен удовлетворять условиям $0 < \eta_j(t) < 2 / Lp$ (здесь Lp – константа Липшица для оптимизируемой функции), а для стохастического случая – условиям Дворецкого. При этом вполне естественным представляется выбор шага, обеспечивающего максимальную скорость сходимости.

Максимизация функции

$$W_j(t) = \|\tilde{w}_j(t)\|^2 - \|\tilde{w}_j(t+1)\|^2, \tag{8}$$

где $\tilde{w}_j(t) = w_j - w_j(t)$; w_j – оптимальный вектор синаптических весов, приводит к неконструктивной оценке

$$\eta_j(t) = \frac{(w_j - w_j(t))^T G_j(t)}{\varepsilon_j(t)\|G_j(t)\|^2}. \tag{9}$$

Однако, если выполняется соотношение, справедливое для выпуклых функций,

$$\begin{aligned} &(w_j - w_j(t))^T G_j(t) \le f_j(w_j^T X_j(t)) - f_j(w_j^T(t) X_j(t)) = \\ &= x(t) - f_j(w_j^T(t) X_j(t)), \end{aligned} \tag{10}$$

из (9) следует, что

$$0 < \eta_j(t) \le \|G_j(t)\|^2. \tag{11}$$

Рассмотрев одношаговый вариант алгоритма Маквардта [21]

$$w_j(t+1) = w_j(t) + \left(G_j(t)G_j^T(t) + \rho(t)E\right)^{-1} G_j(t)\varepsilon_j(t), \qquad (12)$$

где $\rho(t) > 0$; E – единичная матрица, и использовав известные соотношения теории псевдообратных матриц

$$\begin{cases} \lim\limits_{\rho(t)\to 0}\left(G_j(t)G_j^T(t) + \rho(t)E\right)^{-1} = \left(G_j(t)G_j^T(t)\right)^{+}; \\ \left(G_j(t)G_j^T(t)\right)^{+} G_j(t) = \left(G_j^T(t)\right)^{+} = G_j(t)\left\|G_j(t)\right\|^{-2}, \end{cases} \qquad (13)$$

можно записать оптимальный по быстродействию вариант (5) в виде [22]

$$w_j(t+1) = w_j(t) + \frac{x(t) - \hat{x}_j(t)}{\left\|G_j(t)\right\|^2} G_j(t), \qquad (14)$$

совпадающий в линейном случае с алгоритмом настройки синаптических весов Уидроу–Хоффа.

Заметим, что для функций активации типа (6) алгоритм (14) приобретает форму

$$\begin{cases} w_j'(t+1) = w_j'(t) + \dfrac{x(t) - \hat{x}_j(t)}{\gamma_j(1-\hat{x}_j(t))\left\|X_j(t)\right\|^2} X_j(t); \\ w_j''(t+1) = w_j''(t) + \dfrac{x(t) - \hat{x}_j(t)}{\gamma_j\hat{x}_j(t)(1-\hat{x}_j(t))\left\|X_j(t)\right\|^2} X_j(t). \end{cases} \qquad (15)$$

Для того чтобы обеспечить дополнительные сглаживающие свойства алгоритму (14), можно ввести следующую экспоненциально взвешенную модификацию:

$$\begin{cases} w_j(t+1) = w_j(t) + r_j^{-1}(t)(x(t) - \hat{x}_j(t))G_j(t); \\ r_j(t) = \alpha r_j(t-1) + \left\|G_j(t)\right\|^2, \quad 0 \le \alpha \le 1, \quad r_j(0) = 1. \end{cases} \qquad (16)$$

При $\alpha = 0$ она совпадает с (14), а в линейном случае при $\alpha = 1$ – с процедурой стохастической аппроксимации Гудвина–Рэмеджа–Кейнеса [23].

Во втором слое предлагаемой нейронной сети производится попарное объединение выходных сигналов первого слоя в виде

$$\hat{y}_j(t) = c_j(t)\hat{y}_{j-1}(t) + (1 - c_j(t))\hat{x}_{j+1}(t), \qquad (17)$$

где $\hat{y}_0(t) \equiv \hat{x}_1(t)$; $j = 1, 2, \ldots, d-1$, а веса $c_j(t)$ задают сравнительную точность прогнозов $\hat{y}_{j-1}(t)$ и $\hat{x}_{j+1}(t)$ и обеспечивают несмещенность прогноза $\hat{y}_j(t)$.

Чтобы найти значения $c_j(t)$, обеспечивающие оптимальность $\hat{y}_j(t)$, введем $(t \times 1)$ – векторы наблюдений и ошибок

$$\begin{cases} X(t) = (x(1),\ x(2),\ \ldots,\ x(t))^T; \\ \hat{Y}_j(t) = (\hat{y}_j(1),\ \hat{y}_j(2),\ \ldots,\ \hat{y}_j(t))^T; \\ \hat{X}_j(t) = (\hat{x}_j(1),\ \hat{x}_j(2),\ \ldots,\ \hat{x}_j(t))^T; \\ V_j(t) = X(t) - \hat{Y}_j(t); \\ V_{j-1}(t) = X(t) - \hat{Y}_{j-1}(t); \\ V_{x,j-1}(t) = X - \hat{X}_{j+1}(t) \end{cases}$$

и запишем очевидные соотношения

$$V_j(t) = c_j(t)V_{j-1}(t) + (1 - c_j(t))V_{x,j+1}(t); \tag{18}$$

$$\begin{aligned} \|V_j(t)\|^2 &= c_j^2(t)\|V_{j-1}(t)\|^2 + 2c_j(t)(1 - c_j(t))V_{j-1}^T(t)V_{x,j+1}(t) + \\ &+ (1 - c_j(t))^2\|V_{x,j+1}(t)\|^2. \end{aligned} \tag{19}$$

Тогда, решив уравнение

$$\begin{aligned} \frac{\partial\|V_j(t)\|}{\partial c_j(t)} &= c_j(t)\|V_{j-1}(t)\|^2 + V_{j-1}^T(t)V_{x,j+1}(t) - 2c_j(t)V_{j-1}^T(t)V_{x,j+1}(t) - \\ &- \|V_{x,j+1}(t)\|^2 + c_j(t)\|V_{x,j+1}(t)\|^2 = 0, \end{aligned} \tag{20}$$

несложно получить

$$\begin{cases} c_j(t) = V_{x,j+1}^T(t)\dfrac{V_{x,j+1}(t) - V_{j-1}(t)}{\|V_{x,j+1}(t) - V_{j-1}(t)\|^2}; \\ 1 - c_j(t) = V_{j-1}^T(t)\dfrac{V_{j-1}(t) - V_{x,j+1}(t)}{\|V_{j-1}(t) - V_{x,j+1}(t)\|^2}. \end{cases} \tag{21}$$

Подставив (21) в (19), найдем

$$\left\|V_j(t)\right\| = \left\|(\left\|V_{x,j+1}(t)\right\|^2 - V_{j-1}^T(t)V_{x,j+1}(t))V_{j-1}(t) + \right.$$
$$\left. + (\left\|V_{j-1}(t)\right\|^2 - V_{j-1}^T(t)V_{x,j+1}(t))V_{x,j+1}(t)\right\|^2 \left\|V_{j-1}(t) - V_{x,j+1}(t)\right\|^{-4}. \tag{22}$$

Можно показать, что

$$\begin{cases} \left\|V_j(t)\right\|^2 - \left\|V_{x,j+1}(t)\right\|^2 = -\dfrac{\left(\left\|V_{x,j+1}(t)\right\|^2 - V_{j-1}^T(t)V_{x,j+1}(t)\right)^2}{\left\|V_{j-1}(t) - V_{x,j+1}(t)\right\|^2} \le 0; \\ \left\|V_j(t)\right\|^2 - \left\|V_{j-1}(t)\right\|^2 = -\dfrac{\left(\left\|V_{j-1}(t)\right\|^2 - V_{j-1}^T(t)V_{x,j+1}(t)\right)^2}{\left\|V_{j-1}(t) - V_{x,j+1}(t)\right\|^2} \le 0, \end{cases} \tag{23}$$

т.е. точность объединенного прогноза $\hat{y}_j(t)$ никогда не может быть ниже точности объединяемых прогнозов $\hat{y}_{j-1}(t)$ и $\hat{x}_{j+1}(t)$. Весовой коэффициент $c_j(t)$ задает вклад $\hat{y}_{j-1}(t)$ в $\hat{y}_j(t)$ и тем самым близость реального процесса $x(t)$ к $\hat{y}_{j-1}(t)$ или $\hat{x}_{j+1}(t)$. Изменение значения $c_j(t)$ может служить признаком изменения свойств последовательности $x(t)$, а вектор $C(t) = (c_1(t), c_2(t), \ldots, c_{d-1}(t))^T$ может использоваться в качестве вектора диагностических признаков.

Для работы в режиме текущего времени целесообразно представить (21) в рекуррентной форме. Вводя обозначения

$$\begin{cases} E_j(t) = V_{x,j+1}(t) - V_{j-1}(t); \\ v_{j-1}(t+1) = x(t+1) - \hat{y}_{j-1}(t+1); \\ v_{x,j+1}(t+1) = x(t+1) - \hat{x}_{j+1}(t+1); \\ e_j(t+1) = v_{x,j+1}(t+1) - v_{j-1}(t+1), \end{cases}$$

можно записать

$$\begin{cases} c_j(t) = V_{x,j+1}^T(t)\dfrac{E_j(t)}{\left\|E_j(t)\right\|^2}; \\ c_j(t+1) = V_{x,j+1}^T(t+1)\dfrac{E_j(t+1)}{\left\|E_j(t+1)\right\|^2} = \\ = \dfrac{V_{x,j+1}^T(t)E_j(t) + v_{x,j+1}(t+1)e_j(t+1)}{\left\|E_j(t)\right\|^2 + e_j^2(t+1)}, \end{cases} \tag{24}$$

или окончательно

$$\begin{cases} c_j(t+1) = \dfrac{\Gamma_j(t)}{\Gamma_j(t+1)}c_j(t) + \dfrac{v_{x,j+1}(t+1)e_j(t+1)}{\Gamma_j(t+1)}; \\ \Gamma_j(t+1) = \Gamma_j(t) + e_j^2(t+1). \end{cases} \tag{25}$$

В ряде случаев в алгоритме (25) удобнее использовать не сигналы обновлений, а непосредственно контролируемую последовательность $x(t)$ и ее прогнозы. С учетом того что

$$E_j(t) = X(t) - \hat{X}_{j+1}(t) - X(t) + \hat{Y}_{j-1}(t) = \hat{Y}_{j-1}(t) - \hat{X}_{j+1}(t);$$
$$e_j(t+1) = x(t+1) - \hat{x}_{j+1}(t+1) - x(t+1) + \hat{y}_{j-1}(t) = \hat{y}_{j-1}(t) - \hat{x}_{j+1}(t+1),$$

алгоритм настройки нейронов второго скрытого слоя может быть представлен в виде

$$\begin{cases} c_j(t+1) = \dfrac{\Gamma_j(t)}{\Gamma_j(t+1)}c_j(t) + \dfrac{v_{x,j+1}(t+1)(\hat{y}_{j-1}(t+1) - \hat{x}_{j+1}(t+1))}{\Gamma_j(t+1)}; \\ \Gamma_j(t+1) = \Gamma_j(t) + (\hat{y}_{j-1}(t+1) - \hat{x}_{j+1}(t+1))^2. \end{cases} \tag{26}$$

Таким образом уже на уровне двух слоев предлагаемой искусственной рекуррентной нейронной сети может быть обнаружено изменение свойств диагностической стохастической последовательности путем контроля за вариациями элементов $c_j(t)$. Кроме того, во втором слое решается задача оптимального одношагового прогнозирования последовательности $x(t)$.

Список литературы: 1. *Обнаружение* изменения свойств сигналов и динамических систем: Пер. с англ. / М. Бассвиль, А. Вилски, А. Банвенист и др. М.: Мир, 1989. 278 с. 2. *RombergT.M., Black J.L., Ledwidge T.J.* Signal Processing for Industrial Diagnostics. Chichester: John Wilev&Sons, 1996. 317 p. 3. *Basseville M., Nikiforov I.* Detection of Abrupt Changes: Theory and Application. Englewood Cliffs. New Jersey: PTR Prentice-Hall, 1993. 528 p. 4. *Kerestencioglu F.* Change Detection and Input Design in Dynamical Systems. Tauntont, United Kingdom: Research Studies Press, 1993. 152 p. 5. *Pouliezos A.D., Stravrakakis G.S.* Real Time Fault Monitoring of Industrial Processes. Dordrecht: Kluwer Academic Publishers, 1994. 542 p. 6. *Бодянский Е.В., Руднева И.А.* Об одном адаптивном алгоритме обнаружения разладок в случайных последовательностях // Автоматика и телемеханика. 1995. № 10. С. 101–106. 7. *Бодянский Е.В., Плисс И.П., Соловьева Т.В.* Адаптивное обобщенное прогнозирование многомерных случайных последовательностей // Докл. АН УССР. Сер. А. 1989. № 9. С. 73–75. 8. *Venkatasubramanian V., Chan K.* A neural network methodology for process fault diagnosis // AIChE J. 1989. **35**. P. 1993–2002. 9. *Naidu R.S., Zafiriou E., McAvoy T.J.* Use of neural networks for sensor failure detection in a control system // Inst. of electrical and electronic eng. (IEEE) Control Systems Mag. 1990. **10**. P. 49–55. 10. *Yamashima H., Kumamoto H., Okumura S., Ikesaki T.* Failure diagnosis of a servovalve by neural networks with new learning algorithm and structure analysis // Intern. J. of Production Research. 1990. **28**. N 6. P. 1009 – 1021. 11. *Sorsa T., Koivo H.N., Koivisto H.* Neural networks in process fault diagnosis // IEEE Trans. on System, Man and Cybern. 1991. **21**, N 4. P. 815–825. 12. *Ray A.K.* Equipment fault diagnosis – A Neural network approach // Computers in Industry. 1991. **16**. P. 169 – 177. 13. *Sorsa T., Koivo H.N.* Application of artificial neural network in process fault diagnosis // Automatica. 1993. **29**, N 4. P. 843–849. 14. *Бодянский Е., Воробьев С., Ламонова Н., Штефан А.* Обнаружение изменения свойств стохастических последовательностей на основе искусственных нейронных сетей // АСУ и приборы автоматики. 1997. Вып. 106. С. 75–79. 15. *Connor J.T., Martin R.D., Atlaş L.E.* Recurrent neural networks and robust time series prediction // IEEE Trans. on Neural Networks. 1994. **5**, N 1. P. 240–254. 16. *Aussem A., Murtagh F., Sarazin M.* Dynamical recurrent neural networks – towards environmental time series prediction // Intern. J. Neural Systems. 1995. **6**, N 2. P. 145–170. 17. *Pham D.T., Liu X.* Neural Networks for Identification, Prediction and Control. London: Springer – Verlag, 1995. 238 p. 18. *Chng E.S., Chen S., Mulgrew B.* Gradient radial basis function networks for nonlinear and nonstationary time series prediction // IEEE Trans. on Neural Networks. 1996. **7**, N 1. P. 190–194. 19. *Narendra K.S., Parthasarathy K.* Identification and control of dynamical systems using neural networks // IEEE Trans. on Neural Networks. 1990. **1**, N 1. P. 4–26. 20. *Narendra K.S.* Adaptive control of dynamical systems using neural networks // Handbook of Intelligent Control: Neural, Fuzzy and Adaptive Approaches / Ed. by D.A. White, D.A. Sofge. New York. 1992. P. 141–183. 21. *Marquardt D.* An algorithm for least squares estimation on nonlinear parameters // SIAM J. Appl. Math. 1963. N 11. P. 431–441. 22. *Бодянский Е.В.* Адаптивные алгоритмы идентификации нелинейных объектов управления // АСУ и приборы автоматики. 1987. Вып. 81. С. 43–46. 3. *Goodwin G.C., Ramadge P.J., Caines P.E.* A globally convergent adaptive predictor // Automatica. 1981. **17**, N 1. P. 135–140.

Поступила в редколлегию 17.03.98

Проблемы бионики
1998. Вып. 49. С. 34 — 38

ХТУРЭ

УДК 681.513.7

Е.В. БОДЯНСКИЙ

АВТОМАТИЧЕСКОЕ ОБНАРУЖЕНИЕ РАЗЛАДОК С ПОМОЩЬЮ ИСКУССТВЕННОЙ НЕЙРОННОЙ МЕТАСЕТИ

Эта статья является развитием результатов работы [1]. Предлагаются процедура автоматического обнаружения разладок, основанная на вычислении вероятностей гипотез в выходном нейроне, и архитектура диагностирующей метасети, обладающей более широкими возможностями по сравнению с сетью, описанной в работе [1].

Выходной слой диагностирующей сети образован одним нейроном T_3, в котором производится объединение выходов второго скрытого слоя $\hat{y}(t) = (\hat{y}_1(t), \hat{y}_2(t), \ldots, \hat{y}_{d-1}(t))^T$ в форме

$$\hat{z}(t) = \sum_{j=1}^{d-1} p_j(t)\hat{y}_j(t) = P^T(t)\hat{y}(t). \tag{1}$$

Если на элементы вектора $P(t) = (p_1(t), p_2(t), \ldots, p_{d-1}(t))^T$ наложить ограничения

$$\begin{cases} \sum\limits_{j=1}^{d-1} p_j(t) = P^T(t)I = 1; \\ p_j(t) \geq 0, \quad j = 1, 2, \ldots, d-1, \end{cases} \tag{2}$$

где $I - (d-1) \times 1$ – вектор, образованный единицами, то этим ограничениям можно придать смысл вероятностей определенных гипотез, одна из которых состоит в том, что истинная структура процесса наиболее близка к структуре прогноза $\hat{y}_j(t)$, чья вероятность $p_j(t)$ максимальна.

Для определения диагностического вектора вероятностей *P(t)* введем в рассмотрение лагранжиан

$$\begin{aligned} L(P,\lambda,\mu) &= \sum_{i=1}^{t}\left(x(i) - \sum_{j=1}^{d-1} p_j\hat{y}_j(i)\right)^2 + \lambda\left(\sum_{j=1}^{d-1} p_j - 1\right) - \sum_{j=1}^{d-1}\mu_j p_j = \\ &= \left(X(t) - \hat{Y}(t)P\right)^T\left(X(t) - \hat{Y}(t)P\right) + \lambda\left(P^T I - 1\right) - \mu^T P. \end{aligned} \tag{3}$$

Здесь $\hat{Y}(t) = (\hat{Y}_1(t),\ \hat{Y}_2(t),\ \ldots,\ \hat{Y}_{d-1}(t)) - t \times (d-1)$ – матрица; λ – неопределенный множитель Лагранжа; $\mu - (d-1) \times 1$ – вектор неотрицательных неопределенных множителей Лагранжа, отвечающих условиям дополнительной нежесткости.

Вектор *P(t)* может быть найден либо путем решения системы Куна–Таккера

$$\begin{cases} \nabla_P L(P,\lambda,\mu) = -2\sum\limits_{i=1}^{t} x(i)\hat{y}(i) + 2\left(\sum\limits_{i=1}^{t} \hat{y}(i)\hat{y}^T(i)\right) P + \lambda I - \mu = \\ \qquad = -2\hat{Y}(t)\,X(t) + 2\hat{Y}^T(t)\,\hat{Y}(t)\,P + \lambda I - \mu = 0; \\ \dfrac{\partial L(P,\lambda,\mu)}{\partial \lambda} = \sum\limits_{j=1}^{d-1} p_j - 1 = P^T I - 1 = 0; \\ \dfrac{\partial L(P,\lambda,\mu)}{\partial \mu_j} = -p_j \le 0, \quad \mu_j \ge 0, \quad j = 1,\ 2,\ \ldots,\ d-1, \end{cases} \tag{4}$$

либо, что более удобно при работе в реальном масштабе времени, с помощью процедуры Эрроу–Гурвица, которая в общем случае имеет вид

$$\begin{cases} P(t+1) = P(t) - \gamma_P(t)\nabla_P L(P,\lambda,\mu,t); \\ \lambda(t+1) = \lambda(t) + \gamma_\lambda(t)\partial L(P,\lambda,\mu,t)/\partial\lambda; \\ \mu(t+1) = \mathrm{Pr}_+(\mu(t) + \gamma_\mu(t)\nabla_\mu L(P,\lambda,\mu,t)), \end{cases} \tag{5}$$

где $\gamma_P(t), \gamma_\lambda(t), \gamma_\mu(t)$ – параметры шага поиска; $Pr_+(\bullet)$ – проектор на положительный ортант. Данная процедура, в сущности, является алгоритмом настройки весов выходного нейрона T_3.

С учетом (4) система (5) может быть переписана в виде

$$\begin{cases} P(t+1) = P(t) - \gamma_P(t)(2\xi(t)\hat{y}(t) - \lambda(t)I + \mu(t)); \\ \lambda(t+1) = \lambda(t) + \gamma_\lambda(t)(P^T(t)I - 1); \\ \mu(t+1) = Pr_+(\mu(t) - \gamma_\mu(t)P(t)). \end{cases} \tag{6}$$

Здесь $\xi(t) = x(t) - P^T(t)\hat{y}(t) = x(t) - \hat{z}(t)$ – ошибка прогнозирования выходного слоя сети.

Чтобы оптимизировать скорость процесса настройки выходного слоя, домножим первое соотношение (6) слева на $\hat{y}^T(t)$ и обе части полученного уравнения вычтем из *x(t)*:

$$x(t) - \hat{y}^T(t)P(t+1) = x(t) - \hat{y}^T(t)P(t) - \gamma_P(t)(2\xi(t)\|\hat{y}(t)\|^2 - \\ - \lambda(t)\hat{y}^T(t)I + \hat{y}^T(t)\mu(t)). \tag{7}$$

Выражение в левой части (7) представляет собой апостериорную ошибку $\bar{\xi}(t)$ после одного такта настройки:

$$\bar{\xi}(t) = \xi(t) - \gamma_P(t)(2\xi(t)\|\hat{y}(t)\|^2 - \lambda(t)\hat{y}^T(t)I + \hat{y}^T(t)\mu(t)). \tag{8}$$

Решив уравнение

$$\frac{\partial \bar{\xi}^2(t)}{\partial \gamma_P(t)} = 0, \tag{9}$$

несложно получить оптимальное значение параметра шага поиска

$$\gamma_P(t) = \frac{\xi(t)}{2\xi(t)\|\hat{y}(t)\|^2 - \lambda(t)\hat{y}^T(t)I + \hat{y}^T(t)\mu(t)}. \tag{10}$$

После этого можно записать алгоритм настройки выходного слоя в окончательном виде:

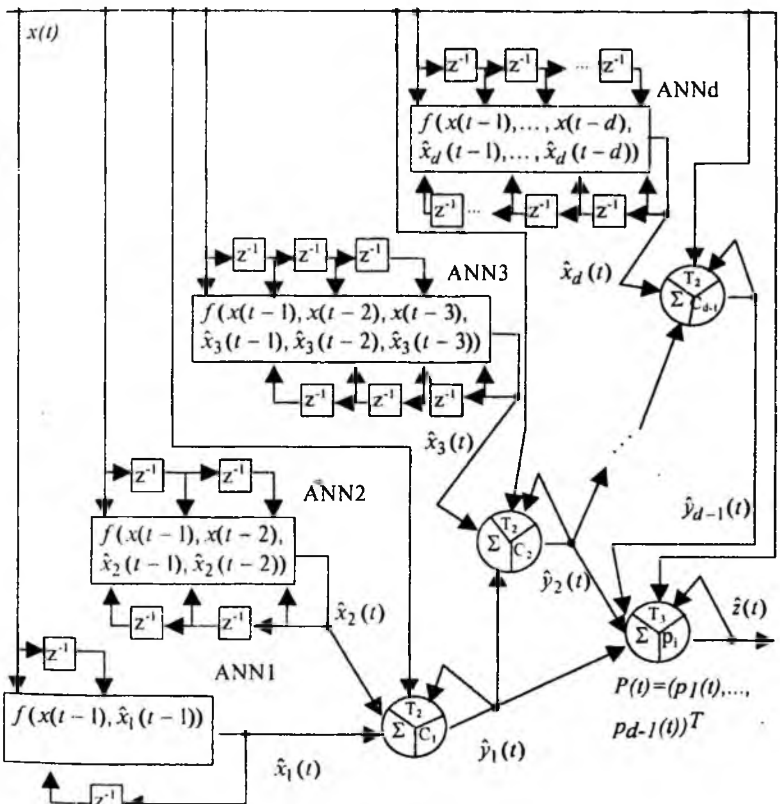
$$\begin{cases} P(t+1) = P(t) + \dfrac{\xi(t)(2\xi(t)\hat{y}(t) - \lambda(t)I + \mu(t))}{2\xi(t)\|\hat{y}(t)\|^2 - \lambda(t)\hat{y}^T(t)I + \hat{y}^T(t)\mu(t)}; \\ \lambda(t+1) = \lambda(t) + \gamma_\lambda(t)(P^T(t)I - 1); \\ \mu(t+1) = Pr_+(\mu(t) - \gamma_\mu(t)P(t)). \end{cases} \tag{11}$$

Если в процессе обучения будут выполнены условия (2), алгоритм (11) автоматически приобретет форму

$$P(t+1) = P(t) + \frac{x(t) - P^T(t)\hat{y}(t)}{\|\hat{y}(t)\|^2}\hat{y}(t), \tag{12}$$

которая представляет собой алгоритм Уидроу–Хоффа, широко применяемый в теории искусственных нейронных сетей [2].

Диагностические и прогностические свойства рассмотренной нейронной сети можно существенно улучшить, если первый скрытый слой сети реализовать не на отдельных нейронах типа T_1, а на нейронных сетях типа многослойного перцептрона, охваченного через элементы чистого запаздывания обратными связями. На возможность и перспективность использования одновременно набора нейронных сетей указывалось и в [3], однако принцип организации взаимодействия сетей не был предложен. На рисунке показана архитектура нейронной метасети, в которой первый скрытый слой сформирован нейронными сетями ANN1, ANN2, ..., ANNd, а взаимодействие между ними организовано через второй скрытый слой. В такой сети высокое качество прогнозирования достигается уже на уровне первого слоя; второй и третий слои служат для определения истинного порядка последовательности *x(t)* и для установления момента его изменения. С вычислительной точки зрения предлагаемая метасеть не намного сложнее рассмотренной выше, поскольку программы, реализующие многослойный перцептрон, входят в состав многих некоммерческих пакетов прикладных программ [4].



Таким образом, нами предложено решение задачи обнаружения свойств стохастических последовательностей, описываемых нелинейными уравнениями авторегрессии – скользящего среднего на основе применения рекуррентных искусственных нейронных сетей. Предложены архитектуры диагностирующих сетей и алгоритмы настройки нейронов, характеризующиеся высоким быстродействием. Описанные сети просты в реализации и обеспечивают высокую скорость обнаружения разладок и точность прогнозирования.

Список литературы: 1. *Бодянский Е.В.* Обнаружение разладок в нелинейных стохастических последовательностях с помощью рекуррентных искусственных нейронных сетей. – См. статью в настоящем сборнике. 2. *Rojas R.* Neural Networks. A Systematic Introduction. Berlin: Springer – Verlag, 1996. 502 p. 3. *Pham D.T., Liu X.* Neural Networks for Identification, Prediction and Control. London: Springer – Verlag, 1995. 238 p. 4. *Braun H., Feulner J., Malaka R.* Practicum Neuronale Netze. Berlin: Springer – Verlag, 1996. 242 S.

Поступила в редколлегию 20.03.98

Проблемы бионики
1998 Вып. 49. С. 39 — 45

ХТУРЭ

УДК 519.7

С.А. ПОСЛАВСКИЙ, В.А. ПОХОДЕНКО,
С.Ю. ШАБАНОВ-КУШНАРЕНКО

УСЛОВИЯ СУЩЕСТВОВАНИЯ ЛИНЕЙНОГО ПРЕДИКАТА И ЕГО СОДЕРЖАТЕЛЬНАЯ ИНТЕРПРЕТАЦИЯ*

Пусть на декартовом квадрате m-мерного векторного пространства M над полем G задан предикат E. Базис $(p_1, p_2, ..., p_m)$ пространства M произвольно фиксирован. Предикат E называется *линейным*, если для любых x, $y \in M$ он может быть выражен в виде

$$E(x, y)=D(F(x), F(y)), \tag{1}$$

где

$$F(x)=(xk_1, xk_2, ..., xk_n), \tag{2}$$

$$xk_i=\xi_1\chi_{1i}+\xi_2\chi_{2i}+...+\xi_m\chi_{mi}. \tag{3}$$

Здесь k_1, k_1, ..., k_n - фиксированные линейно независимые векторы; ξ_1, ξ_2, ..., ξ_m - координаты вектора x; χ_{1i}, χ_{2i}, ..., χ_{mi} - координаты вектора k_i; D - предикат равенства, заданный на $G^n \times G^n$. Имеется в виду, что $n \leq m$. Символом F обозначен линейный оператор, отображающий M в G^n.

Предикат E называется *аддитивным*, если для любых x_1, x_2, y_1, $y_2 \in M$ из x_1Ey_1 и x_2Ey_2 следует $(x_1+y_1)E(x_2+y_2)$. Предикат E называется *однородным*, если для любых $\alpha \in G$ и x, $y \in M$ из xEy следует $\alpha xE\alpha y$. Предикат E называется *n-мерным*, если существуют векторы e_1, e_2, ..., $e_n \in M$ такие, что равенство

$$E(x, \sum_{i=1}^{n} F_i(x)e_i)=1 \tag{4}$$

выполняется для каждого $x \in M$ при единственном наборе коэффициентов $F_1(x)$, $F_2(x)$, ..., $F_n(x)$. Здесь F_1, F_2, ..., F_n - фиксированные функции, определенные на множестве M со значениями в множестве G. Ниже формулируется и доказывается теорема об условиях существования линейного предиката.

Теорема. *Предикат E линеен в том и только том случае, когда он рефлексивен, симметричен, транзитивен, аддитивен, однороден и n-мерен.*
Доказательство. *Необходимость.* Предположим, что предикат E линеен, и

* Статья публикуется в авторской редакции.

выведем отсюда его рефлексивность, симметричность, транзитивность, аддитивность, однородность и n-мерность. Выводим рефлексивность. Для любого $x \in M$ $F(x)DF(x)$. В силу линейности предиката E xEx. Выводим симметричность. Предположим, что $x,y \in M$ таковы, что xEy. Тогда $F(x)DF(y)$, $F(y)DF(x)$, yEx. Выводим транзитивность. Пусть x, y, $z \in M$ таковы, что xEy и xEz. Тогда $F(x)DF(y)$, $F(y)DF(z)$, $F(x)DF(z)$, xEz. Выводим аддитивность. Пусть x_1, x_2, y_1, $y_2 \in M$ выбраны так, что x_1Ey_1 и x_2Ey_2. Отсюда следует, что $F(x_1)DF(y_1)$ и $F(x_2)DF(y_2)$. Ввиду линейности предиката E для любого $i=1,2, ...,n$ имеем $x_1k_i=y_1k_i$ и $x_2k_i=y_2k_i$. Отсюда, пользуясь свойствами арифметического пространства, выводим $x_1k_i+x_2k_i = y_1k_i+y_2k_i$, $(x_1+x_2)k_i = =(y_1+y_2)k_i$. Следовательно $F(x_1+x_2)DF(y_1+y_2)$, а значит, $(x_1+x_2)E(y_1+y_2)$. Выводим однородность. Пусть x, $y \in M$ таковы, что xEy. Это означает, что $xk_i=yk_i$. Для произвольного $\alpha \in G$ имеем $\alpha(xk_i)=\alpha(yk_i)$, $(\alpha x)k_i=(\alpha y)k_i$, Отсюда следует $F(\alpha x)DF(\alpha y)$ и $\alpha xE\alpha y$.

Выводим n-мерность. Выберем векторы e_1, e_2, ..., $e_n \in M$ так, чтобы набор (e_1, e_2..., e_n) был дуален набору векторов (k_1, k_2, ..., k_n). Как известно [1], такой набор всегда существует. Требование n-мерности предиката E означает, что уравнение (4) при каждом $x \in M$ однозначно разрешимо относительно набора коэффициентов ($F_1(x)$, $F_2x)$, ..., $F_n(x)$). Докажем это. Уравнение (4), согласно (1)-(3), равносильно системе уравнений

$$xk_j=(\sum_{i=1}^{n} F_j(x)e_i)k_j,$$

где $j=1,2, ...,n$. Используя законы арифметического пространства, последнюю систему равенств переписываем в виде

$$xk_j=(\sum_{i=1}^{n} F_i(x))(e_ik_j).$$

В силу дуальности наборов векторов (e_1, e_2..., e_n) и (k_1, k_2..., k_n) имеем $e_ik_j=0$ при $i \neq j$ и $e_ik_i=1$ при $i=j$. Поэтому последняя система равенств запишется в виде $F_i(x)=xk_i$. Это означает, что коэффициенты $F_1(x)$, $F_2(x)$, ..., $F_n(x)$ при каждом x однозначно определены. Необходимость доказана.

Достаточность. Предположим, что предикат E рефлексивен, симметричен, транзитивен, аддитивен, однороден и n-мерен, и выведем отсюда его линейность. Докажем сначала, что функции $F_1(x)$, $F_1(x)$, $F_2(x)$, ..., $F_n(x)$, фигурирующие в условии n-мерности, линейны. Для этого нужно убедиться в их адитивности и однородности [2]. В силу n-мерности предиката E для любых x, y M имеет место равенство (4), а также равенства

$$E(y, \sum_{i=1}^{n} F_i(y)e_i)=1, \tag{5}$$

$$E(x+y, \sum_{i=1}^{n} F_i(x+y)e_i)=1. \tag{6}$$

Пользуясь свойством аддитивности предиката E, из (4) и (5) выводим

$$E(x+y, (\sum_{i=1}^{n} F_i(x)+\sum_{i=1}^{n} F_i(y))e_i)=1. \tag{7}$$

В силу n-мерности предиката E множители при векторах e_i в (6) и (7) совпадают, поэтому $F_i(x+y)=F_i(x)+F_i(y)$ при всех $i=1,2, ...,n$. Таким образом, функции $F_1(x)$, $F_2x)$, ..., $F_n(x)$ аддитивны. В силу n-мерности предиката E для любых $\alpha\in G$ и x M имеем:

$$E(\alpha x, \sum_{i=1}^{n} F_i(\alpha x)e_i)=1. \tag{8}$$

Используя однородность предиката E, из (3) выводим

$$E(\alpha x, \sum_{i=1}^{n} F_i(x)e_i)=1. \tag{9}$$

Производим преобразования в последнем равенстве с помощью равенств (5) и (6):

$$E(\alpha x, \sum_{i=1}^{n} (\alpha F_i(x))e_i)=1 \tag{10}$$

В силу n-мерности предиката E множители при векторах e_i в (8) и (9) совпадают, поэтому $F_i(\alpha x)=\alpha F_i(x)$ при любых α и x для всех $i=1,2, ...,n$. Таким образом, функции $F_1(x)$, $F_2(x)$, ..., $F_n(x)$ однородны, следовательно, они линейны. Любая линейная функция $F(x)$ на M со значениями в множестве G может быть представлена в виде $F(x)=xk$ [3], где k - некоторый вектор из M. Значит, найдутся векторы k_1, k_2, ..., k_n G такие, что

$$F_1(x)=xk_1,\ F_2(x)=xk_2,\ ...,\ F_n(x)=xk_n. \tag{11}$$

Докажем далее, что для любых x, y M xEy в том и только том случае, когда $F_i(x)=F_i(y)$ для всех $i=1,2, ...,n$. Пусть xEy. Из этого соотношения и из

(5) с помощью свойства транзитивности предиката E выводим:

$$E(x, \sum_{i=1}^{n} F_i(y)e_i)=1.$$

Из (12) и только что записанного равенства, используя свойство n-мерност предиката E, находим $F_i(x)=F_i(y)$ для всех $i=1,2, ...,n$. Пусть тепер $F_i(x)=F_i(y)$ для всех $i=1,2, ...,n$. Отсюда следует

$$\sum_{i=1}^{n} F_i(x)e_i=\sum_{i=1}^{n} F_i(y)e_i.$$

Из последнего равенства и (5) находим

$$E(y, \sum_{i=1}^{n} F_i(x)e_i)=1.$$

Из (4) и только что записанного равенства с помощью свойств симметри . ности и транзитивности выводим xEy.

Доказанное означает, что условие xEy равносильно равенству $(F_1(x), F_2(x), ..., F_n(x))=(F_1(y), F_2(y), ..., F_n(y))$. Для каждой линейной функции $F: M\to G$ найдется такое $k\in M$, что $F(x)=xk$ для всех $x\in M$. Поэтому $F_i(x)=xk_i$ и $F_i(y)=yk_i$ для любых $x, y\in M$ и всех $i=1,2, ...,n$. Это означает, что условие xEy равносильно равенству $(xk_1, xk_2, ..., xk_n)=(yk_1, yk_2, ..., yk_n)$, т.е. равенству $F(x)=F(y)$. Итак, мы доказали, что любой рефлексивный, симметричный, транзитивный, аддитивный, однородный и n-мерный предикат может быть выражен в виде (1)-(3) при подходящем выборе векторов $k_1, k_2, ..., k_n$. Осталось доказать линейную независимость векторов $k_1, k_2, ..., k_n$. С этой целью установим, что функции $F_1(x), F_2(x), ..., F_n(x)$ линейно независимы. Для этого достаточно доказать, что равенство

$$\sum_{i=1}^{n} \gamma_i F_i(x)=0$$

выполняется для всех $x\in M$ лишь в том случае, когда все коэффициенты γ равны нулю.

Докажем последнее утверждение. Доказательство ведем от противно го. Предположим, что это утверждение неверно. Тогда найдется такой но мер j из множества $\{1,2, ...,n\}$, для которого при любом x

$$F_j(x)= \sum_{i=1,\ j\neq i}^{n} \beta_i F_i(x), \qquad (12$$

где $\beta_1, \beta_2, \ldots, \beta_{j-1}, \beta_{j+1}, \ldots, \beta_n$ - подходящие коэффициенты. Предикат E рефлексивен, значит, $E(e_j, e_j)=1$ для любого $j=1, 2, \ldots, n$. Иначе говоря, $E(e_j, 0e_1+0e_2+\ldots+0e_{j-1}+1e_j+0e_{j+1}+\ldots+0e_n)=1$. По свойству n-мерности предиката E из последнего равенства выводим $F_j(e_j)=1$, для всех же $i\neq j$ имеем $F_i(e_j)=0$. Вместе с тем, подставляя в (12) $x=e_j$, находим $F_j(e_j)=\beta_1F_1(e_j)+\beta_2(e_j)+\ldots+\beta_{j-1}F_{j-1}(e_j)+\beta_{j+1}F_{j+1}(e_j)+\ldots+\beta_nF_n(e_j) = \beta_1 0+\beta_2 0+ +\ldots+\beta_{j-1}0+\beta_{j+1}0+\ldots+\beta_n 0=0$. Получили равенство 1=0. Однако известно [4], что в любом векторном пространстве $1\neq 0$. Мы пришли к противоречию. Отсюда вытекает линейная независимость функций $F_1(x)$, $F_2x)$, ..., $F_n(x)$, а следовательно, и линейная независимость векторов k_1, $k_2\ldots$, k_n. Достаточность доказана. Итак, мы доказали, что все рефлексивные, симметричные, транзитивные, аддитивные, однородные и n-мерные предикаты, и только такие предикаты, могут быть представлены в виде $E(x,y)=D((F_1(x), F_2x), \ldots, F_n(x)), (F_1(x), F_2(x), \ldots, F_n(x)))$, где $F_1(x), F_2(x), \ldots, F_n(x)$ - подходящие линейно независимые линейные функции. Теорема доказана.

Прикладное значение теоремы состоит в том, что она указывает полную систему признаков линейной конечномерной операции F, идентифицируемой компараторным методом. Если предикат E, реализуемый системой компараторной идентификации объекта F, обладает свойствами рефлексивности, симметричности, транзитивности, однородности, аддитивности и n-мерности, то объект идентификации F можно описать в виде линейного оператора. В противном случае это невозможно. Таким образом, теорема указывает практический способ распознавания любого объекта, который можно идентифицировать рассматриваемым способом. Рассмотрим способ практической проверки характеристических свойств линейного объекта на примере зрительной системы человека.

Специфика проверки свойств аддитивности, однородности и n-мерности в случае цветового зрения человека состоит в том, что реально не существует световых излучений со спектрами, имеющими отрицательные компоненты. Кроме того, испытуемому нельзя предъявить излучения слишком большой интенсивности, иначе глаз ослепнет. Таким образом, в множестве M теоретически возможных входных сигналов органа зрения имеется некоторая часть M_1 всех тех световых излучений, которые реально могут быть предъявлены испытуемому в процессе идентификации его зрительной системы. Поскольку законы рефлексивности, симметричности и транзитивности выполняются для всех входных сигналов органа зрения, содержащихся в множестве M_1, то они могут быть чисто формально распространены на все множество M. Производя такое доопределение предиката E, мы не входим в противоречие с фактами зрения.

Закон аддитивности следует проверять на всех тех реальных излучениях x_1, x_2 y_1, y_2 M_1, суммы которых x_1+y_1 *и* x_2+y_2 также не выходят за пределы множества M_1. Кроме того, в роли равновыглядящих излучений можно брать любые равные элементы x и x множества M с физически нереализуемыми спектрами, поскольку закон рефлексивности только что был распространен на все множество M. Таким образом, если реальные излучения x_1 и y_1 выглядят для данного испытуемого равноцветными, то должны выглядеть равноцветными и излучения x_1+x и y_1+x, даже если сигнал x имеет физически нереализуемый спектр с отрицательными компонентами. Единственное ограничение при выборе сигнала x состоит в том, чтобы сигналы x_1+x и y_1+x были физически реализуемы, т.е. входили в состав множества M_1. Если это условие не будет выполняться, то опыт с предъявлением суммарных излучений просто нельзя будет выполнить на практике. Известно, что при всевозможных проверках такого рода закон аддитивности выполняется с той точностью, с которой реализуется предикат E [5]. Этот результат дает нам право, не вступая в противоречие с фактами, распространить действие закона аддитивности на все элементы множества M.

Закон однородности следует проверять на всех тех реальных излучениях $x, y \in M_1$ и числах $\alpha \in G$, произведения которых αx и αy не выходят за пределы множества M_1. Известно [6], что при всевозможных проверках такого рода закон однородности выполняется с той точностью, с которой фактически реализуется предикат E. Опираясь на этот факт, мы распространяем действие закона однородности на все элементы множества M и множества G. Пользуясь законом однородности, можно еще более разнообразить проверку закона аддитивности. Пусть имеются две пары x_1, x_2 и y_1, y_2 одноцветных излучений. Тогда при любых α, $\beta \in G$ элементы αx_1, αx_2 и βy_1, βy_2 следует признать одноцветными, даже если они физически нереализуемы. Если суммарные излучения $\alpha x_1+\beta y_1$ и $\alpha x_2+\beta y_2$ могут быть предъявлены испытуемому (т.е. принадлежат множеству M_1), то они должны породить в сознании испытуемого одинаковые цвета. В практике колориметрических измерений проводились и такого рода опыты, они неизменно подтверждали закон аддитивности [7].

Закон n-мерности при данной интерпретации в эксперименте проверяется при $n=3$. Предположим, что на левом поле сравнения сформировано световое излучение $x+\alpha'_1e_1+\alpha'_2e_2+\alpha'_3e_3$, а на правом - излучение $\alpha''_1e_1+\alpha''_2e_2+\alpha''_3e_3$. Здесь e_1, e_2, e_3 - специально подобранные элементы множества M. В роли этих элементов не обязательно использовать физически реализуемые световые излучения. Коэффициенты α'_1, α''_1, α'_2, α''_2, α'_3, α''_3 также можно брать произвольными из множества G. Символом x обозначен произвольный элемент множества M. В частности, им может быть элемент множества M_1, т.е. физически реализуемое световое излучение. Эксперимент состоит в том, что испытуемый сравнивает цвета полей сравнения и устанавливает их совпадение или несовпадение.

В качестве значений функций $F_1(x)$, $F_2(x)$, $F_3(x)$, фигурирующих в законе трехмерности, следует брать коэффициенты α'_1-α''_1, α'_2-α''_2, α'_3-α''_3. Принимая эти разности коэффициентов, мы следуем методу, указанному Шабановым-Кушнаренко [8]. Закон трехмерности будет выполняться, если для каждого элемента x, принадлежащего множеству M, в эксперименте наблюдается равенство цветов полей сравнения при единственном наборе значений коэффициентов $F_1(x)$, $F_2(x)$, $F_3(x)$. Обширная практика колориметрических измерений [9] показывает, что закон трехмерности выполняется во всех без исключения случаях (для лиц с нормальным зрением, в патологических случаях этот закон выполняется в двумерной или одномерной формулировке) с той точностью, с которой испытуемый реализует предикат E. Коэффициенты $F_1(x)$, $F_2(x)$, $F_3(x)$ называются *координатами цвета*, соответствующего световому излучению x.

Итак, можно с полным основанием утверждать, что все условия, при которых вступает в силу теорема, выполняются применительно к зрительной системе человека. Это означает, что преобразование светового излучения в цвет, осуществляемое органом зрения человека, может быть математически описано в форме линейного оператора, отображающего m-мерное пространство световых излучений, где m - число линий в спектре светового излучения, в трехмерное пространство цветов. Подобно тому, как это только что сделано для цветового зрения человека, с помощью теоремы можно провести структурную идентификацию любого физического, технического или социально-экономического объекта и в результате выяснить, можно ли его отнести к классу конечномерных линейных объектов. Если да, то этот объект можно будет математически описать, пользуясь методами параметрической компараторной идентификации.

Список литературы: 1. *Фаддеев Д.К.* Лекции по алгебре. М.: Наука, 1984. 415 с. 2. *Гельфанд И.М.* Лекции по линейной алгебре. М.: Наука, 1971. 271 с. 3. *Глазман И.М., Любич Ю.И.* Конечномерный линейный анализ. М.: Наука, 1969. 475 с. 4. *Варден Б.Л.* Алгебра. М.: Наука, 1979. 623 с. 5. *Нюберг Н.Д.* Грассмана законы // Физ. энцикл. слов. 1960. Т. 1. С. 136. 6. *Нюберг Н.Д.* Курс цветоведения. М.: Гизлегпром, 1932. 140 с. 7. *Гуревич М.М.* Цвет и его измерение. М.; Л.: Изд-во АН СССР, 1950. 268 с. 8. *Шабанов-Кушнаренко Ю.П.* Математическое моделирование некоторых функций человеческого зрения: Дис.... д-ра техн. наук. Х., 1968. 280 с. Машинопись 9. *Федоров Н.Т.* Общее цветоведение. М.: Гостехтеориздат, 1939. 183 с.

Поступила в редколлегию 24.11.97

Проблемы бионики
1998. Вып. 49. С. 46—53

ХТУРЭ

УДК 519.7

О.Г. ЛЕБЕДЕВ, В.А. ПОХОДЕНКО, С.Ю. ШАБАНОВ-КУШНАРЕНКО

ИДЕНТИФИКАЦИЯ ОБЪЕКТОВ, ОПИСЫВАЕМЫХ ВЕКТОРАМИ*

Многие сигналы естественно представлять в виде векторов, т.е. наборов отдельных компонентов (признаков) [1]. Число компонентов вектора называется его размерностью. В роли компонентов вектора обычно выступают вещественные числа [2]. Множество сигналов, каждый из которых представлен в виде вектора размерности n, называется пространством размерности n. Например, световые излучения обычно представляют в виде спектров, т.е. конечных или бесконечных наборов вещественных чисел. Говорят о двумерности поля зрения человека, имея в виду, что каждая его точка может быть охарактеризована парой вещественных чисел - ее координатами. Говорят также о трехмерном пространстве цветовых ощущений человека, о многомерном пространстве векторов, характеризующих место работы человека (зарплата, продолжительность отпуска, расстояние от места жительства до места работы и т.п.). Действия над векторами чаще всего описываются линейными операциями. Пространство, на котором определены линейные операции, называется *линейным* [3]. Примерами линейных операций могут служить сложение световых излучений, усиление или ослабление интенсивности светового излучения, преобразование светового излучения в цвет его ощущения, оценка человеком места работы и т.п. В этой статье ставится задача отыскания таких полных перечней свойств, с помощью которых можно было бы осуществлять компараторную идентификацию объектов, подпадающих под понятие линейного пространства и под понятие линейной операции над векторами линейных пространств. Разработку сформулированной проблемы начнем с идентификации сигналов, описываемых в виде векторов некоторого пространства.

Пусть A - множество каких-нибудь сигналов. Предположим, что на A определены два предиката эквивалентности $E_1(x, y)$ и $E_2(x, y)$, удовлетворяющие условию:

$$\forall x, y \in A \ (E_1(x, y) \wedge E_2(x, y) \supset x = y). \qquad (1$$

С помощью условия (1) в дальнейшем будет введено понятие двумерного пространства сигналов. Чтобы пояснить, как это делается, мы предварительно решим задачу о введении поля зрения человека. Рассмотрим испы-

* Статья публикуется в авторской редакции.

туемого, голова которого неподвижна в пространстве и находится в вертикальном положении. Один глаз испытуемого смотрит прямо перед собой, его взгляд направлен на неподвижную точку фиксации, второй глаз закрыт. Испытуемому по очереди предъявляют различные точки ξ окружающего физического пространства и предлагают ответить на вопрос, видит он их или нет. В роли предъявляемой точки может использоваться, например, точечный источник света, перемещаемый в пространстве. Пусть M - множество всех точек пространства, $N(\xi)$ - предикат на M, реализуемый испытуемым в данном опыте. Предикат N делит все пространство M на две части: если $N(\xi)=1$, то точка ξ находится в зоне видимости $N\subseteq M$ испытуемого; если же $N(\xi)=0$, то - за ее пределами, т.е. в области $M\backslash N$. Область N имеет вид конуса неправильной формы (приблизительно кругового) с вершиной в оптическом центре глаза.

Из множества N произвольно выбираем две точки ξ и η и предъявляем их испытуемому. Последний должен установить, находятся ли они в точности одна за другой или нет. Своими ответами испытуемый реализует некоторый предикат $Q(\xi, \eta)$ на $N\times N$. Если $Q(\xi, \eta)=1$, то субъективные образы x и y точек ξ и η совмещаются друг с другом в поле зрения испытуемого; если же $Q(\xi, \eta)=0$, то не совмещаются. Опыт показывает, что предикат Q рефлексивен, симметричен и транзитивен (с той точностью, в пределах которой значения предиката Q можно считать однозначными). Действительно, пусть точкам ξ, η и ζ физического пространства N соответствуют их образы x, y, z - точки поля зрения испытуемого. Тогда при совпадении точек ξ и η совпадут и их субъективные образы x и y в поле зрения; если образы x и y двух точек в паре (ξ, η) совпадают, то они совпадут и для пары точек (η, ξ); если образы x, y и y, z точек в парах (ξ, η) и (η, ζ) совпадают, то образы x и z совпадут и для пары точек (ξ, η). Следовательно, предикат $Q(\xi, \eta)$ есть эквивалентность. Он разбивает всю видимую часть N физического пространства M на слои, каждый из которых представляет собой бесконечный луч, исходящий из оптического центра глаза. Каждый такой луч определяет одну точку поля зрения. Множество всех точек поля зрения обозначаем символом A.

Будем предъявлять испытуемому пары (x, y) точек x и y поля зрения, предлагая ему установить, видятся ли они на одной вертикали или нет. В другой серии экспериментов испытуемому предлагается установить, видятся ли точки x и y на одной горизонтали или нет. Важно подчеркнуть, что от испытуемого требуется определить не взаимное положение двух точек физического пространства, а лишь субъективно воспринимаемое взаимное положение образов этих точек. Если испытуемому кажется, что предъявленные ему две точки поля зрения находятся на одной вертикали, то отсюда еще не следует, что соответствующие им точки физического пространства тоже лежат на одной вертикали. Своими ответами испытуемый реализует два предиката $E_1(x, y)$ и $E_2(x, y)$. Если $E_1(x, y)=1$, то точки x и y поля зрения кажутся испытуемому находящимися на одной вертикали, если же $E_1(x, y)=0$, то они кажутся не находящимися на ней. Если $E_2(x, y)=1$, то точки x и y поля зрения воспринимаются лежащими на одной горизонтали, если же $E_2(x, y)=0$, то они лежат на разных горизонталях. Опыты показывают, что предикаты E_1 и E_2 суть эквивалентности. Они формируют два разбиения поля зрения: одно - в виде семейства горизонтальных линий и другое - в виде семейства вертикальных линий. Ясно, что любые вертикальная и горизонтальная линии могут пересекаться не более чем в одной точке поля зрения. Случай, когда вертикаль и горизонталь не пересекаются, возможен: это происходит тогда, когда точка их пересечения попадает в область слепого пятна или же выходит за границы поля зрения. Таким образом, предикаты E_1 и E_2 удовлетворяют условию (1), которое в данной интерпретации гласит: если точки x и y поля зрения A лежат одновременно на одной вертикальной линии и на одной горизонтальной , то они совпадают друг с другом, т.е. $x=y$. Образуем множество B_1 всех вертикальных и множество B_2 всех горизонтальных линий в поле зрения A. Теперь каждую точку x поля зрения A можно представить в виде пары соответствующих ей координат (u_1, u_2), где $u_1 \in B_1$ - вертикаль, проходящая через точку x, и $u_2 \in B_2$ - горизонталь, проходящая через ту же точку. Таким образом, множество точек поля зрения A мы превратили за счет введения двух эквивалентностей E_1 и E_2 в двумерное пространство T, являющееся подмножеством декартова произведения $B_1 \times B_2$ множеств B_1 и B_2.

В общем случае двумерное пространство T для множества A вводится предикатами E_1 и E_2, удовлетворяющими условию (1) следующим образом. Формируем разбиения B_1 и B_2 и характеристические функции $f_1{:}A \to B_1$ и $f_2{:}A \to B_2$, соответствующие эквивалентностям E_1 и E_2. Классы $u_1=f_1(x)$ и $u_2=f_2(x)$ разбиений B_1 и B_2 принимаем в качестве абсциссы и ординаты точки x. Пара (u_1, u_2) однозначно определяет точку x. Совокупность всех пар (u_1, u_2), соответствующих точкам x множества A, принимаем в роли двумерного пространства T для множества A. Пространство T является подмножеством декартова произведения $B_1 \times B_2$ множеств B_1 и B_2. Множества B_1 и B_2 принимаем в роли координатных осей пространства T. В том случае, когда пространство T совпадает с $B_1 \times B_2$, оно называется *полным*. Для полноты пространства T необходимо и достаточно, чтобы эквивалентности E_1 и E_2, его вводящие, дополнительно удовлетворяли условию:

$$\forall x_1, x_2 \in A\ \exists y \in A(E_1(x_1, y) \wedge E_2(x_2, y)). \qquad (2)$$

Если T - полное пространство, то существует сюръекция $g:B_1 \times B_2 \rightarrow A$, взаимно однозначно переводящая векторы (u_1, u_2) пространства T в соответствующие им точки x множества A. Обратный перевод точки x множества A в компоненты u_1 и u_2 ее вектора (u_1, u_2) осуществляется функциями f_1: $A \rightarrow B_1$ и f_2: $A \rightarrow B_2$. Изложенный выше способ введения двумерного пространства легко обобщается на случай пространства произвольной размерности n. Неполное пространство T для множества A вводится эквивалентностями $E_1, E_2, ..., E_n$ на A, удовлетворяющими условию:

$$\forall x, y \in A\ (E_1(x, y) \wedge E_2(x, y) \wedge .... \wedge E_n(x, y) \supset x=y). \qquad (3)$$

Для полноты пространства T необходимо и достаточно, чтобы эквивалентности $E_1, E_2, ..., E_n$, его вводящие, дополнительно удовлетворяли условию:

$$\forall x_1, x_2 \ldots x_n \in A \exists y \in A(E_1(x, y) \wedge E_2(x. y) \wedge .... \wedge E_n(x, y)). \qquad (4)$$

Пространство T для множества A вводится предикатами $E_1, E_2, ..., E_n$, удовлетворяющими условию (3), следующим образом. Формируем разбиения $B_1, B_2, ..., B_n$ и характеристические функции $f_1:A \rightarrow B_1$, $f_2:A \rightarrow B_2$, ..., $f_n:A \rightarrow B_n$, соответствующие эквивалентностям $E_1, E_2, ..., E_n$. Классы $u_1=f_1(x)$, $u_2=f_2(x)$, ..., $u_n=f_n(x)$ разбиений $E_1, E_2, ..., E_n$ принимаем в качестве координат точки x. Набор $(u_1, u_2, ..., u_n)$ однозначно определяет точку x. Совокупность всех наборов $(u_1, u_2, ..., u_n)$, соответствующих точкам x множества A, принимаем в роли n-мерного пространства для множества A. Пространство T является подмножеством декартова произведения $B_1 \times B_2 \times ... \times B_n$ множеств $B_1, B_2, ..., B_n$. Множества $B_1, B_2, ..., B_n$ принимаем в роли координатных осей пространства T. Только в том случае, когда пространство T полно, оно совпадает с декартовым произведением $B_1 \times B_2 \times ... \times B_n$. Любое n-мерное пространство T для множества A полностью характеризуется множествами $B_1, B_2, ..., B_n$, сюръекциями $f_1:A \rightarrow B_1$, $f_2:A \rightarrow B_2$, ..., $f_n:A \rightarrow B_n$ и однозначным отображением g, действующим из $B_1 \times B_2 \times ... \times B_n$ в A. Для полного пространства T отображение g превращается в сюръекцию g: $B_1 \times B_2 \times ... \times B_n \rightarrow A$. Сюръекции $f_1, f_2, ..., f_n$ однозначно определяют эквивалентности $E_1, E_2, ..., E_n$, а наличие однозначного отображения g обеспечивает выполнение условия (3). Наличие же сюръекции g обеспечивает выполнение условия (4).

Пусть A - множество и $T=B_1 \times B_2 \times ... \times B_n$ - его полное пространство. Предположим, что $B_1=B_2=...=B_n=R$ - множество вещественных чисел с заданными на нем операциями сложения и умножения. Множество R называется *числовым полем*, а его элементы - *скалярами*. Множество $T=R^n$ называется *n-мерным арифметическим пространством* [4], если на нем введены операции $x+y$ сложения векторов x, $y \in R^n$ и операция αx умножения вещественного числа $a \in R$ на вектор $x \in R^n$, определяемые следующим образом: если $x=(\alpha_1, \alpha_2, ..., \alpha_n)$ и $y=(\beta_1, \beta_2, ..., \beta_n)$, то

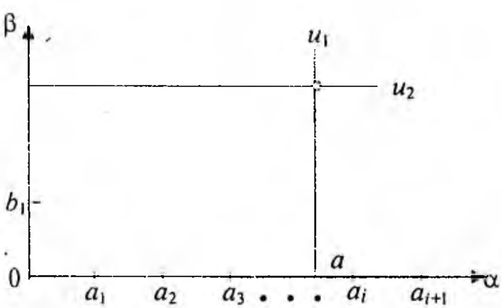
$$x+y=(\alpha_1+\beta_1, \alpha_2+\beta_2, ..., \alpha_n+\beta_n); \tag{5}$$

если $x=(\alpha_1, \alpha_2, ..., \alpha_n)$, то

$$ax=(a\alpha_1, a\alpha_2, ..., a\alpha_n). \tag{6}$$

Из этих определений следует, что для любых α, β, $\gamma \in R$ и x, y, $z \in R^n$ $\alpha+\beta=\beta+\alpha$, $\alpha\beta=\beta\alpha$, $(\alpha+\beta)+\gamma=\alpha+(\beta+\gamma)$, $(\alpha\beta)\gamma=\alpha(\beta\gamma)$, $(\alpha+\beta)\gamma=\alpha\gamma+\beta\gamma$, $0+\alpha=\alpha$, $\alpha+(-\alpha)=0$, $\alpha 1=\alpha$, $\alpha(1/\alpha)=1$, $x+y=y+x$, $(x+y)+z=x+(y+z)$, $x+0=x$, $x+(-x)=0$, $1x=x$, $\alpha(\beta x)=(\alpha\beta)x$, $(\alpha+\beta)x=\alpha x+\beta x$, $\alpha(x+y)=\alpha x+\alpha y$. Здесь $0=(0,0, ...,0)$, $-x= =(-1)x$.

Пользуясь приведенными определениями, покажем, что поле зрения человека можно с определенным приближением идентифицировать как двумерное арифметическое пространство. С этой целью сначала установим, что совокупности B_1 и B_2 всех вертикальных и горизонтальных линий поля зрения A можно отождествить с множествами вещественных чисел. Проведем горизонталь α и вертикаль β через точку фиксации 0 поля зрения (см. рисунок), называя эти линии *координатными осями поля зрения* - осью абсцисс и осью ординат. Поставим во взаимно однозначное соответствие каждой вертикали u_1 точку a ее пересечения с осью абсцисс α и каждой горизонтали u_2 - точку b ее пересечения с осью ординат β. Вместо вертикалей и горизонталей поля зрения теперь будем рассматривать соответствующие им точки на осях абсцисс и ординат. Выберем на оси абсцисс какую-нибудь точку a_1, близкую к точке фиксации 0, но не совпадающую с ней. Затем правее точки a_1 находим на оси α точку a_2, такую



чтобы расстояния между точками 0, a_1 и a_1, a_2 совпали друг с другом. После этого справа от точки a_2 отыскиваем точку a_3, исходя из условия равенства расстояний между точками 0, a_1 и a_2, a_3, и т.д.

Операцию отыскания точки a_{i+1} по точке a_i отождествляем с функцией счета $a_{i+1}=a_i+1$, точку a_1 отождествляем с единицей натурального ряда, а бесконечный ряд точек a_1, a_2,... на оси абсцисс отождествляем со всем натуральным рядом. Все четыре аксиомы натурального ряда, приведенные в работе [5], выполняются. Приближенность такого способа идентификации поля зрения обнаруживается в том, что после выполнения некоторого конечного числа шагов мы доходим до границы поля зрения и ряд точек обрывается. Аналогичным образом на оси α выявляются точки, соответствующие рациональным и вещественным числам, а также выявляются операции над этими точками, которые может производить испытуемый, соответствующие сложению и умножению чисел. Кроме того, выявляется способность испытуемого упорядочивать точки поля зрения на оси абсцисс, которую идентифицируем как отношение порядка на множестве вещественных чисел. Неточность такого способа идентификации состоит в том, что точки, достаточно близкие друг к другу, глазом не различаются ввиду его ограниченной разрешающей способности, кроме того, операции сложения и умножения точек поля зрения на оси α оказываются не всюду определенными, а частичными всякий раз, когда сумма или произведение представляет собой точку, выходящую за пределы поля зрения или попадающую в область слепого пятна. В остальном все аксиомы теории вещественных чисел, перечисленные в работе [6], выполняются. Аналогично, точки оси ординат поля зрения также идентифицируем как вещественные числа. При введении точки b_1, соответствующей единице натурального ряда на оси β, следует учесть, что расстояния между точками 0, a_1 и 0, b_1 должны совпадать. Сложение произвольных точек поля зрения и умножения их на число формализуется с помощью определений (5) и (6). Оказывается, что испытуемый обладает способностью производить такие операции. Возможность производить эти действия в конечном счетѐ основывается на способности испытуемого устанавливать порядок на множестве расстояний между точками поля зрения.

Рассмотрим еще одну необходимую для дальнейшего изложения содержательную интерпретацию арифметического пространства. Речь идет о представлении световых излучений векторами n-мерного арифметического пространства. Как известно, любое световое излучение можно разложить призмой в спектр, т.е. на простые составляющие. Субъективно спектр светового излучения воспринимается как линейно упорядоченнное в поле зрения множество зрительных ощущений разной цветности. Полученный отрезок разбиваем на n участков a_1, a_2, ..., a_n, выбирая число n и размеры участков с таким расчетом, чтобы цветность на каждом участке не менялась. Мощность светового излучения измеряем *болометром*, представляющим собой термометр специальной конструкции. Опыт показывает, что имеет место взаимно однозначное соответствие между показаниями болометра и яркостью зрительного ощущения соответствующей линии в спектре. Этот факт дает возможность идентифицировать совокупность B_i всех яркостей

i-й линии спектра как множество вещественных чисел. Пусть A - множество всех световых излучений. Каждое из множеств B_i ($i=\overline{1,n}$) абстрактно вводится с помощью эквивалентности E_i, определяемой следующим образом: если яркости зрительных ощущений i-й линии спектров двух световых излучений совпадают, то $E_i(x, y)=1$, если же не совпадают, то $E_i(x, y)=0$.

Опыт показывает, что испытуемый способен практически воспроизводить своим поведением каждый из предикатов E_i с довольно высокой точностью. Кроме того, оказывается, что предикаты E_i подчиняются условиям (3) и (4). Следовательно, эквивалентностями E_i можно ввести полное пространство $T=B_1\times B_2\times\ldots\times B_n$ размерности n, соответствующее множеству A. Сложение, умножение и порядок на множествах B_i определяем как соответствующие операции над мощностями соответствующих спектральных линий, рассматривая эти мощности просто как вещественные числа. Сложение световых излучений определяем равенством (5) как покоординатное сложение их спектральных линий. Умножение вещественного числа на световое излучение определяем равенством (6). Физически сложение излучений осуществляется простым совмещением их в пространстве. Умножение числа на световое излучение достигается диафрагмированием светового потока (это число представлено площадью отверстия диафрагмы) или же приближением (удалением) источника света от освещаемой поверхности (в этом случае множитель светового излучения обратно пропорционален квадрату расстояния).

В роли базисных элементов $p_1, p_2, \ldots, p_n$ пространства T можно взять n излучений $p_1(\lambda)$, $p_2(\lambda)$, ..., $p_n(\lambda)$, заданных соответственно на интервалах длин волн $[\lambda_0, \lambda_1]$, $[\lambda_1, \lambda_2]$, ..., $[\lambda_{n-1}, \lambda_n]$. Каждое из этих излучений имеет на интервале своего задания постоянную интенсивность и охватывает единичную площадь. Здесь $[\lambda_0, \lambda_n]$ - диапазон длин волн электромагнитных колебаний, видимых глазом (λ_0=380 нм, λ_n=780 нм) [6]. Пусть $x(\lambda)$ - непрерывный спектр светового излучения x, понимаемый в физическом смысле этого слова. Определим числа $\alpha_1, \alpha_2, \ldots, \alpha_n$ следующим образом:

$$\alpha_i=\sum_{\lambda_{i-1}}^{\lambda_i} x(\lambda)d\lambda. \tag{7}$$

Выбирая в роли n достаточно большое натуральное число и разбивая достаточно равномерно интервал $[\lambda_0, \lambda_n]$ точками $\lambda_1, \lambda_2, \ldots, \lambda_{n-1}$, всегда можно добиться того, чтобы спектр $x(\lambda)$ любого излучения x с требуемой точностью совпал со спектром

$$\sum_{i=1}^{n}\alpha_i p_i(\lambda)$$

аппроксимирующего излучения. Это означает, что с достаточной точностью выполняется равенство

$$x = \sum_{i=1}^{n} \alpha_i p_i(\lambda) , \qquad (8)$$

следовательно числа $\alpha_1, \alpha_2, ..., \alpha_n$ можно принять в роли координат вектора x.

Характер световых излучений таков, что все свойства конечномерного векторного пространства для них выполняются в пределах точности измерения спектров излучений. Правда, реализуются эти свойства на несколько суженной основе. Дело в том, что спектр ($\alpha_1, \alpha_2, ..., \alpha_n$) любого реального излучения не может иметь отрицательных компонентов, поскольку числа $\alpha_1, \alpha_2, ..., \alpha_n$ задают мощность излучения на отдельных участках его спектра, которая всегда неотрицательна. Поэтому выполнение свойств арифметического векторного пространства может быть экспериментально продемонстрировано не для всех векторов пространства T, а только для некоторой его части T_0, называемой *положительным конусом* [8]. Таким образом, операции над скалярами и векторами, введенные в векторном пространстве, в физическом смысле оказываются не всюду определенными, а частичными. Тем не менее ничто не мешает при математических действиях со скалярами и векторами использовать также и физически неинтерпретируемые спектры с отрицательными компонентами, когда это окажется целесообразным. Так например, к спектру реального излучения можно прибавить спектр фиктивного излучения с отрицательными компонентами при условии, что в результате получится спектр излучения из множества T_0, т.е. такого излучения, которое можно физически предъявить испытуемому. Свойства арифметического пространства можно условно распространить на все векторы полного пространства T, но при этом надо иметь в виду, что эти свойства приобретают физический смысл только в том случае, когда в них будут фигурировать лишь векторы из множества T_0.

Список литературы: 1. *Хедли Дж.* Линейная алгебра: Пер. с англ. М.: Наука, 1984. 415 с. 2. *Шилов Г.Е.* Математический анализ (конечномерные линейные пространства). М.: Наука, 1969. 432 с. 3. *Шикин Е.В.* Линейные пространства и отображения. М.: Изд-во МГУ, 1987. 312 с. 4. *Мальцев А.И.* Основы линейной алгебры. М.: Наука, 1970. 392 с. 5. *Баталин А.В., Дударь З.В., Пославский С.А., Шабанов-Кушнаренко С.Ю.* О теории натурального ряда // АСУ и приборы автоматики. 1998. № 107. С. 24-31. 6. *Баталин А.В., Пославский С.А., Шабанов-Кушнаренко С.Ю.* О теории рациональных и вещественных чисел // Там же. С. 32-45. 7. *Мешков В.В.* Основы светотехники: В 2 ч. М.; Л.: Госэнергоиздат, 1961. Ч. 2. 416 с. 8. *Данфорд Н., Шварц Дж.Т.* Линейные операторы, общая теория: Пер. с англ. М.: Изд-во иностр. лит., 1974. 895 с.

Поступила в редколлегию 24.11.97

Проблемы бионики
1998. Вып. 49. С. 54—58

ХТУРЭ

УДК 519.711.3

О.В. АНДРИАНИ, Е.М. РОНИН, В.А. ЧИКИНА

КОМПЬЮТЕРНЫЙ ГРАММАТИЧЕСКИЙ СЛОВАРЬ

За последнее десятилетие мощность и быстродействие выпускаемых компьютеров возросли более чем на порядок, резко увеличился и объем выпуска коммерческого программного обеспечения. Однако общие концепции разработки пользовательского интерфейса программ изменились мало. Нынешние программы общаются с пользователем на том же концептуальном уровне, что и десять лет назад. Изменилось в основном оформление, улучшились визуальные характеристики (утвердился графический пользовательский интерфейс, развились Интернет-технологии).

Наиболее удобной системой диалога человека с компьютером было бы, очевидно, применение естественного языка. Разработки как речевых, так и письменных диалоговых систем ведутся уже давно, и результаты их свидетельствуют о возможности построения качественно новых систем анализа естественноязычных текстов. Но зачастую подобные системы ориентируют лишь на специальные области знания; их авторы ограничивают словарь и отказываются от возможности его расширения в пользу быстродействия и скорости разработки. В данной работе представлен прототип универсальной системы для работы с грамматической информацией на языках с развитым словоизменением.

Для проведения аналитической работы с текстами больших объемов на естественном языке был разработан программный комплекс "Компьютерный грамматический словарь", ориентированный на хранение, анализ и обработку грамматической информации.

Рассмотрим функции его модулей.

Модуль Storehouse обеспечивает управление базой данных (БД) и позволяет производить основные изменения в ее файлах — такие, как вставка и добавление записей, обновление, сортировка и другие служебные операции. Этот модуль обеспечивает высокую производительность и быстродействие программного комплекса в работе с БД.

Модуль программного интерфейса Linx формирует запросы, которые затем передаются для выполнения в модуль Storehouse. Запросы представлены в виде промежуточных таблиц БД, которые содержат исходную информацию при пополнении БД и промежуточный результат поиска. Такая схема передачи данных между приложением и программным комплексом позволяет упростить алгоритм обработки запросов и унифицировать потоки данных.

Модуль Anword обеспечивает анализ поступающей грамматической информации и формирует запросы на ввод-вывод словоформ и парадигм словоизменения. Этот модуль может быть изменен и дополнен в различных пользовательских приложениях, что позволяет сделать обработку информации гибкой и предоставляет возможность создания многоязычных и специализированных приложений.

Модуль интерфейса Userface является необязательным элементом программного комплекса и может корректироваться при подключении различных приложений. Интерфейс пользователя состоит из визуальной системы пополнения БД, включающей в себя формы ввода для всех частей речи русского языка, и независимого пользовательского интерфейса приложения.

БД, с которой работает программный комплекс, состоит из четырех таблиц, содержащих списки основ слов и окончаний, а также матрицу связей между ними и соответствующими грамматическими признаками.

Таблица 1

Номер поля	Название поля	Размерность поля
1	Номер основы	4-байтовое число
2	Основа слова	25 символов
3	Указатель парадигмы	2-байтовое число
4	Часть речи	1-байтовое число
5	Тип словоизменения	1-байтовое число
6	Идентификационный код	4-байтовое число

Табл. 1 хранит основы слов с указанием на соответствующую парадигму, часть речи и тип словоизменения, а также идентификационный код, объединяющий слова с изменяющимися основами.

Таблица 2

Номер поля	Название поля	Размерность поля
1	Номер парадигмы	4-байтовое число
2 . . . n	Набор грамматических признаков	Необходимое количество полей размером 2 байта
n+1	Ссылка на окончание	2-байтовое число

Табл. 2 представляет собой матрицу, которая связывает конкретную парадигму с соответствующими грамматическими признаками и содержит ссылки на окончания.

Таблица 3

Номер поля	Название поля	Размерность поля
1	Номер окончания	2-байтовое число
2	Окончание	6 символов

Табл. 3 хранит окончания вместе с порядковыми номерами. Окончания расположены в таблице в алфавитном порядке и не содержат повторений.

Таблица 4

Номер поля	Название поля	Размерность поля
1	Номер окончания	2-байтовое число
2	Номер парадигмы	4-байтовое число

Кроме того, в БД включена табл. 4, которая является матрицей обратной связи. Табл. 4 содержит номера окончаний и номера соответствующих парадигм словоизменения, что позволяет осуществлять обратный переход от найденного окончания к нужной основе.

Структура БД дает возможность охватывать до 100 тысяч словоформ русского языка и является достаточно гибкой для хранения как регулярных форм, так и исключений.

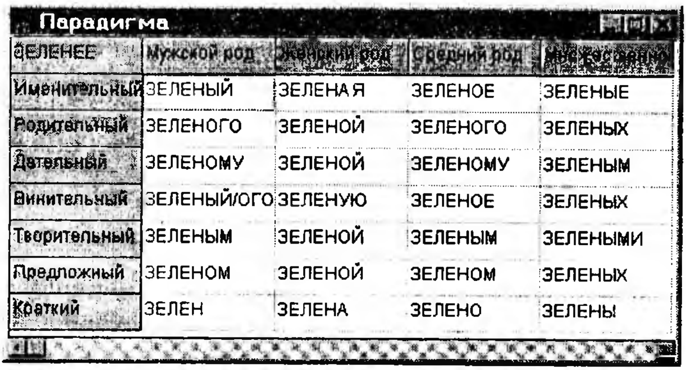
В качестве тестирующего приложения в демонстрационную версию программного комплекса включена подсистема морфологического анализа русских словоформ. Для проведения подобного анализа используется алгоритм деления слова по морфологическому шву "основа — окончание". В соответствии с алгоритмом анализируемое слово делится последовательно справа налево. Истинным считается тот вариант, при котором полученная основа содержится в табл. 1 БД, отделенное окончание принадлежит списку окончаний (табл. 3), а матрица связей (табл. 2) однозначно связывает их друг с другом. После того как процесс деления завершен, найденные варианты с помощью матрицы связей сопоставляются с конкретной грамматической информацией из БД.

Тестирующее приложение работает в диалоговом режиме и может по желанию пользователя переключаться в альтернативные режимы представления грамматической информации — списочный и табличный.

В списочном режиме предоставляется расширенная информация, демонстрируется полный набор грамматических признаков. На рис. 1 показан пример анализа слова «зеленому», введенного в тестирующее приложение. Элементы управления — кнопки «Назад» и «Дальше» позволяют просматривать омонимические варианты слов либо все варианты словоформ данного слова. Кнопки «Парадигма» и «Парадигмы» предназначены для перехода в табличный режим.

В табличном режиме реализуется стандартный способ отображения парадигм словоизменения. На рис. 2 представлены все формы прилагательного «зеленый».

Рис. 1


Парадигма

ЗЕЛЕНЕЕ	Мужской род	Женский род	Средний род	Множественно
Именительный	ЗЕЛЕНЫЙ	ЗЕЛЕНАЯ	ЗЕЛЕНОЕ	ЗЕЛЕНЫЕ
Родительный	ЗЕЛЕНОГО	ЗЕЛЕНОЙ	ЗЕЛЕНОГО	ЗЕЛЕНЫХ
Дательный	ЗЕЛЕНОМУ	ЗЕЛЕНОЙ	ЗЕЛЕНОМУ	ЗЕЛЕНЫМ
Винительный	ЗЕЛЕНЫЙ/ОГО	ЗЕЛЕНУЮ	ЗЕЛЕНОЕ	ЗЕЛЕНЫХ
Творительный	ЗЕЛЕНЫМ	ЗЕЛЕНОЙ	ЗЕЛЕНЫМ	ЗЕЛЕНЫМИ
Предложный	ЗЕЛЕНОМ	ЗЕЛЕНОЙ	ЗЕЛЕНОМ	ЗЕЛЕНЫХ
Краткий	ЗЕЛЕН	ЗЕЛЕНА	ЗЕЛЕНО	ЗЕЛЕНЫ

Рис. 2

В рамках разработки данного приложения создана усовершенствованная система ввода информации в БД, позволяющая вводить ее в автоматизированном и ручном режиме.

Работа с программным комплексом "Компьютерный грамматический словарь" не требует специальных навыков от пользователя, знакомого с интерфейсом Microsoft Windows (tm). Благодаря гибкой системе диалога с пользователем, программный комплекс позволяет легко получать и вводить информацию.

Для функционирования программного комплекса необходим персональный компьютер типа IBM PC AT с процессором 80386 или выше, обладающий не менее чем 4 Мбайт оперативной памяти, видеоадаптером и монитором VGA. Компьютер должен иметь как минимум 1 Мбайт свободного дискового пространства для установки программного комплекса.

Компьютерный грамматический словарь предназначен для работы в среде Microsoft Windows (tm) 3.1 и выше (тестирован в среде MS Windows 3.1, Win'95, Windows NT 4.0). Для функционирования программного комплекса необходимо установить драйверы Borland Database Engine.

Программный комплекс написан для среды компилятора Borland Delphi Client/Server 1.0 и отлажен в ней. Эта система выбрана в связи с тем, что она обладает широким набором средств работы с БД. Кроме того, данная версия создает 16-битные Windows-приложения, работающие в среде MS Windows версии 3.1 и выше.

В результате создания Компьютерного грамматического словаря открываются возможности проведения обширной статистической работы по определению регулярных синтаксических и смысловых конструкций естественноязычных текстов. Такого рода исследования необходимы и для теоретических разработок, посвященных сложным вопросам языкознания, и для решения прикладных задач. Приложения, созданные на базе программного комплекса, можно использовать для идентификации авторства, классификации по предметным областям, а также для построения гибких систем автоматического перевода. Появляется возможность создания специализированных словарей и приложений, предназначенных для автоматизированного аннотирования и реферирования текстов.

Список литературы: 1. *Зализняк А.А.* Грамматический словарь русского языка: Словоизменение. М.: Рус. яз., 1980. 880 с. 2. *Бондаренко М.Ф., Осыка А.Ф.* Автоматическая обработка информации на естественном языке. К.: Учеб.-метод. каб. высш. образования, 1991. 144 с. 3. *Мартин Дж.* Организация баз данных в вычислительных системах: Пер. с англ. М.: Мир, 1980. 662 с.

Поступила в редколлегию 23.01.98

Проблемы бионики
1998. Вып. 49. С. 59—66

ХТУРЭ

УДК 519.7

В.А. ЧИКИНА

СХЕМНАЯ РЕАЛИЗАЦИЯ НЕКОТОРЫХ МОРФОЛОГИЧЕСКИХ ОТНОШЕНИЙ

В работе [1] были математически описаны и схемно реализованы предикаты $P_1(x_1, x_2)$ и $P_2(x_2, x_3)$, характеризующие связи соответственно между первой, второй, третьей буквами окончаний полных непритяжательных имен прилагательных. Обозначим символом $R_1(x_1, x_2, x_3)$ предикат, характеризующий связь между всеми тремя буквами окончания. Зададимся вопросом, совпадает ли предикат $R_1(x_1, x_2, x_3)$ с конъюнкцией $P_1(x_1, x_2) \bullet P_2(x_2, x_3)$ предикатов P_1 и P_2. Если бы такое совпадение имело место, тогда можно было бы отношение $R_1(x_1, x_2, x_3) = 1$ задать просто системой уравнений, которая является объединением систем уравнений, задающих отношения $x_1P_1x_2$ и $x_2P_2x_3$. В роли же схемы, реализующей отношение $R_1(x_1, x_2, x_3) = 1$ (рис. 1, *а*), можно было бы взять последовательное соединение схем, реализующих отношения $x_1P_1x_2$ и $x_2P_2x_3$, (рис. 1, *б*). Предикат R_1 можно представить в виде

$$\begin{aligned}
R_1(x_1, x_2, x_3) = {} & x_1^{\text{а}} x_2^{\text{я}} x_3^{-} \vee x_1^{\text{я}} x_2^{\text{я}} x_3^{-} \vee x_1^{\text{у}} x_2^{\text{ю}} x_3^{-} \vee x_1^{\text{ю}} x_2^{\text{ю}} x_3^{-} \vee \\
& \vee x_1^{\text{о}} x_2^{\text{е}} x_3^{-} \vee x_1^{\text{е}} x_2^{\text{е}} x_3^{-} \vee x_1^{\text{о}} x_2^{\text{г}} x_3^{\text{о}} \vee x_1^{\text{е}} x_2^{\text{г}} x_3^{\text{о}} \vee x_1^{\text{о}} x_2^{\text{м}} x_3^{-} \vee \\
& \vee x_1^{\text{е}} x_2^{\text{м}} x_3^{-} \vee x_1^{\text{о}} x_2^{\text{м}} x_3^{\text{у}} \vee x_1^{\text{е}} x_2^{\text{м}} x_3^{\text{у}} \vee x_1^{\text{о}} x_2^{\text{й}} x_3^{-} \vee x_1^{\text{е}} x_2^{\text{й}} x_3^{-} \vee \\
& \vee x_1^{\text{о}} x_2^{\text{ю}} x_3^{-} \vee x_1^{\text{ы}} x_2^{\text{м}} x_3^{\text{и}} \vee x_1^{\text{и}} x_2^{\text{м}} x_3^{\text{и}} \vee x_1^{\text{ы}} x_2^{\text{х}} x_3^{-} \vee x_1^{\text{и}} x_2^{\text{х}} x_3^{-} \vee \\
& \vee x_1^{\text{ы}} x_2^{\text{м}} x_3^{-} \vee x_1^{\text{и}} x_2^{\text{м}} x_3^{-} \vee x_1^{\text{ы}} x_2^{\text{е}} x_3^{-} \vee x_1^{\text{и}} x_2^{\text{е}} x_3^{-} \vee x_1^{\text{ы}} x_2^{\text{й}} x_3^{-} \vee x_1^{\text{и}} x_2^{\text{й}} x_3^{-} .
\end{aligned} \tag{1}$$

Он задает 26 всевозможных окончаний полных непритяжательных имен прилагательных.

Предикаты P_1 и P_2 в соответствии с табл. 1 и табл. 3 в работе [1] представим следующими формулами:

$$\begin{aligned}
P_1(x_1, x_2) = {} & x_1^{\text{у}} x_2^{\text{ю}} \vee x_1^{\text{ю}} x_2^{\text{ю}} \vee x_1^{\text{о}} x_2^{\text{ю}} \vee x_1^{\text{е}} x_2^{\text{ю}} \vee x_1^{\text{о}} x_2^{\text{г}} \vee x_1^{\text{е}} x_2^{\text{г}} \vee \\
& \vee x_1^{\text{о}} x_2^{\text{м}} \vee x_1^{\text{е}} x_2^{\text{м}} \vee x_1^{\text{о}} x_2^{\text{й}} \vee x_1^{\text{е}} x_2^{\text{й}} \vee x_1^{\text{о}} x_2^{\text{е}} \vee x_1^{\text{е}} x_2^{\text{е}} \vee x_1^{\text{ы}} x_2^{\text{и}} \vee x_1^{\text{и}} x_2^{\text{м}} \vee \\
& \vee x_1^{\text{ы}} x_2^{\text{й}} \vee x_1^{\text{и}} x_2^{\text{й}} \vee x_1^{\text{ы}} x_2^{\text{е}} \vee x_1^{\text{и}} x_2^{\text{е}} \vee x_1^{\text{ы}} x_2^{\text{х}} \vee x_1^{\text{и}} x_2^{\text{х}} \vee x_1^{\text{а}} x_2^{\text{я}} \vee x_1^{\text{я}} x_2^{\text{я}} .
\end{aligned} \tag{2}$$

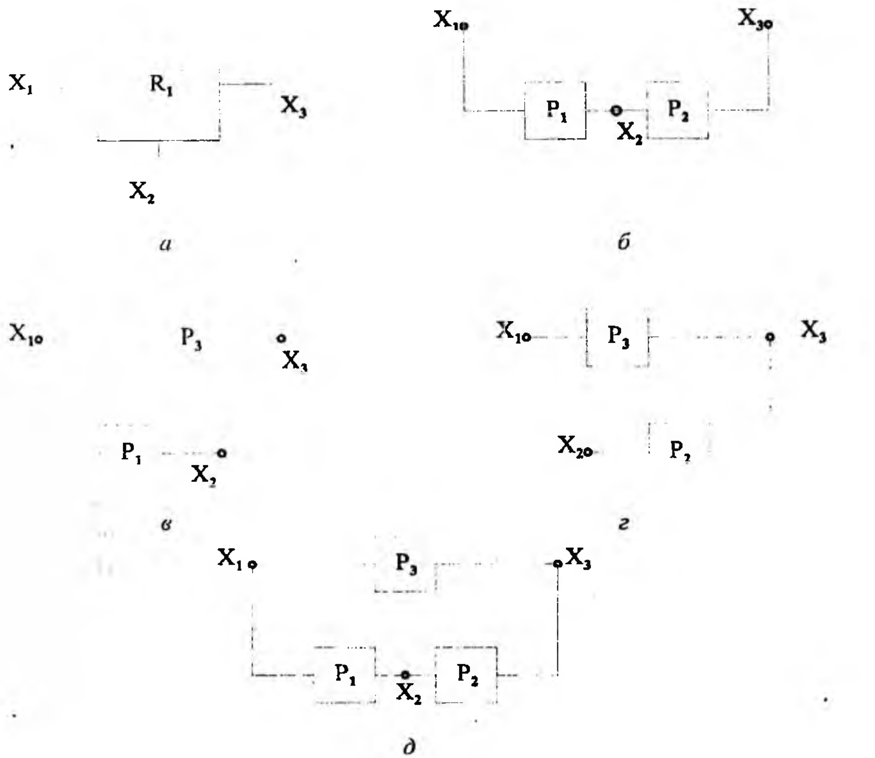


Рис. 1

$$P_2(x_2,x_3) = x_2^{\text{й}}x_3^{-} \vee x_2^{\text{я}}x_3^{-} \vee x_2^{\text{ю}}x_3^{-} \vee x_2^{\text{е}}x_3^{-} \vee x_2^{\text{х}}x_3^{-} \vee x_2^{\text{м}}x_3^{-} \vee \\ \vee x_2^{\text{м}}x_3^{\text{у}} \vee x_2^{\text{м}}x_3^{\text{и}} \vee x_2^{\text{г}}x_3^{\text{о}}. \qquad (3)$$

Производя перемножение предикатов P_1 и P_2, имеем:

$$P_1(x_1,x_2) \bullet P_2(x_2,x_3) = R_1(x_1,x_2,x_3) \vee x_1^{\text{о}}x_2^{\text{м}}x_3^{\text{и}} \\ \vee x_1^{\text{е}}x_2^{\text{м}}x_3^{\text{и}} \vee x_1^{\text{ы}}x_2^{\text{м}}x_3^{\text{у}} \vee x_1^{\text{и}}x_2^{\text{м}}x_3^{\text{у}}. \qquad (4)$$

Итак, видно, что предикат P_1P_2 не совпадает с предикатом R_1. Предикат R_1 охватывается предикатом P_1P_2. Предикат P_1P_2, кроме всех окончаний полных непритяжательных имен прилагательных, охватывает и "лишние" буквосочетания *оми, еми, ыму, иму*. Введем предикат $P_3(x_1, x_3)$, характеризующий связь между первой и третьей буквами окончаний полных непритяжательных имен прилагательных. Он представлен табл. 1. Аналогичным способом убедимся, что предикаты P_1P_3 и P_2P_3 не совпадают с предикатом R_1. Однако предикат $P_1P_2P_3$ совпадает с предикатом R_1. Таким образом, схемы, изображенные на рис. 1, *б-г*, не воспроизводят отношения $R_1(x_1, x_2, x_3) = 1$; схема же, представленная на рис. 1, *д*, воспроизводит его.

Таблица 1

$x_1 \| x_3$	у	ю	а	я	о	е	ы	и
–	1	1	1	1	1	1	1	1
у					1	1		
о					1	1		
и							1	1

$P_3(x_1, x_3)$

Выполним декомпозицию предиката P_3 на функции $u_3 = f_3(x_1)$, $v_3 = g_3(x_3)$ и предикат $G_3(u_3, v_3)$ описанным выше способом. В результате получим следующую систему уравнений:

$u_3^у \sim x_1^у \vee x_1^ю \vee x_1^а \vee x_1^я$ (5); $u_3^о \sim x_1^о \vee x_1^е$ (6); $u_3^ы \sim x_1^ы \vee x_1^и$ (7);

$v_3^у \sim x_3^у \vee x_3^о$ (8); $u_3^у \supset x_{\bar{3}}$ (9); $u_3^о \supset x_{\bar{3}} \vee v_3^у$ (10);

$x_{\bar{3}} \sim u_3^у \vee u_3^о \vee u_3^ы$ (11); $v_3^у \sim u_3^о$ (12); $x_3^и \supset u_3^ы$ (13).

Соответствующая этим уравнениям схема, которая реализует отношение $x_1P_3x_3$, представлена на рис. 2.

При соединении всех трех схем, реализующих отношения $x_1P_1x_2$, $x_2P_2x_3$, $x_1P_1x_3$ (см. рис. 2 и рис. 5, 6 в работе [1]), по способу, указанному на рис. 1, *д*, получим схему, реализующую отношение $R_1(x_1, x_2, x_3) = 1$. Сложность исходных схем составляет соответственно 23, 14 и 18 входов элементов. Сложность получаемой схемы измеряется 49 входами элементов. Уменьшение на 6 числа входов по сравнению с суммарным (23+14+18) достигается за счет совмещения и упрощения элементов: элементы *49* и *50* излишни, поскольку их функции могут выполнять соответственно элементы *11* и *13*, при этом можно принять: $u_3^0 = u_1^0$; $u_3^ы = u_1^ы$.

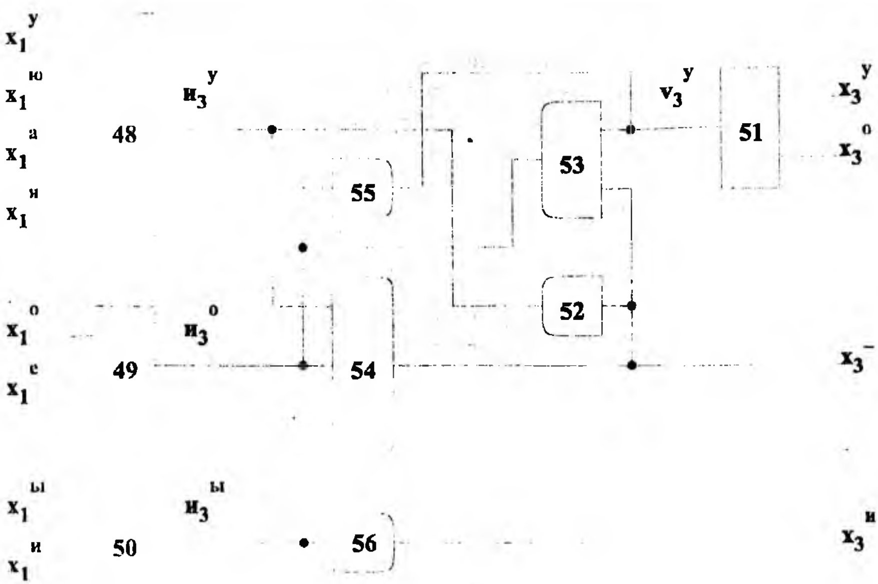


Рис. 2

Вместо четырехвходового элемента *48* можно воспользоваться двухвходовым элементом первого рода, на входы которого подаются выходные сигналы элементов *30* и *12*.

Тернарное отношение $R_1(x_1, x_2, x_3) = 1$ можно образовать, отправляясь от бинарных отношений $x_1P_1x_2$, $x_2P_2x_3$, $x_1P_3x_3$, еще и другим способом [2]. Возьмем, например, отношение $x_2P_2x_3$, представляющее собой следующее множество пар (см. табл. 3 в работе [1]): {*(й,-), (я,-), (ю,-), (е,-), (х,-), (м,-), (м,у), (м,и), (г,о)*}. Каждую пару обозначим каким-либо своим символом: например, пронумеруем пары по порядку. Для номеров пар введем переменную S_1. Связь $R_2(x_2, x_3, S_1) = 1$ между номером S_1 и элементами x_2, x_3 пары (x_2, x_3), соответствующей этому номеру, описывается такими уравнениями:

$x_2^{й}x_3^{-} \sim S_1^1$ (14); $x_2^{я}x_3^{-} \sim S_1^2$ (15); $x_2^{ю}x_3^{-} \sim S_1^3$ (16);

$x_2^{е}x_3^{-} \sim S_1^4$ (17); $x_2^{х}x_3^{-} \sim S_1^5$ (18); $x_2^{м}x_3^{-} \sim S_1^6$ (19);

$x_2^{м}x_3^{у} \sim S_1^7$ (20); $x_2^{м}x_3^{и} \sim S_1^8$ (21); $x_2^{г}x_3^{о} \sim S_1^9$ (22).

Образуем бинарный предикат $P_4(x_1, S_1)$, характеризующий связь между первой буквой окончания x_1 и двухбуквенной частью окончания x_2, x_3, которая представлена своим номером S_1. Предикат P_4 задан табл. 2.

Таблица 2

$x_1 \mid S_1$	3	9	7	1	6	4	5	8	2
у	1								
ю	1								
о	1	1	1	1	1	1			
е	1	1	1	1	1	1			
ы				1	1	1	1	1	
и				1	1	1	1	1	
а									1
я									1

$P_4(x_1,S_1)$

Произведем декомпозицию предиката P_4 на функции $u_4 = f_4(x_1)$, $v_4 = g_4(S_1)$ и предикат $G_4(u_4, v_4)$ изложенным выше способом. В результате получим систему уравнений

$$u_4^{\text{у}} \sim x_1^{\text{у}} \vee x_1^{\text{ю}} \quad (23); \qquad u_4^{\text{о}} \sim x_1^{\text{о}} \vee x_1^{\text{е}} \quad (24); \qquad u_4^{\text{ы}} \sim x_1^{\text{ы}} \vee x_1^{\text{и}} \quad (25);$$

$$S_4^2 \sim x_1^{\text{а}} \vee x_1^{\text{я}} \quad (26); \qquad v_4^9 \sim S_1^9 \vee S_1^7 \quad (27); \qquad v_4^1 \sim S_1^1 \vee S_1^4 \vee S_1^6 \quad (28);$$

$$v_4^5 \sim S_1^5 \vee S_1^8 \quad (29); \qquad u_4^{\text{у}} \supset S_1^3 \quad (30); \qquad u_4^0 \supset S_1^3 \vee v_4^9 \vee v_4^1 \quad (31);$$

$$u_4^{\text{ы}} \supset v_4^1 \vee v_4^5 \quad (32); \qquad S_1^3 \supset u_4^{\text{у}} \vee u_4^0 \quad (33); \qquad v_4^9 \supset u_4^0 \quad (34);$$

$$v_4^1 \supset u_4^{\text{о}} \vee u_4^{\text{ы}} \quad (35); \qquad v_4^5 \supset u_4^{\text{ы}} \quad (36).$$

Введем элемент третьего рода, реализующий отношение

$$x^a y^{\text{в}} \sim z^c . \qquad (37)$$

Варианты его условного обозначения изображены на рис. 3. Из этих элементов построена схема (рис. 4), реализующая систему уравнений (14)—(22), которая задает отношение $R_2(x_2, x_3, S_1) = 1$. Номера элементов соответствуют номерам реализуемых ими уравнений. Сложность схемы измеряется 18 входами элементов. Схема, реализующая систему уравнений (23)—(36), которая задает отношение $x_1P_4S_1$, изображена на рис. 5. Она составлена из элементов первого и второго ряда и всего содержит 27 входов элементов. Соединим цепи, соответствующие предикатам P_2, R_2 и P_4, способом, который указан на рис. 6. Получим схему, реализующую отношение $R_1(x_1, x_2, x_3) = 1$. Сравнивая ее со схемой того же назначения на рис. 1, *д*, видим, что последняя несколько сложнее (59 входов элементов против 49). Таким образом, в данном случае первый способ построения схемы, соответствующей предикату R_1, дал более экономный результат, чем второй. Вместе с тем следует обратить внимание на неуниверсальность первого способа. Дело в том, что далеко не для любого предиката R_1 "треугольная" схема, изображенная на рис. 1, *д*, реализует именно то отношение, которому соответствует предикат R_1. В случае несовпадения предиката R_1 и предиката, реализованного "треугольной" схемой, первый метод просто не позволяет достичь цели. Второй же метод, в противоположность первому, универсален, он всегда приводит к построению искомой схемы.

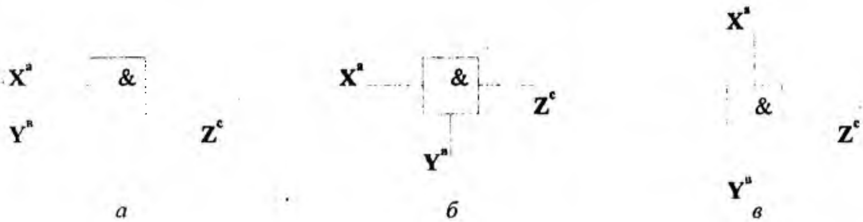


Рис. 3

Выше сформировано и схемно реализовано отношение $R_1(x_1, x_2, x_3) =$ $= 1$, связывающее все три буквы окончания, путем присоединения первой буквы окончания x_1 к паре (x_2, x_3), образованной из его второй и третьей букв. Кроме этого (первого) варианта, возможны еще второй и третий варианты получения отношения R_1 тем же методом, а именно присоединением буквы x_3 к паре букв (x_1, x_2) и буквы x_2 к паре букв (x_1, x_3).

Нами выбран первый вариант, поскольку он настойчиво навязывается языковой интуицией; носитель русского языка ясно ощущает, что первая буква x_1 окончания противопоставлена паре букв (x_2, x_3), например: *е-го, е-му, и-м*. И все же будет нелишним сравнить выводы на основе интуиции с результатами расчета сложности схем, получаемых во всех трех вариантах. Оценим сложность схемной реализации второго варианта. Отношение P_1 содержит 22 пары букв (табл. 1 в работе [1]). Следовательно, схема, присваивающая имена элементам этого отношения, будет содержать 44 входа элементов. Схема, реализующая отношение P_1, содержит 29 входов элементов (рис. 4 в работе [1]). Вместе взятые, эти две схемы дают 73 входа, что значительно больше, чем суммарное число входов (59), необходимое для реализации всех трех схем для первого варианта. Таким образом, второй вариант гораздо сложнее для реализации, чем первый.

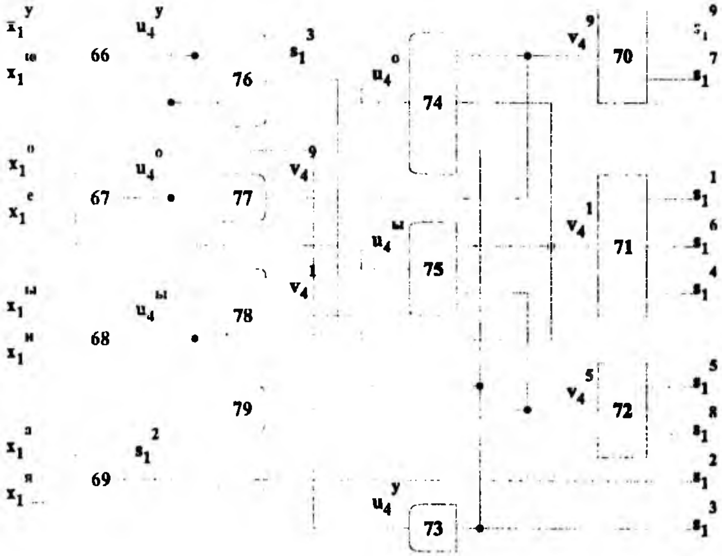


Рис. 4

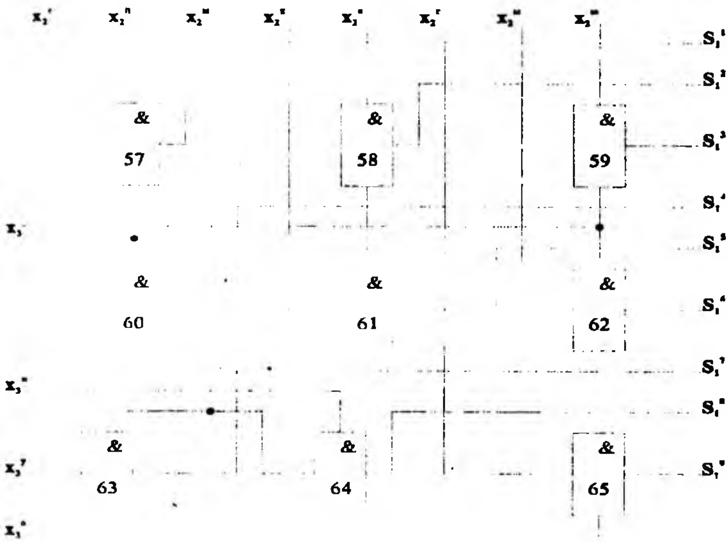


Рис. 5

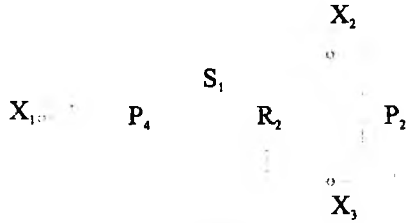


Рис. 6

Перейдем к оценке сложности реализации третьего варианта. Отношение P_3 содержит 14 пар (табл. 4 в работе [1]). Следовательно, схема, присваивающая имена элементам этого отношения, будет содержать 28 входов элементов. Схема, реализующая отношение P_3, содержит 18 входов элементов рис. 2. Для первого варианта требовалось соответственно18 (см. рис. 4) и 13 (рис. 6 в работе [1]) входов элементов, что существенно меньше. Схема, связывающая x_2 с именем пары, содержит 31 вход элементов (сама схема и соответствующие ей уравнения здесь не приводятся). Это больше, чем 27 входов элементов схемы того же назначения для первого варианта (см. рис. 5). Итак, третий вариант по всем показателям оказывается сложнее первого. Поэтому можно утверждать, что выводы, основанные на языковой интуиции, и результаты подсчетов сложности реализации вариантов получения отношения R_1 согласуются друг с другом.

Список литературы: 1. *Бондаренко М.Ф., Чикина В.А.* О методе математического описания морфологических отношений и их схемной реализации // Проблемы бионики. 1998. Вып. 48 С. 3–11. 2. *Шабанов-Кушнаренко Ю.П.* Теория интеллекта. Математические средства. Х. Выща шк. Изд-во при Харьк. ун-те, 1984. 144 с.

Поступила в редколлегию 09.04.98

ХТУРЭ

Проблемы бионики
1998. Вып. 49. С. 67—71

УДК 681.3:621.311

А.Л. ЕРОХИН

РАСПОЗНАВАНИЕ АВАРИЙНЫХ СИТУАЦИЙ В ЭНЕРГОСИСТЕМАХ

Задачи распознавания и таксономии являются предметом повышенного внимания не только математиков, но и специалистов в прикладных областях, в частности энергетиков.

Сложность современных энергосистем обусловливает создание высокоэффективных автоматизированных средств контроля и управления. Для обеспечения бесперебойного снабжения всех видов потребителей электроэнергией необходимо улучшать важнейшие характеристики энергосистем — надежность, живучесть и экономичность.

К основным функциям современных средств управления энергосистемами относятся: сбор, обработка информации, дистанционное управление объектами и взаимодействие с диспетчером. Системы дистанционного управления и сбора данных начали бурно развиваться с конца 60-х гг., благодаря появлению мини-ЭВМ. В практическую электроэнергетику такие системы управления пришли только в 80-х гг., с появлением персональных ЭВМ [1].

Аварийные события в энергосетях современного индустриального мира становятся частыми явлениями и приносят огромные убытки как потребителям энергии, так и их поставщикам. При этом размер материальных убытков находится в обратной зависимости от скорости обнаружения места аварии и ее ликвидации. Поскольку энергосистемы представляют собой распределенные системы, расстояние от диспетчерского пункта до предполагаемого места аварии на линии может измеряться сотнями километров.

Отсюда вытекают следующие задачи современного диспетчерского пункта: распознавание факта аварии в энергосистеме, мгновенная регистрация времени аварии, экспресс-расчет места нахождения аварии, расчет оптимального маршрута проезда ремонтной бригады, установление предполагаемой причины аварии, протоколирование аварийного события.

Для решения задач таксономии и распознавания аварии в энергосистеме необходимо сделать соответствующие постановки задач и систематизировать уже известные методы решения.

Постановка задачи распознавания [2]. Пусть имеется конечный набор ситуаций $X_1, X_2, \ldots, X_n$, относительно которых известно, что они представляют собой реализации k различных неизвестных генеральных совокупностей (образов) S_1, S_2, ..., S_k. Таким образом, существует механизм образования ситуаций, порожденных совокупностью S_i. Этот механизм неизменен, но неизвестен. Существуют стабильные генераторы ситуаций $Г_i$ (i = 1,...,k), генерирующие последовательность ситуаций X_1, X_2, ..., X_n.

Известна обучающая последовательность, на которой задана функция принадлежности

$$i = \phi (j), (i = 1, ..., k; j = 1, ..., N).$$

Последняя определяет, к какому образу i относится ситуация X_j.

Необходимо построить решающее правило (функцию), чтобы на основе имеющихся конечного набора ситуаций и функции принадлежности выявить принадлежность любой другой ситуации к одному из образов. Итак, задача распознавания сводится к определению функции

$$i = F(X),$$

где F – решающее правило, определяющее образ, к которому принадлежит ситуация X.

Для решения задачи нужно получить обучающую последовательность ситуаций, о которой все известно (источник аналогий), и установить степень совпадения ситуаций.

Для этого в пространство ситуаций {X} вводят меру – такую функцию, которая к двум любым ситуациям X_i и X_m однозначно относит число

$$P_{im} = f (X_i, X_m).$$

Подходы к постановке задачи таксономии. Функция принадлежности в данном случае неизвестна (отсутствует источник аналогий). Задача таксономии значительно сложнее, чем задача распознавания, поскольку необходимо сгруппировать ситуации – определить число таксонов k, затем решить задачу синтеза решающего правила F.

С методологической точки зрения задача таксономии относится к задачам образования понятий. Исследование процессов таксономии, их формализация и автоматизация применительно к задачам электроэнергетики на данном этапе ее развития весьма актуальны.

С появлением на рынке ЭВМ современных промышленных контроллеров высокого качества стало возможным в короткие сроки создать высоконадежные и экономичные автоматизированные системы контроля и управления для энергетических объектов (энергосистем) снабжающих предприятий и потребителей энергии.

Важнейшими объектами, определяющими надежность, «живучесть» и экономичность энергосистем, являются подстанции. Большое число устройств сложной релейной защиты и комплексной противоаварийной автоматики нуждается в постоянном контроле правильности их работы. Актуальной задачей остается быстрый анализ аварийных ситуаций на подстанции и прилегающей сети [3].

Для регистрации электрических событий в последнее время начинают разрабатывать и применять цифровые аварийные осциллографы (их еще называют регистраторами аварийных состояний). Основное назначение этих устройств – сбор, первичная обработка и архивирование эксплуатационно-технических параметров как аварийных, так и штатных процессов в основном оборудовании электростанции и на энергетических объектах электроснабжающих предприятий и потребителей сетей [4].

Применение высоконадежных современных контроллеров линии позволило обеспечить высокие параметры цифровых аварийных осциллографов, большую наработку на отказ (свыше 50000 ч) в широком температурном диапазоне, легкую разработку и модификацию программного обеспечения.

Смена программы регистратора производится с диспетчерского компьютера программированием флэш-памяти прибора в течение несколько минут, для чего используются встроенные в контроллер утилиты. Все остальные характеристики каналов измерения прибора задаются с диспетчерского компьютера.

Гибкая модульная структура таких устройств позволяет легко подстраивать их для различных вариантов организации АСУ подстанции. Регистраторы обычно объединяются между собой по схеме «общая шина» или «звезда» и работают независимо от состояния других регистраторов и диспетчерской ЭВМ.

Главными функциями эксплуатируемых в Украине цифровых аварийных осциллографов являются: измерение с определенной частотой (обычно несколько килогерц) текущих значений аналоговых и дискретных сигналов, анализ их на соответствие заданным значениям и запись в энергонезависимую память. Все остальные функции – передача по запросу диспетчера текущего состояния, формирование файла протокола аварии – выполняются в ручном режиме. Некоторые из этих устройств выполняют и автоматический расчет расстояния до места повреждения. Разрешающая способность современных цифровых аварийных осциллографов – обычно 40 – 60 точек на период промышленной частоты при погрешности 0,025 %.

Однако нерешенными остаются наиболее сложные задачи:

– создание концепции предсказания аварийных ситуаций в энергосистемах;

– разработка системы слежения за нарастанием аварии в энергосистеме для своевременного реагирования – выполнения экспресс-анализа аварии;

– повышение точности распознавания факта аварии путем применения сравнительного спектрального анализа;

– разработка методики и подсистемы принятия машинного решения о наличии аварийной ситуации путем сравнения кодов (образа) аварии, хранимых в памяти, и кодов процессов, происходящих в энергосистеме (в реальном масштабе времени);

– определение способа экономного кодирования сигналов в целях увеличения объема хранимых данных об аварии;

– выбор способа синтеза сигналов аварии из кодов для последующего изучения и экспертизы;

– машинный анализ динамики развития аварии («истории развития аварии»);

– организация накопления статистических данных об аварийных и предаварийных ситуациях.

Экспресс-анализ аварии должен включать в себя: 1) быстрый анализ причин аварии с использованием системы распознавания образов аварии (например, должны идентифицироваться короткое замыкание вследствие падения опор линий электропередачи или падение напряжения из-за контакта с выросшим деревом); 2) оценку степени достоверности результатов анализа причины аварии; 3) определение места нахождения аварии или расстояния до места нахождения аварии; 4) выработку рекомендаций диспетчеру относительно принимаемых мер, синтез речевого сообщения персоналу подстанции энергосистемы или электростанции.

После экспресс-анализа аварии образ аварии в виде кода должен храниться в памяти. Образ аварии представляет собой совокупность значений напряжения, силы тока, частоты и гармонических составляющих энергосистемы.

Когда понадобится подробный анализ аварийной ситуации, например для проведения экспертизы или определения виновных, тогда надо обеспечить: вызов из памяти и декодирование образа аварии, воспроизведение «истории развития аварии» за достаточно продолжительный промежуток времени. При этом не должны уменьшаться информативность кода и искажаться образ аварии.

Должно осуществляться хранение сигнала, полученного не только с момента возникновения аварии, но и за предаварийный период. Это даст возможность просмотра и анализа предаварийной ситуации («истории развития аварии»), однако потребует значительного объема памяти для хранения кодов.

Таким образом, нужно искать способы экономного кодирования информации об образе аварии, чтобы сохранять информативность кода, т.е. не искажать образ аварии при декодировании и воспроизведении.

Для решения поставленных задач необходимо создать систему, включающую в себя цифровые осциллографы и современный промышленный компьютер, а также, развивая теорию распознавания образов, разработать эффективные способы распознавания любых аварийных событий в энергосистемах. Это даст возможность реализовать систему поддержки принятия решений при аварийной ситуации в энергосистеме.

Разработка системы для распознавания аварии и поддержки принятия решений позволит: отказаться от услуг большого количества экспертов по авариям, повысить оперативность распознавания аварийного события и скорость устранения аварии, эффективно выявить причину аварии и ее виновника, обеспечить оперативное переключение потребителей энергоснабжения и не допустить отказа всей энергосистемы.

Кроме того, следует учесть, что конечные выключатели, применяемые в электросетях, рассчитаны всего лишь на несколько циклов включения-выключения. Если выявлять и заранее отключать потенциально аварийные части энергосистемы (еще в предаварийном режиме), то можно уменьшить силы токов, протекающих через выключатели, и увеличить ресурс работы последних. Благодаря предсказанию аварийной ситуации можно предотвратить выходы из строя выключателей. В масштабах энергосистемы страны это даст колоссальный экономический эффект.

Наконец, своевременное обнаружение процессов нарастания аварии позволит вовремя спланировать и осуществить перетоки мощностей или отключить аварийные части энергосистемы, уменьшить убытки от аварии и предотвратить развал энергосистемы.

Список литературы: 1. *ЭВМ* в управлении энергосистемами: Темат. вып.: Пер. с англ. М.: Мир, 1987. 173 с. (Тр. Ин-та инж. по электротехнике и радиоэлектронике; Т. 75, № 12). 2. *Месарович М., Такахара Я.* Общая теория систем: Математические основы: Пер. с англ. М.: Мир, 1978. 311 с. 3. *Методы* оптимизации режимов энергосистем / Под ред. В.М. Горнштейна. М.: Энергоиздат, 1981. 336 с. 4. *Макарова О.* Задача для системного интегратора // CompUnity. 1996. N 4. С.91–93.

Поступила в редколлегию 22.01.98

Проблемы бионики
1998. Вып. 49. С. 72—80

ХТУРЭ

УДК 612.821:007

С.И. МАТОРИН

ДЕТЕРМИНАНТНЫЙ АНАЛИЗ СИСТЕМЫ ПЕРЕРАБОТКИ ИНФОРМАЦИИ ЧЕЛОВЕКА

В последнее время в работах по искусственному интеллекту (ИИ) наблюдается принципиальное изменение приоритетов и целей исследований, знаменующее собой смену основной парадигмы этого научного направления. Исследования по ИИ стали проводиться, в первую очередь, с целью "лучше понять функционирование человеческого разума" [1, с. 10]. Таким образом, наконец-то получил признание тот факт, что "в действительности именно человек является основным объектом изучения, и можно быть уверенным, что когда задача будет решена, программы искусственного интеллекта будут иметь самостоятельную ценность независимо от современных компьютеров" [1, с. 18]. Следовательно, можно утверждать, что особенно актуальным становится системный анализ моделируемых средствами ИИ информационных функций человеческого интеллекта в целях последующего детального их сравнения с информационными процессами, посредством которых осуществляется это моделирование.

Функции человеческого интеллекта и информационные процессы в современных компьютерах представляют собой сложные, слабоструктурированные системы. Эффективный анализ сложных объектов и процессов возможен только в рамках системных исследований, точнее, средствами системологии, которая является современным вариантом системного подхода, соответствующим новому, ноосферному этапу развития науки. Системология располагает аппаратом для анализа причины возникновения системы, этапов ее адаптации, а также сущностных свойств, т. е. функциональных свойств, ради наличия и для поддержания которых сформировалась данная система. Этот аппарат использует методологию так называемого детерминантного анализа [2].

В ходе детерминантного анализа изучается главное, функциональное свойство системы, по отношению к которому остальные свойства лишь поддерживают его изнутри (внутренняя детерминанта системы), а также функциональный запрос надсистемы, определяющий выбор внутренней детерминанты, на конкретные взаимодействия рассматриваемой системы с другими системами данной надсистемы, и, следовательно, причина формирования системы (внешняя детерминанта). Введено понятие текущей внутренней детерминанты, которая характеризует существующую в данный момент, достигнутую к данному моменту текущую функциональную способность системы, а также предельной внутренней детерминанты, характеризующей необходимую в конечном счете надсистеме (предельную) функциональную способность данной системы, которая должна реализоваться в процессе адаптации этой системы к запросу надсистемы [2].

Методология детерминантного анализа тесно связана с классификационным анализом, также рассматривающимся как инструмент выявления существенных свойств объектов (систем) [3; 4]. Поэтому последовательное проведение детерминантного анализа системы рекомендуется проводить путем построения трех классификаций: партитивной (или мерономической), генетической (или стадиальной), родовидовой (или таксономической) [2]. Детерминантный анализ с помощью названных классификаций соответствует методологии определения сущности системы [5] и позволяет путем взаимной корректировки этих классификаций в максимальной степени приблизиться к пониманию функционального запроса надсистемы к рассматриваемой системе и таким образом выявить направление ее становления (эволюции или адаптации), а также текущее состояние. Это, в свою очередь, позволяет правильно сформулировать сущностные свойства системы (сущность) и оценить степень достигнутого системой соответствия данным свойствам. Полученные таким образом знания об объекте дают возможность более эффективно анализировать текущую ситуацию, прогнозировать изменение состояний объекта, вырабатывать рекомендации для принятия решений и управления.

Партитивная классификация, в частности, призвана путем анализа частей (компонентов и элементов, т. е. подсистем) данной системы выявить сложившиеся в результате адаптации к запросу надсистемы на данный момент времени внутренние (поддерживающие) свойства системы. Последние, как частные функции системы, поддерживают ее текущие целостные функциональные свойства, и их анализ, таким образом, позволяет оценить текущую внутреннюю детерминанту рассматриваемой системы.

Применим методологию детерминантного анализа к исследованию системы информационных процессов у человека. Для этого, в первую очередь, рассмотрим партитивную (мерономическую) классификации названных процессов.

Современная антропология рассматривает человека в целом [6] как элемент антропосферы, являющейся подсистемой биосферы, которая обусловливает в значительной степени существование, развитие и поведение человека. Кроме того, при анализе антропосферы выделяют ноосферу и социосферу, "определяемые отношениями людей в обществе, связанными как законом сохранения биосферы и его следствиями (инстинктами), так и разумом, свойственным исключительно человеку..." [6, с. 140]. Следовательно подсистемы человека должны поддерживать его функционирование в биотической, психической и социальной сферах.

В связи с этим представляет интерес, на наш взгляд, рассмотрение, по аналогии с подходом в [7], в качестве подсистем системы «человек» компонентов сенсомоторной активности, обеспечивающих управление функционированием системы в различных ситуациях. Сенсомоторное управление образует замкнутую схему функционирования системы, качество которого подчинено процессу ее эволюции или степени ее адаптации. Каждый уровень управления в этой схеме образуется за счет взаимодействия трех компонентов (блоков): сенсорного распознавания, инстанции принятия решений и моторной активности.

В отличие от подхода в [7] предлагается рассматривать эти компоненты на нескольких уровнях, обусловленных существующими у человека уровнями отражения (сенсорного распознавания), мышления (инстанции принятия решений) и поведения (моторной активности). В соответствии с данными теории отражения и психологии мышления [8–10] целесообразно учитывать четыре уровня сенсомоторной активности, которые и предлагается рассматривать как подсистемы, образующие систему «человек». Показательно, что похожий результат получен экспериментальной психологией при создании модели уровней построения движений в иерархической структуре когнитивных процессов [11]. Рассмотрим более подробно эти подсистемы (рисунок), каждая из которых состоит из трех упомянутых блоков.

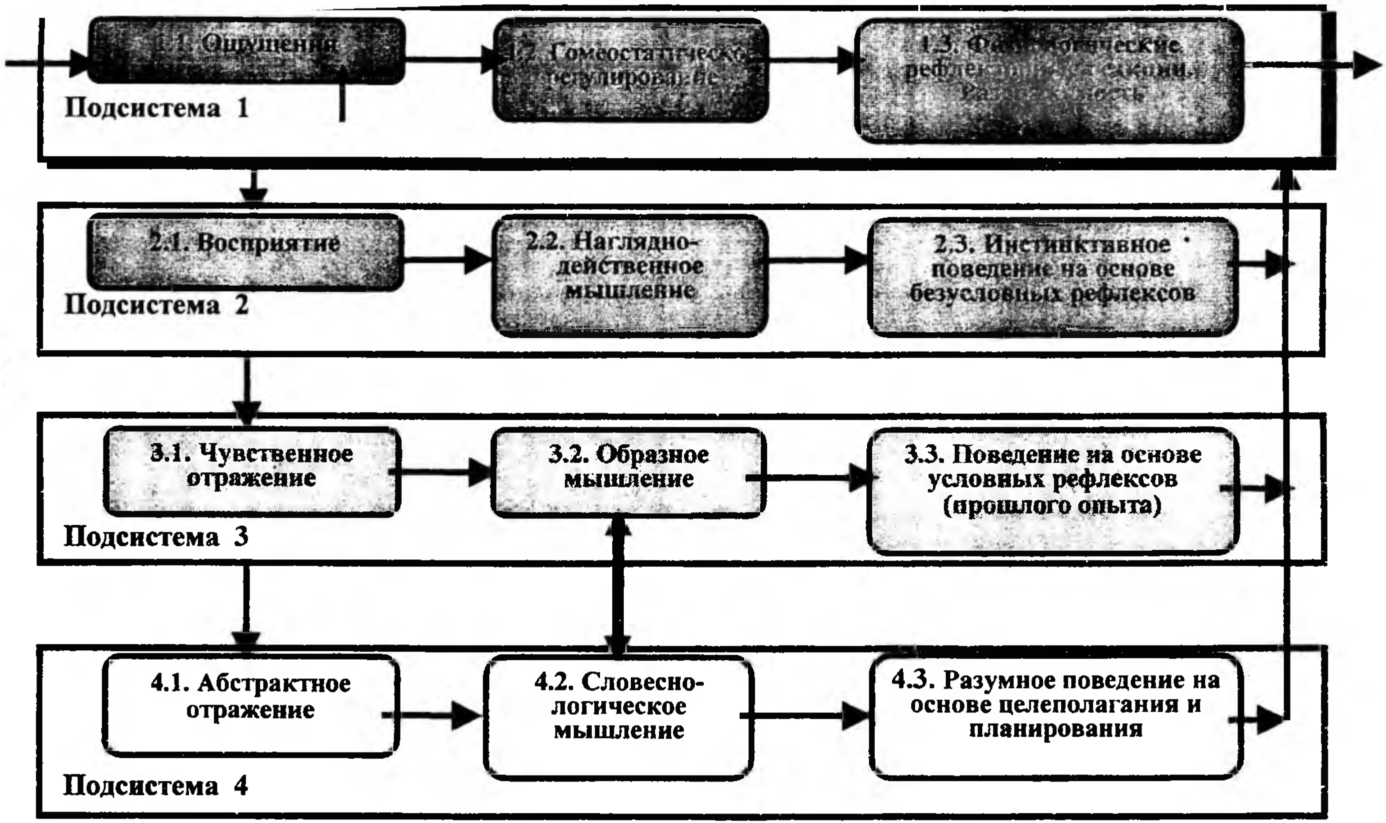
Подсистема 1. Блоки:

1.1. Прием, сенсорное распознавание физических воздействий внешней и внутренней среды на уровне отдельных ощущений (зрительных, слуховых, обонятельных, осязательных, вкусовых).

1.2. Гомеостатическое регулирование, т. е. поддержание постоянства характеристик внутренней среды подсистемы, для обеспечения самосохранения.

1.3. Моторная активность в виде проявлений раздражимости, физических отправлений и физиологических рефлекторных реакций (сердцебиение, дыхание, потоотделение, сужение и расширение сосудов и зрачков, сокращение мышц, работа системы выделения и т.д.) в целях поддержания постоянства характеристик внутренней среды.

В эту подсистему включаются, таким образом, только физико-химические процессы, происходящие в системе «человек», — от приема входных воздействий до физиологических реакций и физической активности


Ощущения
1.2. Гомеостатическое регулирование
1.3.
Подсистема 1
2.1. Восприятие
2.2. Наглядно-действенное мышление
2.3. Инстинктивное поведение на основе безусловных рефлексов
Подсистема 2
3.1. Чувственное отражение
3.2. Образное мышление
3.3. Поведение на основе условных рефлексов (прошлого опыта)
Подсистема 3
4.1. Абстрактное отражение
4.2. Словесно-логическое мышление
4.3. Разумное поведение на основе целеполагания и планирования
Подсистема 4

на выходе. В ней самой управление основано на гомеостатическом механизме поддержания постоянства характеристик внутренней среды, который представляет собой конкретизацию для данного вида организмов инстинкта самосохранения.

Подсистема 2. Блоки:

2.1. Восприятие текущих сигналов различной модальности на основе сенсорного распознавания воздействий подсистемой 1, а также распознавание заложенных от рождения сигналов. Формирование целостного конкретного образа текущей ситуации.

2.2. Выработка и принятие решений в связи с текущей ситуацией за счет различения жизненно важных ее характеристик, в том числе удовольствия и боли, средствами наглядно-действенного мышления, осуществляемого с помощью реального, физического преобразования ситуации и опробования свойств объектов.

2.3. Активность в текущей ситуации в виде инстинктивного поведения на основе врожденных безусловных рефлексов (пищевого, полового и т.д.).

Данная подсистема обеспечивает первичную информационную (семантическую) обработку воздействий, принятых подсистемой 1. Она представляет собой, таким образом, уже информационный процесс, в ходе которого физико-химические явления, сопровождающие воздействия среды на предыдущую подсистему, интерпретируются как сигналы, т. е. как информация. Последняя используется в ходе наглядно-действенного анализа текущей ситуации, а также для реализации безусловных, заложенных от рождения рефлексов.

Подсистема 3. Блоки:

3.1. Чувственное отражение сенсорной информации, поступающей из подсистемы 2. Оно состоит в распознавании конкретных целостных образов ситуации "по ассоциации по сходству" [12] с обобщенными образами. Предварительное формирование обобщенных образов, имеющих сходство с отражаемыми объектами (ситуациями), осуществляется путем наложения друг на друга текущих образов и запоминания повторяющихся деталей с помощью так называемого механизма суммации [13].

3.2. Выработка и принятие решений средствами образного мышления, осуществляемого с помощью образного представления ситуации и воображения ее изменений с учетом многообразия различных воспринятых и отраженных характеристик объектов.

3.3. Активность в виде мотивированного поведения на основе условных приобретенных рефлексов (прошлого опыта).

Данная подсистема представляет собой также информационный процесс, в ходе которого сенсорная информация, полученная в предыдущей подсистеме, подвергается вторичной семантической обработке. На основе этой информации формируются и сохраняются чувственно воспринимаемые, так называемые обобщенные образы, которые используются образным мышлением, обеспечивающим существование и функционирование чувств (эмоций, желаний, стремлений), волевых импульсов, оценок, воображения, учет прошлого опыта и условно-рефлекторную деятельность.

Подсистема 4. Блоки:

4.1. Абстрактное отражение информации, поступающей из подсистемы 3. Оно заключается в распознавании образов, отражающих не обязательно чувственно воспринимаемые свойства объектов окружающей среды, "по ассоциации по смежности" [12] с образами, имеющими знаковую природу. Предварительное формирование абстрактных (знаковых) образов, в первую очередь слов естественного языка – понятий, осуществляется путем выдвижения и проверки гипотез о существенных свойствах объектов посредством так называемого механизма активного поиска [13].

4.2. Выработка и принятие решений средствами словесно-логического мышления, реализуемого с помощью понятий и логических конструкций на базе языковых средств, а также с помощью обобщения путем абстрагирования.

4.3. Активность в виде сознательного, рационального поведения на основе целеполагания, с учетом планирования и прогнозирования развития и изменения данной ситуации.

Данная подсистема представляет собой также информационный процесс, в ходе которого информация, полученная в предыдущей подсистеме, подвергается дальнейшей обработке. На основе этой информации формируются и сохраняются чувственно не воспринимаемые, так называемые абстрактные образы, которые используются словесно-логическим мышлением, обеспечивающим понимание, работоспособность интеллекта (синтез идей и выработку гипотез), прогнозирование, планирование и, таким образом, рациональное, сознательное, разумное, целенаправленное поведение (самосознание).

Анализ схемы уровней сенсомоторной активности человека показывает, что каждая следующая по номеру подсистема, представляя собой, конечно, новое качество, не может, однако, функционировать без информации, вырабатываемой предыдущей подсистемой. Можно даже утверждать, что каждая следующая подсистема является эволюционной надстройкой над предыдущей. Например, целеполагание и планирование поведения на уровне подсистемы 4 осуществляется благодаря наличию определенных оценок и мотивов на уровне подсистемы 3. Соответственно эти оценки и мотивы поведения являются результатом наличия на уровне подсистемы 2 возможности восприятия удовольствия и боли. Распознавание удовольствия и боли основано на способности обеспечивать постоянство характеристик внутренней среды подсистемой 1. Кроме того, каждая следующая подсистема обеспечивает бо́льшую степень активности и свободы действий, чем предыдущая, и, таким образом, возможность решения все более разнообразных и более сложных задач системой в целом, а следовательно, бо́льшую устойчивость системы к внешним воздействиям.

Функции и особенности уровней сенсомоторной активности могут быть уточнены на примерах. Одноклеточные, например, представляют собой только подсистему 1, насекомые – совокупность подсистем 1 и 2, птицы и многие млекопитающие имеют подсистемы 1–3. Наличие у организма следующей по номеру подсистемы повышает качество и эффективность функционирования предыдущей. Необходимо отметить, что у разных насекомых или млекопитающих качество подсистемы верхнего уровня различается и соответствует уровню развития данного вида. Эволюция органического мира может быть представлена как процесс повышения качества имеющихся уровней сенсомоторной активности и добавления новых уровней по мере усложнения среды обитания.

Совершенствование активности (поведения) осуществляется: на уровне подсистемы 2 – в процессе смены поколений путем накопления положительных для данной среды изменений у целого вида; на уровне подсистемы 3 – путем приобретения положительного (в данных условиях существования) опыта индивидуумом, т. е. в процессе обучения, которое может происходить и без привлечения опыта других индивидуумов. Нормальное функционирование и совершенствование подсистемы 4 обеспечивается только путем индивидуального обучения, в ходе которого обязательно привлекаются знания, накопленные предыдущими поколениями.

Естественно предположить, что группы людей, значительно отличающиеся друг от друга по историческим, климатическим и географическим условиям возникновения (расы), имеют различия в особенностях функционирования подсистемы 2; отличающиеся друг от друга условиями жизнедеятельности (нации и народности), – различия на уровне подсистемы 3, которые затрагивают передаваемые в процессе обучения нового поколения оценки, традиции, нормы поведения и т. д.; отличающиеся друг от друга своей социальной (профессиональной) функцией – различия на уровне подсистемы 4.

Правильным (гармоничным) алгоритмом функционирования рассмотренной системы переработки информации является единовременный учет в любой ситуации состояния внешней и внутренней среды, текущей обстановки, накопленного прошлого опыта, а также целей функционирования с учетом прогнозирования будущего развития и изменения ситуации. В настоящее время, однако, алгоритм, в соответствии с которым осуществляется жизнедеятельность человека, как правило, не обеспечивает гармоничного функционального единства рассмотренных подсистем. Современные данные о человеке и человеческом сообществе свидетельствуют о том, что их основными чертами являются рациональность, интеллектуальность и превращение продуктов их деятельности – науки и технологии – в геологический фактор эволюции всей планеты [14]. Более того, наличие рационального сознания, интеллекта, словесно-логического мышления часто рассматривается как отличительный видовой признак человека [6; 7]. Следовательно, на сегодняшний день основная функциональная способность (текущая внутренняя детерминанта) человека (как вида) заключается в рациональном разумном поведении, которое основано на абстрактном отражении, словесно-логическом мышлении, целеполагании и планировании.

Предлагаемая партитивная классификация, согласуясь с современными научными данными, хорошо согласуется и с представлениями мыслителей прошлого о человеке, зафиксированными так называемой эзотерической психологией [15; 16].

Основным следствием структурирования интеллектуальных функций человека является осознание необходимости решения, наконец, вопроса о допустимости и, таким образом, целесообразности создания искусственного интеллекта, тождественного естественному по функциональным свойствам. Создание компьютерной системы с внутренней детерминантой, тождественной человеческой, превращает ее в систему того же уровня, что и человек. Внешняя детерминанта такой системы будет тождественна человеческой либо будет являться запросом той же самой надсистемы. Следовательно, построению действительно интеллектуальных компьютерных систем должны предшествовать работы по изучению на основе системологического подхода внешней детерминанты и надсистемы естественного интеллекта.

Список литературы: 1. *Лорьер Ж.-Л.* Системы искусственного интеллекта: Пер. с фр. М.: Мир, 1991. 568 с. 2. *Разработка* подсистемы лингвистического обеспечения АИС документально-фактографического типа в области материаловедения: Отчет о НИР / Спец. конструкт.-технол. бюро информ. систем Ин-та пробл. материаловедения (ИПМ); Руководитель Г.П. Мельников; № ГР 01.86.0103243. М., 1986. 452 с. 3. *Мельников Г.П.* Функции разума в биосфере и его технические усилители. К., 1990. 24 с. (Препр. АН УССР. Ин-т кибернетики; 90-56). 4. *Бреховских С. М.* Основы функциональной системологии материальных объектов. М.: Наука, 1986. 192 с. 5. *Маторин С.И.* Методология объективного определения сущности системы. Х., 1995. 27 с. Деп. в ГНТБ Украины 20.02.95, № 487- Ук.95. 6. *Косыгин Ю.А.* Человек. Земля. Вселенная. М.: Наука, 1995. 335 с. 7. *Кликс Ф.* Пробуждающееся мышлени История развития человеческого интеллекта: Пер. с нем. К.: Выща шк. Изд-во при Киев. у те, 1985. 295 с. 8. *Психология*: Слов./ Под общ. ред. А.В. Петровского, М.Г. Ярошевского. М Политиздат, 1990. 494 с. 9. *Современная* философия: Слов. и хрестоматия. Ростов н/Д: Фе никс, 1996. 511 с. 10. *Тихомиров О.К.* Психология мышления. М.: Изд-во МГУ, 1984. 271 с. 1 *Величковский Б.М.* Когнитивная наука и психологические проблемы изучения интеллекта Компьютеры и познание: Очерки по когитологии. М., 1990. С. 6–21. 12. *Мельников Г.П.* Системоло гия и языковые аспекты кибернетики. М.: Сов. радио, 1978. 368 с. 13. *Соловьев А.В.* Эксперимен тальное исследование психологических механизмов формирования понятий: Автореф. дис. канд. техн. наук. М., 1973. 20 с. 14. *Вернадский В.И.* Научная мысль как планетное явлени М.: Наука, 1991. 271 с. 15. *Ошо Р.* Психология эзотерического. Корни и крылья. М.: АС 1992. 435 с. 16. *Человек* и его видимый и невидимый состав. Х.: Инарт, 1991. 55 с.

Поступила в редколлегию 26.05.9

Ɂроблемы бионики
998. Вып. 49. С. 81—89

УДК 612.821:007

Д. А. ГЕВЧУК, С. И. МАТОРИН

СРАВНИТЕЛЬНЫЙ АНАЛИЗ СИСТЕМ ПЕРЕРАБОТКИ ИНФОРМАЦИИ КОМПЬЮТЕРА И ЧЕЛОВЕКА

Считается, что "мы уже подошли к границе, за которой системы искусственного интеллекта вплотную приблизятся к человеку, а в чем-то и превзойдут человеческие возможности" [1, с. 119]. Эта ситуация требует "придания естественности порождаемым нами искусственным системам" и "снова выдвигает вопрос об определении самого понятия «разумности»" [1, с. 120— 121]. Поэтому не случайно в 80 — 90-е гг. предмет и содержание искусственного интеллекта (ИИ) подверглись серьезному переосмыслению. Ведущая тенденция связана с возрастанием удельного веса коннекционизма и эволюционного моделирования в структуре исследований по ИИ [2].

Названные обстоятельства свидетельствуют о том, что актуальной задачей является сравнительный анализ функций человеческого интеллекта, моделируемых средствами ИИ, и компьютерных информационных процессов, посредством которых осуществляется это моделирование. Результаты такого сравнительного анализа могут оказаться полезными при решении конкретных задач интеллектуализации компьютерных систем и технологий.

Сравнительный анализ систем переработки информации человека и современного компьютера может быть проведен путем сравнения их партитивных (целочастных) классификаций. Для этого необходимо представить компьютерную систему в виде подсистем, соответствующих уровням сенсомоторной активности, т. е. в виде, подобном партитивной классификации человека, приведенной в предыдущей статье [3]. Рассмотрим эти подсистемы (рисунок), каждая из которых состоит из блоков (компонент), аналогичных описанным в упомянутой статье.

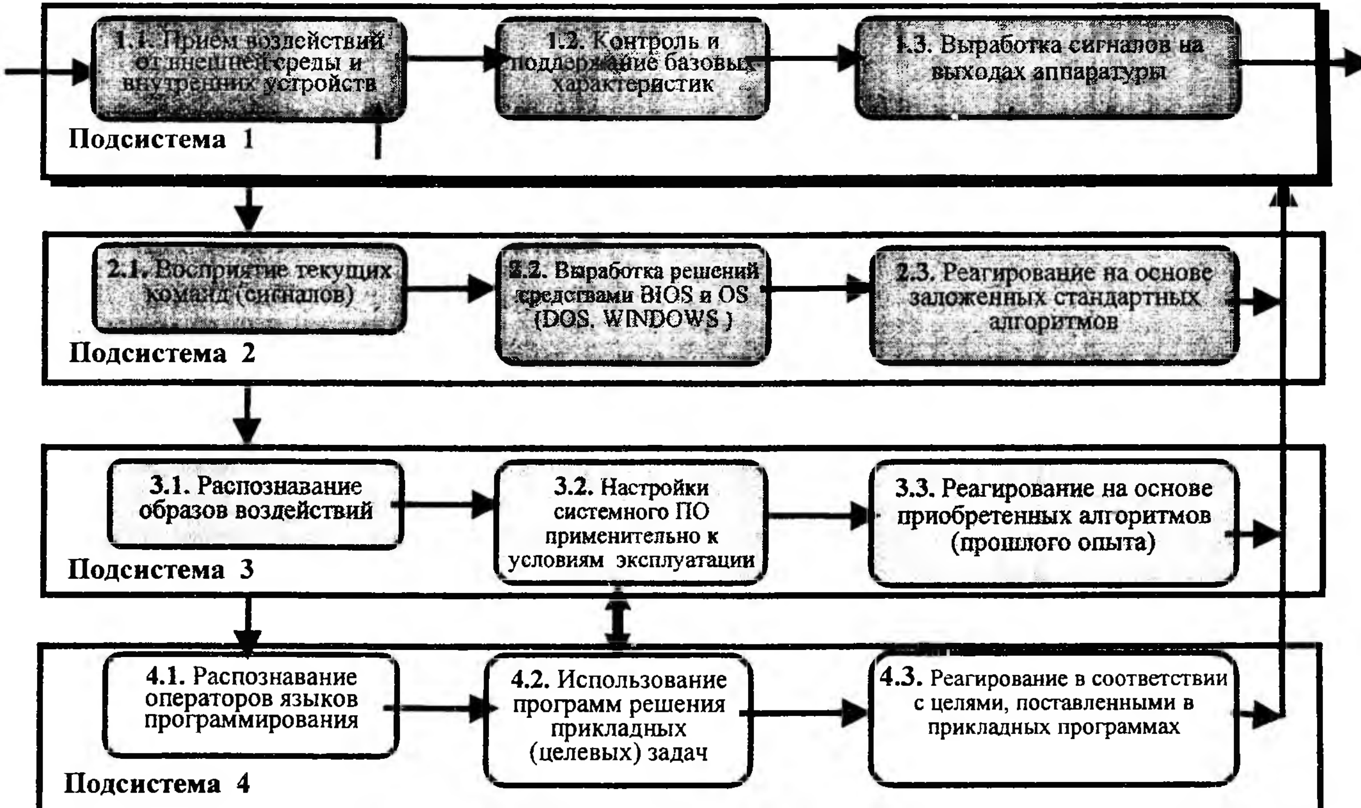
Подсистема 1. Аппаратура. Блоки:

1.1. Прием физических воздействий внешней среды и внутренних устройств.

1.2. Контроль и поддержание постоянства базовых характеристик подсистемы.

1.3. Активность в виде электрических процессов на выходах аппаратуры (срабатывание микроконтроллера клавиатуры, обеспечение требуемой частоты вращения CD-диска и т.д.).

В эту подсистему включаются, таким образом, только физиче-


1.1. Прием воздействий от внешней среды и внутренних устройств
1.2. Контроль и поддержание базовых характеристик
1.3. Выработка сигналов на выходах аппаратуры
Подсистема 1
2.1. Восприятие текущих команд (сигналов)
2.2. Выработка решений средствами BIOS и OS (DOS, WINDOWS)
2.3. Реагирование на основе заложенных стандартных алгоритмов
Подсистема 2
3.1. Распознавание образов воздействий
3.2. Настройки системного ПО применительно к условиям эксплуатации
3.3. Реагирование на основе приобретенных алгоритмов (прошлого опыта)
Подсистема 3
4.1. Распознавание операторов языков программирования
4.2. Использование программ решения прикладных (целевых) задач
4.3. Реагирование в соответствии с целями, поставленными в прикладных программах
Подсистема 4

ские процессы, происходящие в компьютерной системе, – от приема входных воздействий до возмущений на выходах аппаратуры. В ней самой управление основано на поддержании постоянства базовых характеристик, которые свойственны системе с данной архитектурой или конфигурацией.

Подсистема 2. Системное программное обеспечение. Блоки:

2.1. Восприятие текущих сигналов на основе воздействий, принятых подсистемой 1, распознавание текущих команд, заложенных в системном программном обеспечении (ПО).

2.2. Выработка и принятие решений в связи с текущей ситуацией средствами BIOS, OS или другого системного ПО.

2.3. Активность в текущей ситуации на основе заложенных, стандартных алгоритмов.

Данная подсистема обеспечивает первичную информационную (семантическую) обработку воздействий, принятых подсистемой 1. Она представляет собой, таким образом, уже информационный процесс, в ходе которого физические явления, сопровождающие воздействия на предыдущую подсистему, интерпретируются как сигналы (команды), т. е. как информация. Последняя используется в ходе анализа текущей ситуации, а также для реализации заложенных, стандартных алгоритмов.

Подсистема 3. Системно-прикладное ПО. Блоки:

3.1. Отражение информации, поступающей из подсистемы 2. Оно состоит в распознавании конкретных образов ситуации по сходству с предварительно сформированными образами. Предварительное формирование образов, имеющих сходство с отражаемыми ситуациями, осуществляется в процессе настроек системного ПО применительно к конкретным условиям эксплуатации компьютерной системы.

3.2. Выработка и принятие решений средствами прикладной части системного ПО, реализующей представление и изменение образов ситуаций с помощью инструментальных программ, макросов и текущих настроек.

3.3. Активность в виде реагирования на основе приобретенных алгоритмов (прошлого опыта).

Данная подсистема обеспечивает вторичную информационную (семантическую) обработку информации. Она представляет собой также информационный процесс, в ходе которого информация, полученная в предыдущей подсистеме, подвергается дальнейшей обработке. На основе этой информации формируются образы ситуаций, которые используются прикладной частью системного ПО, обеспечивающей учет прошлого опыта и, таким образом, реагирование на основе приобретенных алгоритмов.

Подсистема 4. Прикладное ПО. Блоки:

4.1. Отражение информации, поступающей из подсистемы 3. Оно состоит в распознавании знаков (операторов языков программирования). Предварительное формирование знаковых образов в компьютерной системе осуществляется путем обучения через установку соответствующих программ.

4.2. Выработка и принятие решений средствами прикладных программ, обеспечивающих использование логических конструкций на базе языковых средств.

4.3. Активность в виде реагирования в соответствии с целями, поставленными в прикладных программах.

Данная подсистема представляет собой, таким образом, также информационный процесс, в ходе которого информация, полученная в предыдущей подсистеме, подвергается дальнейшей обработке. На основе этой информации знаковые образы (операторы языков программирования) используются прикладными программами для реагирования в соответствии с поставленными в них целями.

Можно согласиться с тем, что "вполне уместно и даже полезно принять одну очень специфическую точку зрения и в соответствии с открываемой ею перспективой рассматривать человека как систему обработки информации. И поскольку вычислительная машина (машина Тьюринга) представляет собой универсальную систему обработки информации, то естественно сопоставлять человека, рассматриваемого с этих позиций, с вычислительной машиной" [4, с. 226]. Анализ предложенной партитивной классификации компьютерной системы показывает, что эти подсистемы, в общих чертах, аналогичны уровням сенсомоторной активности человека, описанным в работе [3], и сопоставимы с ними. Рассмотрим, в чем конкретно проявляется эта аналогия.

В данном случае, как и при рассмотрении человеческой системы переработки информации, можно утверждать, что каждая следующая по номеру подсистема, представляя собой, конечно, новое качество, не может, однако, функционировать без информации, вырабатываемой предыдущей подсистемой. Таким образом, каждая следующая подсистема является как бы эволюционной надстройкой над предыдущей. Например, прикладные программы на уровне подсистемы 4 могут функционировать только при наличии настроенного для конкретных условий эксплуатации системного ПО на уровне подсистемы 3. Соответственно, эти настройки – результат наличия на уровне подсистемы 2 стандартных алгоритмов. Работа последних, в конце концов, основана на возможностях подсистемы 1. Кроме того, каждая следующая подсистема, как и аналогичный уровень сенсомоторной активности, обеспечивает большую степень разнообразия выходных функций и, таким образом, дает возможность решения все более разнообразных и более сложных задач системой в целом.

Функции и особенности данных подсистем, как и при рассмотрении уровней сенсомоторной активности, могут быть проиллюстрированы примерами. Счеты и арифмометр представляют собой только подсистему 1, калькулятор – совокупность подсистем 1 и 2, компьютер без системного ПО (программируемый калькулятор) – совокупность подсистем 1 – 3. Наличие у компьютерной системы следующей по номеру подсистемы обусловливает необходимость повышения качества и эффективности функционирования предыдущей и, кроме того, у разных компьютерных систем качество подсистемы верхнего уровня будет различным, соответствующим уровню развития систем данного класса. История развития компьютеров может быть представлена как процесс повышения качества имеющихся подсистем, соответствующих уровням сенсомоторной активности, и добавления новых подсистем (уровней) по мере усложнения задач, решаемых компьютерами.

Процесс совершенствования функционирования подсистем компьютерной системы также может быть рассмотрен аналогично совершенствованию уровней сенсомоторной активности человека. Например, повышение качества подсистемы 2 происходит в результате появления нового поколения системного ПО для целого класса компьютерных систем. Совершенствование подсистемы 3 осуществляется путем приобретения конкретной компьютерной системой положительного (в данных условиях функционирования) опыта, т. е. в процессе обучения – инсталляций, настроек, создания макросов и т. д. В этом процессе может и не использоваться опыт, полученный ранее на другом компьютере в подобных условиях эксплуатации. Наконец, нормальное функционирование и совершенствование подсистемы 4 обеспечивается только путем предварительной настройки и подготовки конкретной компьютерной системы, в ходе которых обязательно привлекаются результаты, полученные ранее на других компьютерах (трансляторы, компиляторы и т.д.).

Некоторая аналогия с уровнями сенсомоторной активности человека наблюдается и при обнаружении особенностей одноименных подсистем у различных компьютерных систем. Компьютеры, отличающиеся друг от друга системным ПО, очевидно, имеют определенные различия в особенностях функционирования подсистемы 2; отличающиеся друг от друга условиями эксплуатации – различия в подсистеме 3; отличающиеся друг от друга функциональным предназначением – различия в подсистеме 4.

Однако между подсистемами компьютерной системы и уровнями сенсомоторной активности человека, рассмотренными в [3], существуют принципиальные различия, отраженные в таблице. Глубокое осознание этих

№ блоков	Функции блоков подсистем		Аналогия (результаты сравнения)
	человека	компьютерных систем (КС)	
1.1	Прием физических воздействий	Прием физических воздействий	Очень сильная
1.2	Поддержание постоянства характеристик внутренней среды в целях обеспечения самосохранения	Контроль и поддержание постоянства базовых характеристик	Слабая. У КС нет инстинкта самосохранения
1.3	Активность в виде физических проявлений в целях обеспечения самосохранения	Активность в виде физических проявлений	Сильная. Но у КС нет собственной активности.
2.1	Восприятие текущих сигналов. Распознавание заложенных от рождения сигналов. Формирование целостного образа текущей ситуации	Восприятие текущих сигналов. Распознавание заложенных в системном ПО команд (сигналов)	Очень сильная из-за наличия средств распознавания образов
2.2	Выработка и принятие решений в связи с текущей ситуацией за счет различения жизненно важных ее характеристик, в том числе удовольствия и боли, средствами наглядно-действенного мышления	Выработка и принятие решений в связи с текущей ситуацией средствами системного ПО	Слабая. У КС отсутствует способность различать жизненно важные сигналы удовольствия и боли
2.3	Активность в текущей ситуации на основе врожденных безусловных рефлексов (инстинктивное поведение)	Активность в текущей ситуации на основе заложенных стандартных алгоритмов	Сильная. Но у КС нет собственной активности
3.1	Отражение образов конкретных ситуаций, состоящее в их распознавании по ассоциации по сходству с предварительно сформированными обобщенными образами	Отражение образов конкретных ситуаций, состоящее в их распознавании по сходству с предварительно сформированными образами	Очень сильная из-за наличия средств распознавания образов

Окончание таблицы

№ блоков	Функции блоков подсистем		Аналогия (результаты сравнения)
	человека	компьютерных систем (КС)	
3.2	Выработка и принятие решений средствами образного мышления, осуществляемого с помощью образного представления ситуации и воображения ее изменений с учетом различных воспринятых и отраженных характеристик объектов	Выработка и принятие решений средствами прикладной части системного ПО, обеспечивающей представление и изменение образов ситуаций с помощью инструментальных программ, макросов и текущих настроек	Сильная. Но у КС отсутствует собственная оценка при представлении ситуации, а также воображение
3.3	Активность в виде мотивированного поведения на основе условных рефлексов (прошлого опыта)	Активность на основе приобретенных алгоритмов (прошлого опыта)	Сильная. Но у КС нет мотивов и активности
4.1	Отражение информации, состоящее в распознавании чувственно не воспринимаемых образов по ассоциации по смежности с предварительно сформированными знаковыми образами	Отражение информации, состоящее в распознавании знаков (операторов языков программирования) по предварительно сформированным знаковым образам	Очень сильная за счет моделирования понимания языковых знаков
4.2	Выработка и принятие решений средствами словесно-логического мышления, осуществляемого с помощью понятий и логических конструкций на базе языковых средств, а также с помощью обобщения путем абстрагирования	Выработка и принятие решений средствами прикладных программ, обеспечивающих использование логических конструкций на базе языковых средств	Сильная за счет моделирования механизмов вывода и доказательства. Но у КС нет адекватных моделей абстрагирования
4.3	Активность в виде сознательного, рационального поведения на основе целеполагания, с учетом планирования и прогнозирования развития и изменения	Активность в виде реагирования в соответствии с целями, поставленными в прикладных программах	Слабая. У КС отсутствует способность к целеполаганию данной ситуации

различий позволяет определить и сформулировать направление исследований, в результате которых возникает потенциальная возможность уменьшения различий — в целях дальнейшего совершенствования компьютерных систем на качественно новом уровне.

Приведенные результаты сравнительного анализа показывают, что в определении «разумности», или интеллекта, должны быть учтены характеристики всех уровней сенсомоторной активности человека, в том числе инстинкт самосохранения подсистемы 1, а не только рациональность и целеполагание подсистемы 4, как это в основном делается до сих пор. "Вопрос о том, что должна представлять собой «врожденная» система потребностей искусственного субъекта и чем должна определяться для него система ценностей и мотиваций" [1, с. 120] также должен решаться с учетом всех рассмотренных составляющих информационной интеллектуальной деятельности. Это значит, что, если некоторая система претендует на звание системы ИИ, она должна демонстрировать поведение на основе своего собственного целеполагания и планирования. Целеполагание невозможно без мотивов, основанных на представлении об удовольствии и боли. Это представление, в свою очередь, основано на собственном желании жить, существовать.

Подобные соображения относительно целей поведения живых организмов известны. В литературе отмечается, что "глобальные цели почти всегда разбиваются на отдельные подцели, в основе которых почти всегда лежит задача сохранения целостности организма и поддержания состояния внутренней среды в определенных допустимых пределах" [5, с. 23]. Предложенный нами подход, однако, позволяет обнаружить определенную структуру и взаимосвязь этих подцелей, что приводит к важным выводам.

Главный вывод заключается в том, что создание ИИ, действительно моделирующего естественный интеллект, должно предусматривать адекватное моделирование первого уровня сенсомоторной активности человека, т.е. учитывать принцип адекватности биосистемы [6] путем моделирования инстинкта самосохранения.

Продемонстрированная выше возможность представления человека и компьютера с помощью однотипной функциональной структуры позволяет предполагать, что данная методология может рассматриваться как универсальная при анализе сложных систем, обладающих активностью и способностью к отражению.

Список литературы: 1. *Поспелов Д.А.* Искусственная естественность и естественная искусственность // Новости искусств. интеллекта. 1995. № 4. С. 118–121. 2. *Тарасов В.Б.* От искусственного интеллекта к искусственной жизни: новые направления в науках об искусственном // Там же. С. 93–117. 3. *Маторин С.И.* Детерминантный анализ системы переработки информации человека. — См. статью в настоящем сборнике. 4. *Вейценбаум Дж.* Возможности вычислительных машин и человеческий разум. От суждений к вычислениям: Пер. с англ. М.: Радио и связь, 1982. 368 с. 5. *Гаазе-Рапопорт М.Г., Поспелов Д.А.* От амебы до робота: модели поведения. М.: Наука, 1987. 288 с. 6. *Словарь* по кибернетике / Под ред. В.С. Михалевича. К.: Гл. ред. Укр. сов. энцикл., 1989. 751 с.

Поступила в редколлегию 26.05.98

Проблемы бионики
1998. Вып. 49. С. 90--94

УДК 519.7

Н.А. ГВОЗДИНСКАЯ

О ЛОГИЧЕСКИХ ОПЕРАТОРАХ

Проблема формализации естественного языка остается актуальной. Для успешного ее решения необходим такой математический аппарат, как логические операторы. Рассмотрим логические пространства L и D над некоторым полем логических скаляров [1]. Логическим оператором, отображающим пространство L в пространство D, называется такой закон, который каждому вектору из пространства L ставит в соответствие вектор из пространства D. Обозначим логические операторы буквами $\boldsymbol{A}$, $\boldsymbol{B}$ и т. д. Тогда, например, можно записать: A: $L \to D$. Если l — вектор из логического пространства L, то вектор из пространства D, который получается из l посредством преобразования A, обозначим через Al. Вектор Al назовем образом вектора l при операторе A, а вектор l — прообразом вектора Al [2]. Совокупность всех векторов $l \in L$ таких, что $Al = d$, назовем полным прообразом этого вектора в пространстве L [3]. Логический оператор A: $L \to D$ назовем взаимно однозначным, если у каждого вектора $d \in D$ имеется один и только один прообраз $l \in L$. Операторы $\boldsymbol{A}$ и $\boldsymbol{B}$ назовем равными, если для любого вектора $l \in L$ справедливо следующее равенство:

$$\boldsymbol{A}l = \boldsymbol{B}l. \tag{1}$$

Введем понятие произведения логических операторов. Пусть имеются логические операторы $\boldsymbol{A}$ и $\boldsymbol{B}$ такие, что A: $L \to D$, $\boldsymbol{B}$: $D \to V$. Оператор $\boldsymbol{A}$ переводит произвольный вектор $l \in L$ в вектор Al. Если полученный вектор преобразовать с помощью оператора $\boldsymbol{B}$, то получим вектор $\boldsymbol{B}(Al)$. Оператор, переводящий вектор l непосредственно в вектор $\boldsymbol{B}(Al)$, будем называть произведением A и $\boldsymbol{B}$ и обозначать $\boldsymbol{BA}$. Таким образом, из определения вытекает: $(\boldsymbol{B}A)l = \boldsymbol{B}(Al)$. Если A, B: $L \to L$ и к вектору l сначала применить оператор $\boldsymbol{B}$, а затем A, то полученный вектор $A(\boldsymbol{B}l)$, вообще говоря, может и не совпадать с вектором $\boldsymbol{B}(Al)$. Другими словами, произведение логических операторов зависит от порядка сомножителей. Пусть A, $\boldsymbol{B}$, $\boldsymbol{C}$ – произвольные логические операторы и A: $L \to D$, $\boldsymbol{B}$: $D \to V$, $\boldsymbol{C}$: $V \to W$, а l – вектор из пространства L. Тогда согласно определению имеем:

$$(\boldsymbol{C}(\boldsymbol{BA}))\,l = \boldsymbol{C}((\boldsymbol{BA})\,l) = \boldsymbol{C}(\boldsymbol{B}(Al));$$

$$((\boldsymbol{CB})\,A)\,l = (\boldsymbol{CB})(Al) = \boldsymbol{C}(\boldsymbol{B}(Al)),$$

откуда следует, что

$$C\,(BA)=(CB)\,A, \tag{2}$$

т.е. выполняется свойство ассоциативности произведения операторов. Пользуясь свойством (2), легко доказать, что результат умножения любого конечного числа логических операторов, записанных в определенном порядке, от способа расстановки скобок не зависит: $(DC)(BA) = D(C\,(BA)) = D\,((CB)\,A) = (D\,(CB))\,A$. Произведение p сомножителей, равных оператору A, назовем p-й степенью оператора A и обозначим через A^p. Действия со степенями подчиняются обычным правилам в случае, когда A: $L \to L$:

$$A^pA^q = A^{p+q}; \tag{3}$$

$$(A^p)^q = A^{pq}. \tag{4}$$

Если произведение двух логических операторов A и B над логическим пространством L не зависит от порядка входящих в него сомножителей, то такие операторы будем называть перестановочными, или коммутирующими. Из выражения (3) видно, что степени одного и того же логического оператора перестановочны между собой. Если логические операторы A и B перестановочны, то

$$(BA)^2 = BABA = BBAA = B^2A^2$$

и в общем случае

$$(BA)^p = B^pA^p. \tag{5}$$

Если же A и B не являются коммутирующими, то свойство (5) может и не иметь места.

Если некоторый логический оператор каждому вектору ставит в соответствие тот же самый вектор, то такой оператор будем называть единичным, или тождественным, и обозначать через E. Следовательно, для любого вектора $l \in L$ справедливо следующее соотношение:

$$El = l.$$

Пусть A: $L \to D$ – некоторый логический оператор. Из того, что

$$(AE)l = A(El) = Al;$$

$$(EA)l = E(Al) = Al,$$

следует

$$EA = AE = A. \tag{6}$$

Если для логического оператора A: $L \to L$ существует такой логический оператор B: $L \to L$, что

$$AB = E;\ BA = E, \tag{7}$$

то $\boldsymbol{B}$ назовем обратным по отношению к A, а сам оператор A назовем обратимым. Любой обратимый логический оператор логического пространства L имеет только один обратный. Доказательство этого утверждения будем проводить от противного. Предположим, что A имеет два обратных оператора $\boldsymbol{B}$ и C. Тогда, умножая соотношение $\boldsymbol{AC} = \boldsymbol{E}$ слева на $\boldsymbol{B}$ и учитывая соотношения (2) и (7), получаем $\boldsymbol{EC} = \boldsymbol{BE}$, т.е. $\boldsymbol{C} = \boldsymbol{B}$.

Логический оператор, обратный по отношению к A, будем обозначать через A^{-1}. В выражения (7) операторы A и $\boldsymbol{B}$ входят симметрично. Следовательно, если $\boldsymbol{B}$ является обратным по отношению к A, то A является обратным по отношению к $\boldsymbol{B}$, т.е.

$$(A^{-1})^{-1} = A. \tag{8}$$

Условимся считать по определению, что

$$A^0 = \boldsymbol{E}, A^{-n} = (A^{-1})^n \quad (n = 1, 2, ...).$$

Тогда из соотношений (7) и (8) видно, что выражения (3) и (4) имеют место как для целых положительных, так и для целых отрицательных показателей степеней. Например, если оператор A обратим, то

$$(A^{-n})^{-1} = (A^{-1})^{-n} = A^n.$$

Из соотношения

$$\boldsymbol{ABB}^{-1}\boldsymbol{A}^{-1} = \boldsymbol{B}^{-1}\boldsymbol{A}^{-1}\boldsymbol{AB} = \boldsymbol{E}$$

следует, что

$$(\boldsymbol{AB})^{-1} = \boldsymbol{B}^{-1}\boldsymbol{A}^{-1}.$$

Логический оператор A: $L \to D$ назовем линейным, если он переводит произведение вектора $l \in L$ и логического скаляра α в произведение того же скаляра и соответствующего вектора $Al \in D$, а дизъюнкцию векторов из пространства L — в дизъюнкцию соответствующих им векторов из пространства D. Иными словами, логический оператор A: $L \to D$ является линейным, если для любых векторов l, $g \in L$ и любого логического скаляра α выполняются свойство однородности:

$$A(\alpha l) = \alpha(Al)$$

и свойство аддитивности:

$$A(l \vee g) = Al \vee Ag.$$

Если при реализации свойства однородности положить $\alpha = 0$, то получим $A0 = 0$. Иными словами, любой линейный логический оператор переводит нулевой вектор в нулевой. Из свойств однородности и аддитивности вытекает такое свойство линейных логических операторов: если A – линейный логический оператор, то

$$A(\alpha_1 l_1 \vee \alpha_2 l_2 \vee ... \vee \alpha_m l_m) = \alpha_1(Al_1) \vee \alpha_2(Al_2) \vee ... \vee \alpha_m(Al_m), \tag{9}$$

где $\alpha_1, \alpha_2, ..., \alpha_m$ — произвольные элементы поля логических скаляров; $l_1, l_2, ..., l_m \in L$. Для пары векторов l, g логического пространства L справедливо равенство

$$\boldsymbol{E}(\alpha l \vee \beta g) = \alpha l \vee \beta g = \alpha(\boldsymbol{E}l) \vee \beta(\boldsymbol{E}g),$$

откуда следует, что единичный оператор над логическим пространством L является линейным. Оператор, переводящий каждый вектор пространства L в нулевой, назовем нулевым и обозначим через $\boldsymbol{O}$. Если линейный оператор A взаимно однозначный, то назовем его изоморфизмом, а пространства образов D и прообразов L – изоморфными [1].

Теорема. *Если $a_1, a_2, ..., a_n$ — произвольный базис совершенного* [4] *логического пространства L, а $b_1, b_2, ..., b_n$ — произвольные векторы в пространстве D, то существует один и только один линейный логический оператор A: $L \to D$, переводящий векторы $a_1, a_2, ..., a_n$ в векторы $b_1, b_2, ..., b_n$ соответственно.*

Доказательство. Выберем в логическом пространстве L произвольный вектор l и разложим его по базису $a_1, a_2, ..., a_n$:

$$l = \xi_1 a_1 \vee \xi_2 a_2 \vee ... \vee \xi_n a_n . \tag{10}$$

Построим вектор $d \in D$:

$$d = \xi_1 b_1 \vee \xi_2 b_2 \vee ... \vee \xi_n b_n .$$

Оператор, переводящий l в d, обозначим через A. Следовательно, если для l справедливо равенство (10), то

$$Al = \xi_1 b_1 \vee \xi_2 b_2 \vee ... \vee \xi_n b_n . \tag{11}$$

Подставив сюда $\xi_i = 1$; $\xi_j = 0$ $(i \neq j,\ j = 1, ..., n)$, получим $Aa_i = b_i$ $(i = 1, ..., n)$, откуда вытекает, что оператор A переводит векторы $a_1, a_2, ..., a_n \in L$ в $b_1, b_2, ..., b_n \in D$. Докажем линейность оператора A. Умножим обе части равенства (10) на произвольный логический скаляр α. В результате получим равенство

$$\alpha l = (\alpha\xi_1)a_1 \vee (\alpha\xi_2)a_2 \vee ... \vee (\alpha\xi_n)a_n.$$

Сравнив его с выражением (11), запишем

$$A(\alpha l) = (\alpha\xi_1)b_1 \vee (\alpha\xi_2)b_2 \vee ... \vee (\alpha\xi_n)b_n.$$

Иными словами,

$$A(\alpha l) = \alpha(Al). \tag{12}$$

Возьмем вектор $g \in L$ такой, что

$$g = \eta_1 a_1 \vee \eta_2 a_2 \vee ... \vee \eta_n a_n.$$

Пользуясь выражением для дизъюнкции логических векторов, имеем

$$l\vee g = (\xi_1\vee\eta_1)a_1\vee\ldots\vee(\xi_n\vee\eta_n)a_n,$$

откуда следует, что

$$Ag = \eta_1 b_1\vee\eta_2 b_2\vee\ldots\vee\eta_n b_n;$$

$$A(l\vee g) = (\xi_1\vee\eta_1)b_1\vee\ldots\vee(\xi_n\vee\eta_n)b_n. \qquad (13)$$

Из равенства (13) с очевидностью вытекает, что

$$A(l\vee g) = \xi_1 b_1\vee\ldots\vee\xi_n b_n\vee\eta_1 b_1\vee\ldots\vee\eta_n b_n = Al\vee Ag. \qquad (14)$$

Равенства (12) и (14) доказывают линейность логического оператора $\boldsymbol{A}$. Докажем единственность этого оператора. Допустим, что существует еще по крайней мере один оператор $\boldsymbol{B}$, переводящий векторы a_1, a_2, ..., a_n логического пространства L в b_1, b_2, ..., b_n из пространства D соответственно. По условию имеем:

$$\boldsymbol{B}a_i = b_i\ (i = 1, \ldots, n).$$

Отсюда следует, что если для вектора $l\in L$ в этом пространстве имеет место разложение по базису согласно равенству (10), то

$$\boldsymbol{B}l = \xi_1(\boldsymbol{B}a_1)\vee\ldots\vee\xi_n(\boldsymbol{B}a_n) = \xi_1 b_1\vee\ldots\xi_n b_n = \boldsymbol{A}l.$$

Другими словами, $\boldsymbol{B} = \boldsymbol{A}$. Теорема доказана.

Список литературы: 1. *О логических* пространствах / Н.А. Гвоздинская, З.В. Дударь, С.А. Пославский, Ю.П. Шабанов-Кушнаренко // АСУ и приборы автоматики. 1997. Вып. 106. С. 21–30. 2. *Мальцев А.И.* Основы линейной алгебры. М.: Наука, 1975. 400 с. 3. *Шабанов-Кушнаренко Ю.П.* Неполные и полные логические пространства // Проблемы бионики. 1991. Вып. 46. С. 10–17 4. *Мальцев А.И.* Алгебраические системы. М.: Наука, 1970. 392 с.

Поступила в редколлегию 23.04.98

Проблемы бионики
1998. Вып. 49. С. 95—98

ХТУРЭ

УДК 519.85

М.В. НОВОЖИЛОВА

РЕШЕНИЕ ЗАДАЧИ РАЗМЕЩЕНИЯ НЕОРИЕНТИРОВАННОГО ОБЪЕКТА В ОБЛАСТИ С ПЕРЕМЕННЫМИ МЕТРИЧЕСКИМИ ХАРАКТЕРИСТИКАМИ

При решении задач теории исследования операций, теории искусственного интеллекта, распознавания образов возникает необходимость в визуализации и формализации аффинных преобразований над геометрическими объектами — двумерными и трехмерными. Зачастую задачи такого рода можно рассматривать как задачи оптимизационного геометрического проектирования [1].

Рассмотрим следующую задачу оптимизационного геометрического проектирования. Даны два компактных невыпуклых точечных множества T_0 и T_1, называемых далее объектами, с кусочно-линейной границей без самопересечений, состоящей из одной компоненты связности в арифметическом евклидовом пространстве R^2. Гомотопический тип компоненты границы — окружность. Другими словами, объекты T_i имеют непустые внутренности, т.е. $\operatorname{int} T_i \neq \varnothing$, $i = 0,1$. Каждый объект T_i характеризуется своей площадью s_i. Над объектами T_0 и T_1 допустимы аффинные преобразования движения. Далее над объектом T_0 допустимо также преобразование подобия. Необходимо определить такие метрические характеристики объекта T_0 при сохранении пространственной формы объектов, чтобы удовлетворялись следующие условия:

$$\min s_0 \tag{1}$$

$$T_1 \cap T_0 = T_1, \tag{2}$$

где T_1 имеет заданные метрические характеристики.

В силу того что $\operatorname{int} T_i \neq \varnothing$, $i = 0,1$ задача (1) — (2) всегда имеет решение.

Итак, объектами исследования являются невыпуклые многоугольники T_i, с параметрами размещения $u_i = (x_i, y_i, \varphi_i)$, определяющими местоположение центра O_i собственной системы координат $X_i O_i Y_i$ объекта T_i. Здесь (x_i, y_i) — параметры размещения, задающие трансляцию объекта

T_i ; φ_i — угол поворота системы $X_iO_iY_i$, $i = 0, 1$ в фиксированной системе координат XOY .

Область T_0 — компактный односвязный ориентированный многоугольник, и координаты его вершин $t_0^k = (x_0^k, y_0^k)$, $k = 1, 2, \ldots, N_0$ упорядочены. Поэтому площадь T_0 вычисляется по формуле

$$s_0 = \frac{1}{2}\sum_{i=1}^{N_0}(x_0^i - x_0^{i+1})(y_0^i + y_0^{i+1}).$$

Здесь рассматривается гомотетия G области T_0 относительно центра ее собственной системы координат (либо, когда это удобно, относительно другой точки области T_0): $G(T_0) = T_0^*$. Данное преобразование характеризуется коэффициентом гомотетии K.

Применив преобразование G только на множестве вершин объекта T_0, получим выражение для функции цели в виде

$$s_0 = K^2 s_0^*,$$

где s_0^* — площадь объекта T_0 в начальном положении.

Отметим, что в силу выпуклости и монотонности функции цели задачи (из условий задачи $K > 0$ всегда) ее глобальный минимум достигается на границе области допустимых решений D задачи.

Поскольку аффинные преобразования движения и подобия являются собственными конгруэнтными преобразованиями, задача (1) — (2) может рассматриваться в пространстве R^4 независимых переменных задачи $\{x_1, y_1, \varphi_1, K\}$.

Аналитическое описание условия (2), задающее область D задачи, основано на понятии Ф-функции [1].

Пусть $T_0^*(K) = R^2 / \operatorname{int} T_0$. Тогда введем следующее определение.

О п р е д е л е н и е . *Любая всюду определенная непрерывная нелинейная функция в* R^4, *обладающая таким характеристическим свойством:*

$$\Phi_{01}(K, u_1) > 0, \text{ если } T_1(u_1) \cap T_0^*(K) = \varnothing; \tag{3}$$

$$\Phi_{01}(K, u_1) = 0, \text{ если } T_1(u_1) \cap T_0^*(K) \neq \varnothing; \tag{4}$$

$$\operatorname{int} T_1(u_1) \cap \operatorname{int} T_0^*(K) = \varnothing;$$

$$\Phi_{01}(K, u_1) < 0, \textit{ если } \operatorname{int} T_1(u_1) \cap \operatorname{int} T_0^*(K) = \varnothing, \tag{5}$$

называется $\Phi_{01}(K, u_1)$*-функцией.*

Обозначим через G^4 0-поверхность уровня $\Phi_{01}(K,u_1)$-функции в R^4, описывающую условие (4) касания объекта $T_1(u_1)$ области $T_0^*(K)$. Поверхность G^4 состоит из множества нелинейных поверхностей $G^4{}_h$, таких, что $G^4{}_h \subset G^4 \subset R^4$, $h = 1, 2, ..., H$ $(P < H)$.

Анализ задачи показывает, что в силу ее геометрических особенностей каждая $G^4{}_h$ может быть задана в виде системы

$$\begin{cases} A_h x_1 + B_h y_1 + C(\varphi_1, K) = 0; \\ \varphi_1 \geq \varphi^-; \\ -\varphi_1 \geq -\varphi^+ \end{cases} \tag{6}$$

или системы

$$\begin{cases} A_h(\varphi_1) x_1 + B_h(\varphi_1) y_1 + C(\varphi_1, K) = 0; \\ \varphi_1 \geq \varphi^-; \\ -\varphi_1 \geq -\varphi^+. \end{cases} \tag{7}$$

Тогда математическую модель задачи можно представить в следующем виде: найти

$$\min_D K^2 \tag{8}$$

при условии, что

$$\Phi_{01}(K, u_1) \geq 0. \tag{9}$$

Предварительный анализ функций первых ограничений систем (6) — (7) на основе их матриц Гессе показывает, что в общем случае рассматриваемые функции являются знаконеопределенными. Однако ввиду того, что в данной задаче рассматриваются собственные конгруэнтные преобразования, справедлива следующая теорема.

Теорема. *Функции ограничений задачи, накладываемые на область допустимых решений, выпуклые (в частном случае линейные), а область D — вогнутое множество.*

Утверждение. *Глобальный минимум задачи достигается в крайней точке области допустимых решений задачи.*

Каждая крайняя точка области D аналитически описывается системой ранга 4 линейных и нелинейных уравнений вида (6)—(7). Обозначим множество таких систем уравнений через Q.

Оценка M числа всевозможных систем уравнений составляет

$$M = (3N_0 x N_1)^4 .$$

Отметим, что далеко не все такие системы определяют точки множества Q. Поэтому решение задачи (8)—(9) состоит: в построении совместных систем уравнений; их решении с последующей проверкой принадлежности полученных точек области допустимых решений; сравнении достигнутых на данных точках значений функции цели.

При решении задачи были существенно использованы ее геометрические особенности, среди которых значения внутренних углов при вершинах области размещения и размещаемого объекта, разбиение множества вершин области размещения и объекта на подмножества выпуклых и невыпуклых вершин, а также классификация крайних точек области допустимых решений D. Выбор первого же уравнения системы, определяющей крайнюю точку, задает предварительную ориентацию объекта и области размещения. Другими словами, накладываются ограничения на пределы изменения независимых переменных задачи (фиксируется угловой параметр либо параметры трансляции).

Проведенный анализ позволяет: существенно уменьшить оценку систем уравнений, подлежащих решению, которая в конечном итоге выражается формулой

$$M = (N_0 x N_1)^2 ;$$

свести решение системы нелинейных уравнений к решению системы линейных уравнений (с последующей проверкой принадлежности полученной точки области допустимых решений) либо к решению задачи линейного программирования специального вида [2].

Список литературы: 1. *Стоян Ю.Г., Яковлев С.В.* Математическое моделирование и оптимизационные методы геометрического проектирования. К.: Наук. думка. 1986. 286 с. 2. *Новожилова М.В., Лазарева И.Е.* Применение методологии непрерывной оптимизации в одной комбинаторной задаче геометрического проектирования с линейными ограничениями и функцией цели Х., 1997. 24 с. Деп. в УКРНИТИ 15.04.97, № 1257-1397.

Поступила в редколлегию 15.04.98

Проблемы бионики
1998. Вып. 49. С. 99—108

ХТУРЭ

УДК 007.681.518.2

М.Ф. ВОРОНОЙ

ЗАДАНИЕ ТОПОЛОГИИ НА МНОГОАСПЕКТНЫХ КЛАССИФИКАЦИОННЫХ СТРУКТУРАХ ПРИ МОДЕЛИРОВАНИИ КОНЦЕПТУАЛЬНЫХ ЗНАНИЙ

Введение

Современное состояние рынка программных продуктов свидетельствует о наличии широкого спроса на системы, использующие знание-ориентированные технологии. Это обусловлено тем, что к кругу пользователей средств обработки информации для решения повседневных задач примыкает все большее число людей, не обладающих специальными техническими знаниями. В результате формируется спрос на программные продукты, которые, с одной стороны, снимают с пользователя значительную часть рутинной работы, а с другой — обеспечивают управление процессами обработки со стороны пользователя в виде метафор, близких к пользовательскому непрофессиональному уровню навыков и знаний.

Характеризуя данные требования, можно сказать, что средства обработки информации должны обладать интеллектуальным ядром, обеспечивающим представление, приобретение и обработку сложноструктурированной информации об окружающей действительности. В наиболее общем представлении данное интеллектуальное ядро рассматривается как база знаний и способы (которые зависят от целей реализации интеллектуальной компоненты) манипулирования знаниями, имеющимися или поступающими из окружающего мира.

Рассмотрим особенности формализации представления знаний, соответствующих запросам современных информационных технологий.

Считается общепризнанным, что одной из основных задач, решаемых при построении базы знаний, является моделирование понятийного аппарата. Так, в работе [1] утверждается, что большая часть наших знаний о реальном мире относится к понятийным. Поэтому считается, что те структуры данных, которые позволяют представлять этот тип знаний, наиболее пригодны для моделирования знаний. Данная парадигма моделирования понятий (иногда применяется синоним "концепт") и взаимосвязей между ними активно используется во многих современных системах представления знаний.

Исследование и разработка теоретических основ и принципов построения понятийной модели предметной области показывают необходимость наличия в ее основе иерархической (в первую очередь, родовидовой [2]) классификации понятий.

Основной задачей, решающейся в данной работе, является описание элементов классификационных структур с использованием аппарата топологии. Понятия рассматриваются как подмножества семантического пространст ва моделируемой проблемной области, на основании чего открывается воз можность описания формализмов для оперирования семантикой.

Элементы классификационных структур

В качестве средства моделирования концептуальных знаний проблем ных областей предложены классификационные структуры, адаптированны для одновременного решения задач моделирования знаний и вывода на них Для того чтобы описание данных структур было целостным, далее приведень как общепринятые сведения о подобных средствах моделирования знаний, та и специфические, необходимые для решения задач вывода.

Предлагаемая модель знаний представлена многоаспектной классифи кационной структурой. Объектами классификации являются понятия про блемной области. С понятием, отраженным в классификации, ассоциировань некоторые его "свойства". Под последними будем понимать любые признаки которые можно вычленить у рассматриваемого объекта, описываемого поня тием. Будем считать, что объект классификации обладает определенным свой ством, если с понятием, соответствующим объекту, ассоциировано данно свойство.

На всем множестве свойств, выделенных в проблемной области у рас сматриваемых объектов, введем понятие "тип свойства". Для пояснения его смысла рассмотрим следующий пример. Пусть представлены два объекта обладающие соответствующими множествами свойств:

объект 1:{"красное", "гладкое"}, объект 2: {"синее", "шершавое"}.

Здесь выделены два типа свойств: свойство цвета (включающее свойст ва "красное" и "синее") и шероховатости (включающее "гладкое" и "шершавое").

Тип свойства — это множество, в которое объединены признаки одно природы. С введением типов свойств все их множество оказывается разбиты на классы эквивалентности:

$$\forall i,j: \quad TS_i \cap TS_j = \varnothing, i \neq j; i,j = \overline{1,n},$$

где $TS_{i,j}$ — типы свойств некоторой проблемной области, содержащей n раз личных типов.

Возникает закономерное предположение о том, что, поскольку свойств образуют класс, возможно построение целой классификации свойств про блемной области. Реализации данной идеи препятствует ряд проблем. В част ности, должны быть решены вопросы о месте классификации свойств и соот несении ее с классификацией объектов, о вычленении у свойств оснований дл их классифицирования (т.е., по сути, для определения свойств свойства). Од нако в рамках данной работы подобные вопросы не затрагиваются. Их рас смотрению, например, посвящена работа [3]. Для дальнейшего изложения достаточно отметить следующее:

– каждый объект представляется множеством типов свойств;

– в одном типе свойств у некоторого объекта может быть более чем одно свойство, например: "бело-голубое небо";

– объект описывается как множество свойств, принадлежащих одному или нескольким типам.

Одним из важнейших понятий для семантических сетей, в частности для классификаций, является понятие отношений, задаваемых между объектами. Отношения для классификаций, как правило, двуместны и направленны, т.е. для каждого отношения строго заданы выходной и входной объекты.

Выходной объект, по отношению к входному, будем называть надобъектом, или надсистемой (super-system)[1], если речь идет о системах, отраженных в понятиях проблемной области. Входной объект, по отношению к выходному, будем называть подобъектом, или подсистемой (sub-system)[3], если имеется в виду система, отраженная в понятиях проблемной области.

Учитывая особенности описания в данной работе объектов классификации, будем рассматривать отношения как оперирующие над свойствами объекта множества отображений [3].

Столь краткое описание отношений не раскрывает их сути и важности для моделирования знаний, однако по мере раскрытия содержания это будет сделано.

При данном описании элементов классификации определим некоторые ее свойства.

1. С каждой классификацией можно связать направленный граф, узлы которого соответствуют объектам, а дуги — отношениям между объектами, причем дуга направлена от выходного к входному объекту.

2. Классификационные структуры не могут содержать циклических структур, поскольку иначе появляются описания объектов через самих себя, что делает невозможным их рассмотрение применительно к детерминированным системам, использующим моделирование понятий. Тем не менее у одного подобъекта возможно наличие двух и более надобъектов.

3. Наличие совокупности объектов и отношений между ними позволяет рассматривать классификацию как разновидность семантических сетей[4]. При введении такой трактовки необходимо разъяснить смысл понятия семантики по отношению к объектам. Для семантических сетей вообще данное понятие охватывает очень широкий круг интерпретаций, однако для описываемых когнитивных структур определим это следующим образом. Семантикой объекта O будем считать: множество принадлежащих ему свойств, любые признаки, которые ставятся ему в соответствие посредством отношений с

другими объектами, и имена объектов, рассматриваемые как признаки особого рода [5].

Подводя итог, отметим, что семантика объекта включает в себя: описанные выше свойства объектов, поставленное ему в соответствие множество свойств объектов посредством отношений и имена объектов. Отметим, что множество всех имен также образует свой тип свойств и с введением подобного рассмотрения появляется возможность поддержки механизма синонимов на всей семантической сети. В случае синонимии объект обладает несколькими свойствами в типе “имя”.

Проанализируем все множество семантик проблемной области с теоретико-множественной точки зрения. Очевидно, что все указанное множество определяет некоторое пространство $\boldsymbol{S}$. Так же, как и множество свойств, множество семантик можно разбить на подмножества типов, объединяющих семантики разной природы. Это разбиение тоже образует классы эквивалентности:

$$\forall i, j : TS_i \cap TS_j = \varnothing, i \neq j; i, j = \overline{1, k},$$

где k — количество типов семантик, заданных на классификации проблемной области.

Тогда все семантическое пространство описывается так:

$$\mathbf{S} = \bigcup_{i=1}^{k} TS_i .$$

Для пространства, разбитого на классы эквивалентности по типам, уточним основные теоретико-множественные операции между объектами. Прежде всего, отметим, что на исследуемой проблемной области будем рассматривать лишь конечное и фиксированное число типов семантик.

Для каждого объекта классификации разобьем все множество его семантик на два подмножества.

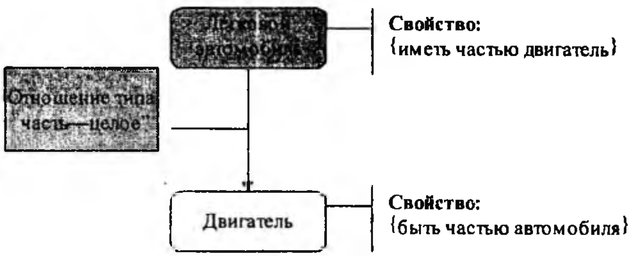
Будем называть Ω-семантикой (омега-семантикой) объекта O_i такое подмножество его семантик, которое существует только в данном объекте, а в связанных с ним объектах воспроизводится только за счет отображений посредством отношений.

Будем называть Ψ-семантикой (пси-семантикой) объекта O_i такое подмножество его семантик, которое отображается посредством отношений из других объектов.

Другими словами, Ω-семантика некоторого объекта O_i — это подмножество, с одной стороны, уникально определяющее объект в классификации (при условии, что Ψ-семантика не рассматривается), а с другой — определяющее, у какого объекта данный набор семантик встречается впервые. В свою очередь, Ψ-семантика всегда отражает тот факт, что объект O_i, обладающий ею, связан набором отношений с другими объектами, обладающими

Ω-семантикой. Таким образом, не существует Ψ-семантики самостоятельно без Ω-семантики или в отрыве от классификации.

Следует обратить внимание на то, что в обоих определениях намеренно не указывается, связан ли рассматриваемый объект с надобъектами или с подобъектами. Дело в том, что хотя отношения на классификации и направленные, но семантика, отражаемая ими, может распространяться в обе стороны. Для примера, представленного на рисунке, указаны свойства, полученные в результате рассмотрения связи между объектами. Хотя пример упрощен, он отражает возможность появления свойств как у надобъекта (за счет наличия подобъекта), так и у подобъекта (за счет связи с надобъектом).



При неправильном понимании определения Ω-семантики может сложиться мнение о том, что ее возникновение не связано с введением отношений. Однако (по крайней мере, для данного примера) семантики {иметь частью двигатель} и {быть частью автомобиля} полностью соответствуют определению Ω-семантики.

Рассмотрим основные свойства объектов, вытекающие из разбиения их множества семантик на два подмножества.

Свойства Ω- и Ψ- семантик

Очевидно, что любой объект в классификации будет определяться объединением его Ω- и Ψ-семантик. Такое объединение будем называть полной семантикой объекта:

$$O_i = \Omega_i \cup \Psi_i .$$

Следует отметить, что возможно существование объектов в классификации, для которых Ω-семантика — пустое множество. Такое состояние свидетельствует о наличии развитой и информативной классификации, для которой удалось полностью описать некоторый объект через содержание других.

Для дальнейшего рассмотрения свойств семантик зафиксируем два объекта O_i и O_j, такие, что $j \neq i$. С их помощью опишем три свойства:

I. Пересечение любых двух Ω-семантик пусто:

$$\Omega_i \bigcap \Omega_j = \varnothing .$$

Данное свойство позволяет сделать математическую запись ранее ука занной "уникальности" для каждой Ω-семантики произвольного объекта:

II. Пересечение Ω- и Ψ- семантики любого объекта всегда пусто:

$$\Omega_i \bigcap \Psi_i = \varnothing .$$

Этим определяется определяется тот факт, что для любого объекта пере сечение его Ω- и Ψ- семантики всегда пусто.

Д о к а з а т е л ь с т в о. Допустим обратное, т.е.

$$\Omega_i \bigcap \Psi_i \neq \varnothing .$$

Тогда, по определению Ψ-семантики, для объекта O_i существует неко торый объект O_j, такой что

$$\Omega_j \bigcap \Psi_i \neq \varnothing .$$

Поскольку любая Ψ-семантика состоит из Ω-семантик других объектов то, без ограничения общности, можно предположить, что

$$\Omega_i \bigcap \Omega_j \neq \varnothing .$$

Однако это невозможно ввиду определения Ω-семантики.

III. Если для двух разных объектов существует пересечение их Ω- и Ψ семантик:

$$\Omega_i \bigcap \Psi_j \neq \varnothing ,$$

то эти объекты непосредственно связаны между собой некоторым отношени ем R:

$$O_i \xrightarrow{R} O_j$$

либо они связаны транзитивно через множество других объектов:

$$O_i \xrightarrow{R_1} \alpha_1 \xrightarrow{R_2} \alpha_2 \xrightarrow{R_3} \cdots \alpha_n \xrightarrow{R_{n+1}} O_j ,$$

где n — количество объектов α_q, $q = \overline{1, n}$, связанных между собой отношения ми $\left\{ R_p \right\}_{p=2}^{n+1}$ и находящихся между объектами O_i и O_j. Далее указанную транзитивность будем обозначать так : $O_i \xrightarrow{+} O_j$.

Необходимо отметить, что Ω-семантики всех объектов классификации разбивают все семантическое пространство S классифицируемой проблемной области на классы эквивалентности*, т.е.:

$$S = \bigcup_{i=1}^{n} \Omega_i ,$$

где n — количество объектов в классификации, обладающих Ω-семантикой.

Рассмотренный перечень свойств классификаций далеко не исчерпывающий, и по мере раскрытия содержания работы он будет пополняться. Однако изложенного материала уже достаточно для формализации классификации в терминах топологии.

Основные определения топологии

Пусть X — произвольное множество и T — совокупность его подмножеств (называемых открытыми). Последние удовлетворяют следующим трем условиям[4]:

1. Пустое множество $\varnothing \in T$, $X \in T$.
2. Пересечение двух множеств из T принадлежит T.
3. Объединение любой совокупности множеств из T принадлежит T.

В этом случае пара (X, T) называется топологией, а множество T — топологической структурой.

Приведем основные определения топологических структур.

Множество X с выделенной топологической структурой T называется топологическим пространством (X,T).

Множество B открытых множеств топологического пространства называется его базой (базой открытых множеств), если каждое открытое множество этого пространства является объединением множеств из B [7]. Очень часто, вводя во множестве X топологию, указывают явно лишь некоторую ее базу. Очевидно при этом, что множество B подмножеств множества X тогда и только тогда может служить базой открытых множеств некоторой топологии на X, когда пересечение любых двух множеств из B является объединением множеств из B [6].

Топологическое пространство X называется хаусдорфовым (или отделимым), если любые две его точки p и q обладают непересекающимися окрестностями[7].

Подмножество C топологического пространства X называется замкнутым, если открыто множество $X \setminus C$. Здесь и далее '\' — операция дополнения, означающая множество элементов X, не содержащихся в C. Множества $\varnothing$ и X

* Особо следует проанализировать свойства классификации, для которой класс эквивалентности типов семантик и класс Ω-семантик совпадают.

замкнуты. Объединение любых двух замкнутых множеств замкнуто. Пересечение любой совокупности замкнутых множеств замкнуто[8].

Для любого подмножества Y топологического пространства X можно рассмотреть наименьшее замкнутое множество, содержащее Y; оно обозначается через $\bar{Y}$ и называется замыканием Y. Другими словами, $\bar{Y} = \bigcap_j F_j$, где F_j — семейство всех замкнутых множеств, содержащих Y.

Если топология $\boldsymbol{T}$ совпадает с множеством $\mathfrak{I}(X)$ всех подмножеств $\boldsymbol{X}$, то в этом случае задается топология на $\boldsymbol{X}$, называемая дискретной.

Сравнение топологий [7]

Пусть на одном и том же носителе $\boldsymbol{X}$ заданы две топологии τ_1 и τ_2. Тем самым определены два топологических пространства:

$$T_1 = (X, \tau_1)\ ;\ T_2 = (X, \tau_2).$$

Топология τ_1 сильнее топологии τ_2, если система множества τ_2 содержится в τ_1. О топологии τ_2 говорят, что она слабее, чем τ_1. В совокупность всех возможных топологий множества $\boldsymbol{X}$ естественным образом вводится частичная упорядоченность (топология τ_2 предшествует τ_1, если она слабее, чем τ_1). В этой совокупности топологий есть максимальный элемент — топология, в которой все множества открыты, и минимальный — топология, в которой открыты только все $\boldsymbol{X}$ и $\varnothing$.

Введение топологии на многоаспектных классификационных структурах

В качестве носителя топологии $\boldsymbol{X}$ рассмотрим объединение всех Ω-семантик проблемной области:

$$X = \bigcup_{i=1}^{n} \Omega_i\ ,$$

где n — количество объектов, обладающих Ω-семантикой.

Зададим топологию $\boldsymbol{T}$ следующим образом:

$$T = \left\{ \bigcup_{i \in \{0,1\}^n} A_i \right\},$$

где n — количество объектов, обладающих Ω-семантикой; i — некоторый вектор, $i = (i_1, i_2, \ldots, i_n) \in \{0, 1\}^n$; A_i — всевозможные комбинации объединений Ω- семантик на $\boldsymbol{X}$,

$$A_{i_k} = \begin{cases} \Omega_k, j_k = 1; \\ \varnothing, j_k = 0; \end{cases}$$

$$A_i = \bigcup_{k=1}^{n} A_{i_k}.$$

Для введенных обозначений пара (***X***, ***T***) является топологическим пространством

Проверим выполнение трех аксиом топологии.

1. Пустое множество $\varnothing$ и само семантическое пространство ***S*** принадлежат ***T***.

2. Пересечение двух множеств из ***T*** принадлежит ***T***, поскольку любой объект на классификации описывается через объединение Ω- и Ψ- семантик, а Ψ-семантика, в свою очередь, является также объединением Ω-семантик. Следовательно, результатом операции объединения будет множество, состоящее из пересечений Ω-семантик.

3. Объединение любой совокупности множеств из ***T*** также принадлежит ***T***, поскольку в указанной топологии перечислены все возможные пересечения Ω- семантик пространства ***S***.

Из описания ***T*** вытекают два требования к классификации:

1. На классификации объектов должно существовать отражение наличия пустого множества $\varnothing$.

Содержательная интерпретация данного требования такова. В работе [3] имеется явное указание на единственную вершину естественной классификации с нулевым (пустым) содержанием. В прагматических (не обязательно существенных) классификациях конкретных проблемных областей следует рассматривать вершину с нулевым содержанием как объект, отражающий классифицируемую проблемную область. Поскольку для построения такой классификации инженер по знаниям должен зафиксировать тот уровень обобщений, выше которого рассмотрение не имеет содержательной ценности, то таковым уровнем и будет объект — “проблемная область” с пустой семантикой, соответствующей введенному требованию о пустом множестве.

2. Свойства без объектов не существуют, так как в противном случае свойство, не принадлежащее никакой Ω-семантике, окажется вне какого-либо объекта и, следовательно, вне рассматриваемой топологии.

Содержательная интерпретация данного требования заключается в том, что оно вполне естественно и отражает тот факт, что любое свойство обязательно проявляется каким-то объектом и, следовательно, может быть вписано в классификацию проблемной области.

Перечислим некоторые свойства введенной топологии.

1. Рассматриваемая топология не является в общем случае дискретной (исключая случай, когда каждый объект представляет собой пустое множество либо множество из одного элемента).

Доказательство. Выделим непустое подмножество A Ω-семантики некоторого объекта O_i т.е. $A \subset \Omega_i$. Поскольку $A \neq \Omega_i$; $A \neq \varnothing$; $\Omega_i \bigcap \Omega_j = \varnothing$ для любого $j \neq i$ (это равносильно тому, что A не принадлежит никакому объединению Ω-семантик), то $A \notin \boldsymbol{T}$, что противоречит определению дискретной топологии.

2. Любое подмножество рассматриваемой топологии замкнуто. Для доказательства этого следует показать, что любое множество A замкнуто ($A \in \boldsymbol{T}$).

Доказательство. Множества $\varnothing$, $\boldsymbol{X}$ замкнуты [8]. Множество Ω_i замкнуто, поскольку $\boldsymbol{X} / \Omega_i = \bigcup\limits_{j \neq i} \Omega_j$ открыто (объединение открытых множеств является открытым множеством [8]). И так как объединение любых двух замкнутых множеств замкнуто, делаем вывод, что любое множество $A \in \boldsymbol{T}$ замкнуто.

3. Топологическое пространство ($\boldsymbol{X}$, $\boldsymbol{T}$) в общем случае не хаусдорфово (исключая случай, когда каждый объект представляет собой пустое множество либо множество из одного элемента).

Доказательство. Достаточно указать две точки топологического пространства, любые окрестности которых имеют непустое пересечение.

Рассмотрение основных свойств введенной топологии позволяет перейти к анализу важных следствий, являющихся результатом изучения семантического пространства концептуальных моделей проблемных областей.

Список литературы: 1. *Dilger W.* Object-oriented knowledge representation an overview // J. New generation of computer systems. 1989. N 2. P. 339–363. 2. *Чупилина Е.И.* Место термина в лексико-семантической системе языка // Вопросы терминологии и лингвистической статистики. Воронеж, 1972. С. 25–31. 3. *Соловьева Е.А., Маторин С.И.* Методы моделирования и модели понятийных знаний // НТИ. Сер. 2. 1989. № 4. С. 2–8. 4. *Искусственный* интеллект: В 3 кн.: Справ. / Под ред. Д.А. Поспелова М.: Радио и связь, 1990. Кн. 2: Модели и методы. 304 с. 5. *Поспелов Д.А.* Ситуационное управление: теория и практика. М.: Наука, 1986. 288 с. 6. *Постников М.М.* Лекции по геометрии. Гладкие многообразия. М.: Наука, 1987. 480 с. 7. *Колмогоров А.Н., Фомин С.В.* Элементы теории функций и функционального анализа. М.: Наука, 1989. 624 с. 8. *Косневски Ч.* Начальный курс алгебраической топологии: Пер. с англ. М.: Мир, 1983. 304 с.

Поступила в редколлегию 09.04.98

Проблемы бионики
1998. Вып. 49. С. 109—112

ХТУРЭ

УДК 007.681.518.2

М.Ф. ВОРОНОЙ

ФОРМАЛИЗАЦИЯ ПРОТИВОРЕЧИВЫХ ЗНАНИЙ НА КОНЦЕПТУАЛЬНЫХ КЛАССИФИКАЦИОННЫХ СТРУКТУРАХ В ТЕРМИНАХ ОТКРЫТЫХ МНОЖЕСТВ АППАРАТА ТОПОЛОГИИ

Исследование и разработка теоретических основ и принципов построения концептуальной модели произвольных предметных областей показывают возможность, необходимость и достаточность моделирования системы понятий как основы базовой модели, а также необходимость наличия в ее основе иерархической (в первую очередь, родовидовой [1]) классификации понятий. При моделировании системы понятий проблемных областей любой природы важным вопросом является необходимость контроля за непротиворечивостью приобретаемых знаний. Рассмотрим один из перспективных способов формального описания понятия противоречивых знаний на классификационных моделях с использованием аппарата топологии.

Базовые положения

Понятия рассматриваются как подмножества семантического пространства моделируемой проблемной области. Объект классификации соответствует некоторому понятию и описывается множеством присущих ему свойств. На основе общности свойств объектов строится иерархическая классификация. С построением классификации объектов проблемной области все множество свойств объекта можно разбить на два подмножества. Одно из последних содержит множество свойств, которые отличают данный объект от других, так называемое видовое различие (это подмножество будем называть Ω-семантикой). Примером Ω-семантики может быть следующее описание молотка при рассмотрении классификации всех слесарных инструментов: {средство для осуществления ударных действий}. Другим подмножеством свойств объекта являются свойства, обеспечиваемые отношениями с другими объектами. Например, при рассмотрении молотка как вида слесарного инструмента Ψ-семантика молотка формулируется так: {инструмент для проведения слесарных работ}. Полная семантика объекта O_i может быть записана в виде

$$\Omega(O_i)\cup\Psi(O_i).$$

На семантическом пространстве классификации зададим топологию. Для ее задания надо выполнить следующие шаги [2]: выделить носитель топологии; определить, что́ является открытым множеством; проверить выполнение трех аксиом топологии. В качестве носителя топологии $\boldsymbol{X}$ рассмотрим объединение всех Ω-семантик классификации, т.е.

$$\boldsymbol{X} = \bigcup_{i=1}^{n} \Omega_i ,$$

где n — количество объектов, обладающих Ω- семантикой.

Следует отметить, что при таком задании X Ω-семантика образует разбиение пространства на классы эквивалентности.

Зададим топологию $\boldsymbol{T}$ следующим образом:

$$\boldsymbol{T} = \left\{ \bigcup_{i \in \{0,1\}^n} A_i \right\},$$

где n — количество объектов, обладающих Ω-семантикой; i — некоторый вектор, $i = (i_1, i_2, \ldots, i_n) \in \{0,1\}^n$; A_i — всевозможные комбинации объединений Ω- семантик на $\boldsymbol{X}$,

$$A_{i_k} = \begin{cases} \Omega_k, i_k = 1; \\ \varnothing, i_k = 0; \end{cases}$$

$$A_i = \bigcup_{k=1}^{n} A_{i_k} .$$

Для введенных обозначений пара ($\boldsymbol{X}$, $\boldsymbol{T}$) является топологическим пространством.

Проверим выполнение трех аксиом топологии. Первая из них выполняется: пустое множество ∅ и само семантическое пространство $\boldsymbol{S}$ принадлежат $\boldsymbol{T}$. Относительно второй отметим, что пересечение двух множеств из $\boldsymbol{T}$ принадлежит $\boldsymbol{T}$, поскольку любой объект на классификации описывается через объединение Ω- и Ψ- семантик, а Ψ-семантика, в свою очередь, является также объединением Ω-семантик. Следовательно, результатом операции объединения будет множество, состоящее из пересечений Ω-семантик. Рассмотрев выполнение третьей аксиомы, получим, что объединение любой совокупности множеств из $\boldsymbol{T}$ также принадлежит $\boldsymbol{T}$, поскольку в данной топологии перечислены все возможные пересечения Ω-семантик пространства $\boldsymbol{S}$.

На основании понятия открытого множества опишем понятие противоречивой информации. Очевидно, что можно указать очень широкий круг разного рода противоречий, недопустимых для описания проблемных областей, однако здесь и далее в этом контексте подразумевается только наличие у объекта взаимоисключающих свойств.

Противоречивой информацией будем называть такой набор семантик, который не является открытым в $\boldsymbol{T}$ множеством. Поскольку все Ω-семантики образуют разбиение пространства X, то очевидно, что любое свойство, выделенное на предметной области, находится в некотором $\Omega_i \in \boldsymbol{T}$. Допустим, что заданы два объекта: 1) $\Omega_i = \{a_1, a_2, \ldots, a_n\}$, где a_j — некоторое свойство принадлежащее объекту с семантикой Ω_i; 2) $\Omega_k = \{b_1, b_2, \ldots, b_q\}$, где b_p — некоторое свойство, принадлежащее объекту с семантикой Ω_k. Тогда при рассмотрении множества свойств вида $\{a_j, b_p\}$ оказывается, что подобное множество не открыто в $\boldsymbol{T}$, так как оно не является ни пустым множеством (∅), ни всем пространством (X), ни объединением никаких Ω-семантик.

Дадим содержательную интерпретацию изложенного. Прежде всего определим, что́ имеется в виду, когда предлагается рассмотреть некоторое множество семантик. Это значит, что в терминах и рамках некоторой зафиксированной (на каком-то шаге итерационного построения) или на законченной классификации делается попытка ответить на ряд вопросов: существует ли объект, отвечающий данным свойствам на классификации, допустима ли данная комбинация семантик для существующей классификации, необходимо ли произвести расширение классификации за счет рассматриваемых семантик — и так далее, в зависимости от задач, решаемых с использованием классификаций.

Поэтому очень важно сформулировать, по крайней мере, необходимые условия для ограничения введения противоречивых свойств.

Поскольку Ω-семантика выражает концепцию атомарного, или в семантическом смысле неразделяемого, множества свойств некоторого объекта, выделенного в проблемной области, то утверждение о том, что некоторое его подмножество может быть выделено у других объектов, противоречит данной концепции. Возникает вопрос о выработке методологии построения классификационных моделей, не содержащих противоречивой информации данного рода.

Общий методологический подход к разрешению ситуаций с противоречивой информацией

Методология разрешения противоречивой информации связана с формированием новой Ω-семантики (т.е., по сути, с созданием нового объекта) для классификаций проблемной области. В предыдущем подразделе были приняты обозначения для двух объектов с противоречивыми семантиками. В соответствии с ними противоречивая семантика записывается как множество вида $\{a_j, b_p\}$. Сформируем два объекта O_1 и O_2 с семантиками следующего вида: $\Omega_1 = \{a_j\}$; $\Omega_2 = \{b_p\}$. Тогда с помощью отношений обеспечим связь с конфликтующими объектами, в результате чего семантика конфликтующих объектов перейдет из множества Ω во множество Ψ и общая семантика объекта останется неизменной.

Фактически произошло изменение введенной топологии. Легко показать [3], что новая топология сильнее изначальной.

Таким образом, получено описание понятий как подмножеств семантического пространства проблемной области.

С использованием базового понятия топологии — открытых множеств рассмотрен один из аспектов представления противоречивых знаний в классификационной модели. Применение понятий сильной и слабой топологии к анализу противоречивых знаний позволило предложить способ разрешения некоторых противоречий с последующим расширением классификации.

Список литературы: 1. *Чупилина Е.И.* Место термина в лексико-семантической системе языка // Вопросы терминологии и лингвистической статистики. Воронеж, 1972. С. 25–31. 2. *Косневски Ч.* Начальный курс алгебраической топологии: Пер. с англ. М.: Мир, 1983. 304 с. 3. *Колмогоров А.Н., Фомин С.В.* Элементы теории функций и функционального анализа. М.: Наука, 1989. 624 с.

Поступила в редколлегию 09.04.98

Проблемы бионики
1998. Вып. 49. С. 113—117

ХТУРЭ

УДК 519.23/25

С.Н. ГЕРАСИН

ОБЛАСТИ ЛОКАЛИЗАЦИИ ХАРАКТЕРИСТИЧЕСКИХ КОРНЕЙ КВАЗИСТОХАСТИЧЕСКИХ МАТРИЦ

При изучении различных биологических, экологических и экономических процессов, обусловленных быстроизменяющимися факторами, часто используются процессы Маркова. В этом случае процесс удобно задавать либо матрицей переходных вероятностей (т.е. стохастической), либо системой уравнений Колмогорова, причем главная матрица системы является квазистохастической [1]. Для таких процессов очень важно определять существование стационарного режима и время перехода к нему. Для этого необходимо устанавливать расположение собственных чисел указанных матриц на комплексной плоскости.

Рассмотрим множество всех стохастических матриц n-го порядка и обозначим через M_n множество характеристических корней всех таких матриц. В работах [2; 3] множество M_n определено полностью. В работе [2] показано, что для $n=2$ M_n представляет собой отрезок действительной оси $[-1,1]$, а для $n=3$ M_n представляет собой объединение треугольника с вершинами в точках $(1,0)$, $\exp(2\pi i/3)$, $\exp(4\pi i/3)$ и отрезка действительной оси $[-1,1]$. В работе [3] показано, что для $n>3$ фигура M_n симметрична относительно действительной оси, заключена в круге $|z|\le 1$ и имеет с окружностью $|z|=1$ общие точки $\exp(2\pi i a/b)$, где $0\le a<b\le n$. Граница M_n состоит из этих точек и соединяющих их в круговом порядке криволинейных дуг. Каждая из этих дуг задается одним из следующих параметрических уравнений:

$$\lambda^{q}(\lambda^{p}-t)^{r}=(1-t)^{r}; \tag{1}$$

$$(\lambda^{b}-t)^{d}=(1-t)^{d}\lambda^{q}, \tag{2}$$

где параметр t изменяется в пределах $0\le t\le 1$, а b,d,p,q,r – натуральные числа, которые определяются следующим образом: пусть концы некоторой дуги, взятые по направлению против часовой стрелки, есть $\exp(2\pi i a'/b')$ и $\exp(2\pi i a''/b'')$. Возможны два случая:

$$b''[n/b'']\ge b'[n/b']; \tag{3}$$

$$b''[n/b'']\le b'[n/b']. \tag{4}$$

Если для некоторой дуги имеет место случай (3), то для комплексно-сопряженной дуги имеет место случай (4), и наоборот. Поэтому в силу симметрии M_n достаточно определить дуги, удовлетворяющие условию (3). Пусть $r_1 = b''$, $r_2 = a''$, $r_3, \dots, r_m$ – последовательность остатков, получающихся при нахождении наибольшего общего делителя чисел b'' и a'' посредством алгоритма Евклида. Если $[n/b''] = 1$ и для некоторого целого S: $r_{2s} = 1$, то дуга, соединяющая точки $\exp(2\pi i a'/b')$ и $\exp(2\pi i a''/b'')$, задается уравнением (1), где $r = r_{2s-1}$, а числа p, q определяются из соотношений

$$a'' p \equiv 1 \bmod b'' (0 < p < b'');$$
$$a'' q \equiv -r \bmod b'' (0 \le q < b'').$$

В противном случае дуга, соединяющая точки $\exp(2\pi i a'/b')$ и $\exp(2\pi i a''/b'')$, задается уравнением (2), причем $d = [n/b''], b = b''$, а q определяется из соотношения

$$a'' q \equiv -1 \bmod b'' (0 < q < b'').$$

Исследуем самый первый участок границы M_n. Это криволинейная дуга, соединяющая точки (1,0) и $\exp(2\pi/n)$, т.е. в данном случае

$$a' = 0, b' = n, a'' = 1, b'' = n.$$

Тогда $b''[n/b''] = n[n/n] = n$, а $b'[n/b'] = n[n/n] = n$, а значит, данный случай относится к (3). Далее, $[n/b''] = [n/n] = 1$ и для целого $s = 1$ $r_2 = 1$ в последовательности остатков алгоритма Евклида $r_1 = n$, $r_2 = n, \dots$.Тогда эта дуга задается уравнением (1), причем $p = n$, а $q = 0$.

Таким образом, для всех $n > 3$ участок границы фигуры M_n, соединяющий точки (1,0) и $\exp(2\pi / \mathrm{n})$, удовлетворяет параметрическому уравнению

$$(\lambda - t)^n = (1 - t)^n, \quad 0 \le t \le 1.$$

В этом уравнении точке (0,1) соответствует значение параметра $t = 1$, а точке $\exp(2\pi / \mathrm{n})$ -- значение параметра $t = 0$:

$$\lambda(0) = \exp(2\pi / n); \ \lambda(1) = 1.$$

Для удобства преобразуем параметрическое уравнение для $\lambda(t)$ в параметрическое уравнение для $\lambda(u)$, где $u = 1 - t$:

$$(\lambda(u) - 1 + u)^n = u^n.$$

Здесь $0 \le u \le 1$; $\lambda(0) = 1$; $\lambda(1) = \exp(2\pi / n)$.

Рассмотрим параметрическое уравнение для вспомогательной дуги $z(u) = \lambda(u) - 1 + u$:

$$z(u)^n = u^n, \text{ где } 0 \le u \le 1 \quad (z(0) = 0, z(1) = \exp(2\pi / n)).$$

Тогда $\arg z(u) = 2\pi / n, |z(u)| = |u|$ для $0 \le u \le 1$. Итак, дуга $z(u)$ представляет собой прямолинейный отрезок, соединяющий точки комплексной плоскости 0 и $\exp(2\pi i / n)$. Координаты точек данного отрезка удовлетворяют параметрическим уравнениям

$$x(u) = \cos(2\pi / n)u; \quad y(u) = \sin(2\pi / n)u, \text{ где } 0 \le u \le 1.$$

Тогда координаты точек отрезка дуги $\lambda(u)$ удовлетворяют параметрическим уравнениям

$$x(u) = (\cos(2\pi / n) - 1)u + 1; \quad y(u) = (\sin(2\pi / n))u, \text{ где } 0 \le u \le 1.$$

Таким образом, отрезок дуги $\lambda(u)$ представляет собой отрезок прямой, соединяющий точки комплексной плоскости 1 и $\exp(2\pi i / n)$.

Рассмотрим произвольную ненулевую квазистохастическую матрицу $A \in R^{n \times n}$, т.е. матрицу, элементы которой $a_{ij} (1 \le i, j \le n)$ удовлетворяют следующим условиям:

$$a_{ii} \le 0, \quad 1 \le i \le n; \tag{5}$$

$$a_{ij} \ge 0, \quad 1 \le i, \quad j \le n, \quad i \ne j; \tag{6}$$

$$\sum_{j=1}^{n} a_{ij} = 0, \quad 1 \le i \le n. \tag{7}$$

Пусть ρ – максимальный по модулю диагональный элемент матрицы A. Из условия (5) следует, что ρ – максимальный по модулю отрицательный элемент матрицы A, а из условий (6) и (7) вытекает, что ρ – максимальный по модулю элемент матрицы A.

Построим матрицу следующим образом:

$$B = (A + |\rho| E)/|\rho|,$$

где E – единичная матрица.

Элементы матрицы $b_{ij}\,(1 \le i, j \le n)$ удовлетворяют следующим условиям:

$$0 \le b_{ij} \le 1\,,\ 1 \le i, j \le n\,; \tag{8}$$

$$\sum_{j=1}^{n} b_{ij} = 1\,,\ 1 \le i \le n\,. \tag{9}$$

Значит, матрица B является стохастической. Найдем границы множества собственных значений квазистохастической матрицы A. Пусть некоторое число λ_0 является характеристическим корнем матрицы A. Тогда оно удовлетворяет характеристическому уравнению матрицы A:

$$|A - \lambda_0 E| = 0\,;$$

$$||\rho|(B - E) - \lambda_0 E| = 0\,;$$

$$||\rho| B - (\lambda_0 + |\rho) E/|\rho|| = 0\,.$$

Итак, число $(\lambda_0 + |\rho|)/|\rho|$ удовлетворяет характеристическому уравнению матрицы B и, следовательно, является характеристическим корнем стохастической матрицы B.

Обозначим множество характеристических корней всех квазистохастических матриц n-го порядка с максимальным по модулю элементом ρ через K_n^{ρ}. Из сказанного выше следует, что все элементы множества K_n получаются из элементов множества M_n умножением на число $|\rho|$ и вычитанием числа $|\rho|$. Таким образом, для квазистохастических матриц n-го порядка с максимальным по модулю элементом ρ справедливы следующие утверждения.

Элемент K_2^{ρ} представляет собой отрезок действительной оси $[-2|\rho|, 0]$, а K_3^{ρ} – объединение треугольника с вершинами в точках (0, 0), $|\rho|\exp(2\pi i/3) - |\rho|$, $|\rho|\exp(4\pi i/3) - |\rho|$ с отрезком действительной оси $[-2|\rho|, 0]$.

Для $n > 3$ фигура K_n^{ρ} заключена в круге $|z + |\rho|| \le |\rho|$ и имеет с окружностью $|z + |\rho|| = |\rho|$ общие точки $|\rho|\exp(2\pi i a/b) - |\rho|$, где $0 \le a < b \le n$. Граница K_n^{ρ} состоит из этих точек и соединяющих их в круговом порядке криволинейных дуг. Отрезки границы множества K_n^ρ, проходящие через точку комплексной плоскости (0, 0), представляют собой отрезки прямых, соединяющих точки $|\rho|\exp(2\pi i(n-1)/n)-|\rho|$ и (0, 0), (0, 0) и $|\rho|\exp(2\pi i/n)-|\rho|$ соответственно.

Таким образом, найдены области локализации характеристических корней квазистохастических матриц. Это позволяет исследовать вопрос о стационарном режиме различных биологических, экологических и экономических процессов, обладающих свойством отсутствия последействия.

Список литературы: 1. *Гантмахер Ф.Р.* Теория матриц. М.: Наука, 1988. 548 с. 2. *Карпелевич Ф.Р.* О корнях матриц с неотрицательными элементами // Изв. АН СССР. Сер. мат. 1951. Т. 15 С. 361–383. 3. *Seneta E., Vere-Jones D.* On quasi-stationary distributions in discrete-time Markov chains with a denumerable infinity of states // J. Appl. Probability. 1966. N 3. P. 403–434.

Поступила в редколлегию 13.02.98

Проблемы бионики
1998. Вып. 49. С. 118—123

УДК 519.85

К.Е. КОРОТИН

ОСОБЕННОСТИ РЕШЕНИЯ СИСТЕМ ЛИНЕЙНЫХ НЕРАВЕНСТВ ПРИ ИСПОЛЬЗОВАНИИ МОДИФИЦИРОВАННОГО МЕТОДА РАЗМЕЩЕНИЯ МНОГОУГОЛЬНИКОВ В ПОЛОСЕ

В течение ряда лет коллективом отдела математического моделирования и оптимального проектирования Института проблем машиностроения НАН Украины разрабатывается система, предназначенная для автоматического моделирования, выбора метода и алгоритма решения задач геометрического проектирования. Одной из основных задач нерегулярного двумерного геометрического проектирования является следующая задача. Пусть имеются полубесконечная полоса S_0 и n ориентированных ограниченных φ-многоугольников [1] S_1, S_2, ... , S_n. Необходимо найти такой вариант размещения многоугольников в полосе, чтобы они между собой не пересекались и длина занятой части полосы z была минимальной.

Впервые данная задача рассмотрена в работе [2]. Было показано, что задачу можно свести к минимизации линейной функции цели z на области D, которая представляет собой невыпуклый и в общем случае несвязный многогранник в пространстве R^k, где $k = 2n + 1$. В работе [2] подробно описан способ представления области D в виде объединения выпуклых многогранников. Выпуклый многогранник, как известно, является пересечением конечного числа замкнутых полупространств, т. е. определяется системой линейных неравенств. На каждом из этих выпуклых многогранников минимум функции цели может быть найден с помощью решения задачи линейного программирования с ограничениями в виде линейных неравенств. Для организации перебора строится так называемое дерево решений. Оно гарантирует, что построенное на последнем уровне множество систем линейных неравенств мажорирует множество систем, реально описывающих выпуклые подобласти области D .

На практике число построенных на последнем уровне систем неравенств оказывается настолько большим, что необходимо проводить дополнительный анализ, позволяющий сократить число систем, подлежащих перебору.

Перейдем к формальной постановке задачи. Свяжем с каждым из рассматриваемых объектов собственную систему координат. Ее начало назовем полюсом объекта. Положение объекта на плоскости определяется его параметрами размещения. Под параметрами размещения объекта $x^i = (x_1^i, x_2^i)$ будем понимать координаты полюса этого объекта в неподвижной системе координат, связанной с левым нижним углом полосы.

Область допустимых решений D может быть представлена в следующем виде:

$$D = \left(\bigcap_{i=1}^{n} \Phi^{0i} \right) \cap \left(\bigcap_{g=1}^{\omega} \Phi^{g} \right), \tag{1}$$

где Φ^{0i} — область, определяемая условиями размещения i-го многоугольника в полосе; Φ^g — область, определяемая условиями взаимного непересечения i-го и j-го многоугольников. Здесь индекс g задает нумерацию этих областей для всех пар объектов, т. е. $g = g(i,j)$, $g = 1, \ldots, \omega$, где $\omega = C_n^2 = n(n-1)/2$.

Очевидно, что области Φ^{0i} являются выпуклыми, а области Φ^g — невыпуклыми. Представим область Φ^g в виде объединения выпуклых подобластей:

$$\Phi^g = \bigcup_{q=1}^{p_g} \Phi_q^g . \tag{2}$$

Введем переменные γ_g, определяющие номер выпуклой подобласти, выделенной из области Φ^g, $g = 1, \ldots, \omega$. Ясно, что $\gamma_g \in \{1, \ldots, p_g\}$. Представим область D в виде объединения выпуклых подобластей:

$$D = \bigcup_{r=1}^{p} D_r , \tag{3}$$

где

$$D_r = \left(\bigcap_{i=1}^{n} \Phi^{0i} \right) \cap \left(\bigcap_{g=1}^{\omega} \Phi_{\gamma_g}^{g} \right), \tag{4}$$

а номер r зависит от конкретных значений переменных γ_g, т. е. $r = r(\gamma_1, \ldots, \gamma_\omega)$. Другими словами, для того чтобы получить одну из областей D_r, необходимо выбрать по одному выпуклому подмножеству из каждой области Φ^g, $g = 1, \ldots, \omega$ и найти их пересечение со всеми множествами Φ^{0i}, $i = 1, \ldots, n$. Общее число областей D_r

$$p = \prod_{g=1}^{\omega} p_g . \tag{5}$$

Исходная задача может быть записана в следующем виде:

$$z^* = \min_{x \in D \subset R^k} f(x) = \min_{r \in I_p} (\min_{x \in D_r} f(x)) ;$$

$$x = (x_1^1, x_2^1, \ldots, x_1^n, x_2^n, z) ; \tag{6}$$
$$f(x) = z .$$

Длина полосы z в качестве независимой переменной входит в структуру ограничений, откуда вытекает, что функция цели f линейна. Минимум функции цели на каждой из выпуклых областей D_r находится с помощью решения задачи линейного программирования. Чаще всего для этого используется симплекс-метод [3].

Как показал проведенный анализ, выделенные таким образом выпуклые подобласти в большинстве являются пустыми либо подобластями других выпуклых областей области D. Формирование и перебор указанных подобластей крайне нежелательны. На практике для сокращения перебора применяется ряд правил отсечения.

Сформулированное ниже утверждение позволяет сократить перебор не только пустых подобластей, но и тех областей, минимум функции цели на которых принимает одинаковые значения.

Утверждение. Пусть x^* — точка минимума функции цели $f(x)$, $x \in R^k$ при ограничениях, заданных системой линейных неравенств $Ax \geq b$, и пусть $A'x \geq b'$ — некоторая совместная система неравенств, отличающаяся от системы $Ax \geq b$ лишь теми неравенствами, которые не являются активными в точке x^* или множители Лагранжа которых равны нулю. Тогда

$$\min_{A'x \geq b'} f(x) = \min_{Ax \geq b} f(x) .$$

Известно, что решение задачи

$$z^* = \min_{Ax \geq b} f(x) \tag{7}$$

достигается в некоторой вершине x^* допустимой области, определяемой системой неравенств $Ax \geq b$. Точнее говоря, среди всех оптимальных точек найдется хотя бы одна вершина. Обозначим через $\hat{A}$ $k \times k$ -матрицу, i -я строка которой состоит из коэффициентов i -го ограничения, активного в x^*. Для того чтобы точка x^* была решением задачи (7), необходимо и достаточно, чтобы для некоторого вектора $\lambda \geq 0$, $\lambda \in R^k$ выполнялось равенство

$$\nabla f(x) = \hat{A}^T \lambda . \tag{8}$$

Вектор λ называется вектором множителей Лагранжа. Требование $\lambda \geq 0$ связано с тем, что появление отрицательного множителя означало бы существование допустимого направления спуска, что противоречило бы оптимальности x^*.

Процедура поиска решения представляет собой обычный итеративный процесс. На каждом шаге определяется направление поиска p и вычисляется шаг α. Пусть x — очередное приближение, причем матрица $\hat{A}$ ограничений, активных в точке x, не вырождена. Обозначим через I множество индексов активных ограничений — так называемый рабочий список. Вектор множителей Лагранжа представляет собой решение невырожденной системы линейных уравнений (8). Если все компоненты $\lambda \geq 0$, то x является решением задачи. Если же среди них найдется отрицательный (скажем, $\lambda_s < 0$), то целевую функцию можно уменьшить, двигаясь по направлению, вдоль которого левая часть s -го ограничения возрастает, а остальные ограничения из рабочего списка остаются равенствами.

Данное утверждение непосредственно следует из условия экстремальности (8), так как последнее зависит лишь от тех ограничений, которые являются активными в точке x^* и множители Лагранжа которых строго положительны.

Дадим геометрическую трактовку данного утверждения для случая с нулевыми множителями Лагранжа, поскольку случай с заменой неактивных ограничений достаточно очевиден.

Как известно, множитель Лагранжа, соответствующий какому-либо активному ограничению, представляет собой скорость изменения целевой функции при движении вдоль направления, не удерживающего относительно этого ограничения и удерживающего относительно всех остальных активных ограничений. Поэтому равенство множителя нулю свидетельствует о том, что функция остается постоянной при указанном перемещении. Предположим теперь, что в рабочем списке системы $Ax \geq b$, соответствующем вершине x^*, есть m ограничений с нулевыми множителями Лагранжа. Это означает, что существует линейное многообразие размерности $k-m$, которое определяется активными ограничениями с положительными множителями Лагранжа и на котором функция цели остается постоянной и равной своему экстремальному значению. Указанное многообразие имеет смысл допустимого множества всех возможных перемещений, на котором остаются активными все ограничения с положительными множителями Лагранжа. Таким образом, исключив из активного набора ограничения с нулевыми множителями, нельзя улучшить функцию цели.

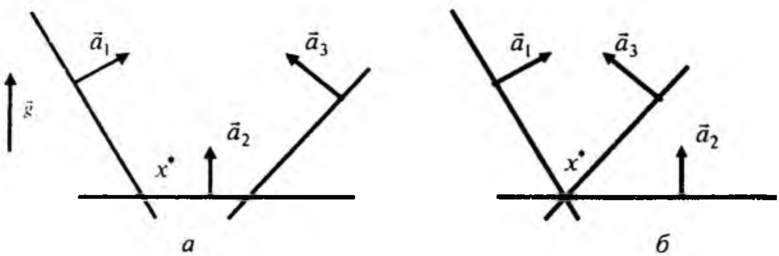


Рис. 1

На рис. 1, *а* данное утверждение проиллюстрировано для двухмерного случая. Активными в точке x^* являются ограничения a_1 и a_2 (обозначены $\vec{a}_i$ — вектор нормали i-го ограничения и $\vec{g}$ — направление градиента). Множитель Лагранжа, соответствующий ограничению a_1, равен нулю, и движение вдоль направления, не удерживающего относительно a_1 и удерживающего относительно a_2, не может привести к уменьшению функции цели, поскольку направление такого движения перпендикулярно к градиенту.

На рис. 1, *б* показан вырожденный случай, когда в точке x^* активными являются сразу три ограничения. Здесь возможны три варианта формирования рабочего списка: (a_1, a_2), (a_2, a_3), (a_1, a_3). При первых двух вариантах множители Лагранжа ограничений a_1 и a_3 равны нулю. Однако вследствие вырождения нельзя сделать положительный шаг в допустимой области вдоль удерживающего ограничения a_2. При третьем варианте множители Лагранжа обоих ограничений положительны. Но, опять-таки вследствие вырождения, разорвав, например, a_1, нельзя сделать положительный шаг вдоль удерживающего ограничения a_3 в направлении убывания функции цели, так как сразу же станет активным ограничение a_2. Следовательно, и в этом случае замена ограничений не приведет к улучшению функции цели.

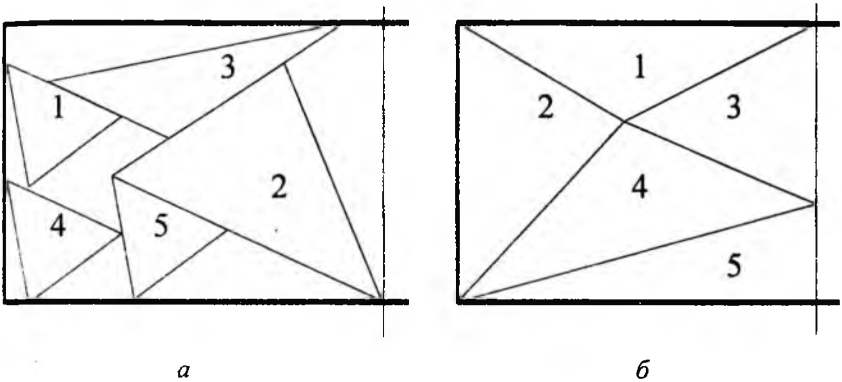


Рис. 2

Изложенный метод решения задачи размещения многоугольников в полосе реализован на языке Паскаль. На рис. 2 показан результат численного эксперимента. Время работы программы на AT PC 486 составило 29 и 16 с для примеров на рис. 2, *а* и 2, *б* соответственно.

Список литературы: 1. *Стоян Ю.Г., Яковлев С.В.* Математическое моделирование и оптимизационные методы геометрического проектирования. К.: Наук. думка, 1986. 286 с. 2. *Магас С.Л.* Методы решения экстремальных задач размещения многоугольных геометрических объектов в полосе: Автореф. дис. ... канд. физ.- мат. наук. М., 1984. 20 с. 3. *Гилл Ф., Мюррей У., Райт М.* Практическая оптимизация: Пер. с англ. М.: Наука, 1985. 325 с.

Поступила в редколлегию 15.04.98

Проблемы бионики
1998. Вып. 49. С. 124—129

ХТУРЭ

УДК 801.1+801.7

А.В. РАССАДНИКОВА

ПРИНЦИПЫ ПОСТРОЕНИЯ ДВУЯЗЫЧНОГО УНИВЕРСАЛЬНОГО ЭЛЕКТРОННОГО СЛОВАРЯ

Создание электронных словарей рассматривается как одна из основных проблем компьютерной лингвистики. С другой стороны, одной из технологически удобных и коммерчески перспективных областей внедрения компьютерной техники неизменно остается лингвистика, особенно двуязычная лексикография. На Западе во впечатляющей прогрессии растет количество электронных словарных массивов; весьма многочисленны лексические базы данных на компакт-дисках, заметная часть которых успешно реализуется на рынке. Ожидается, что в развитых странах к началу следующего тысячелетия около 1/4 книг будет электронной. По мнению авторитетных издателей США, справочники, словари, энциклопедии в самом скором времени будут храниться только на безбумажных носителях.

Большим преимуществом электронной лексикографии по сравнению с традиционной является оптимальное сочетание огромных объемов лексической информации с высокими актуализационными и издательскими возможностями. Трудности заключаются в основном в необходимости усовершенствования системы обработки источниковой лексической информации для оперативного отслеживания неологизмов и актуализации машинных лексикографических массивов.

Поскольку отечественная лингвистическая практика столкнулась с проблемой формализации лексической системы украинского языка для ее сопряжения с развитыми западными аналогами и обеспечения эффективного межъязыкового перекодирования текстовой информации, необходимость внедрения языковой лексикографии в Украине ощущается особенно остро. Создание новых двуязычных электронных словарей с участием украиноязычного звена позволило бы организовать оперативный выпуск столь необходимых полноценных двуязычных словарей для широкой рыночной реализации и существенно облегчить межнациональные языковые контакты. В отечественной лексикографии уже идет поиск концепции решения указанной проблемы. В этом отношении весьма эффективным представляется использование опыта разработки и практического внедрения двуязычных электронных словарей, накопленного в развитых западных странах, а также в России.

В частности, в российской лексикографической практике выделяется две интересные разработки: создание машинного фонда русского языка и создание системы электронных словарей для полумашинного перевода.

Принципы построения машинного фонда русского языка, формируемого под руководством специалистов Института русского языка Российской Академии наук, могут быть использованы в качестве прототипной базы для машинного упорядочения лексической системы украинского языка в целях ее последующего эффективного вывода на билингвистический уровень. Публикации разработчиков машинного фонда русского языка уже довольно многочисленны [1].

Концептуальный анализ лингвистических и машинных аспектов российской системы автоматических словарей для полумашинного перевода научно-технических текстов [2] также может оказаться полезным для отечественной электронной лексикографии, хотя следует подчеркнуть что уровень западных разработок в этой области намного выше. Заметим, что в новейшей российской лексикографии преобладают успехи в создании крупных двуязычных словарей на бумажных носителях, в то время как западная лексикографическая практика развивается преимущественно в направлении безбумажной информатики.

Динамика объемов крупнейших традиционных словарей с англоязычным звеном, изданных в последние годы в России, представлена в таблице.

Библиографические сведения	Англо-русский словарь		Русско-английский Политехнический словарь	
	70—80-е гг.	90-е гг.	70—80-е гг.	90-е гг.
Год первого издания (начала издания)	1972	1993	1980	(1996)
Число томов	2	3	1	4
Количество слов (терминов), тыс.	150	250	(90)	(>500)

Несмотря на внушительный рост объемов и несомненное повышение общего лексикографического уровня, новейшие традиционные двуязычные словари, изданные в России, содержат лексикографические лакуны, имеют недостаточный спектр проработки многозначных слов и, естественно, не поддаются текущей актуализации в период межпубликационного периода, а также не позволяют получать удобных форматных распечаток в сетевых режимах работы, что заметно снижает неоспоримую информационную ценность данных лексикографических произведений.

Таким образом, как в России, так и в Украине существует проблема создания двуязычных универсальных электронных словарей. Здесь необходимо подчеркнуть два существенных, концептуальных обстоятельства. Во-первых, универсальность проектируемого двуязычного электронного словаря предполагает слияние лексикографических полей общей и специальной лексик языка в единое семантически формализованное поле, приближающееся к реальному совокупному лексическому узусу охватываемой пары языков. Такое монолитное лексикографическое произведение будет обладать повышенной лексической наполняемостью, принципиально отличаясь от раздельных и распыленных по отраслевому признаку двуязычных словарей прежних поколений. Это позволит значительно оперативнее решать задачи лингвистического перекодирования информации. Во-вторых, данный гипотетический двуязычный универсальный электронный словарь будет ориентирован как непосредственно на полумашинный перевод, так и на создание мощной лексикографической базы для совершенствования систем машинного перевода. Известно, что даже при наличии в развитых странах многочисленных разработок в технологии машинного перевода последний остается лишь подспорьем в межъязыковых системах перекодирования. Реальной базой для межнациональных контактов на лингвистическом уровне продолжает устойчиво оставаться живой (человеческий) перевод. Однако его процесс в цивилизованных странах интенсивно перестраивается на основе использования электронных словарей, приобретая отчетливые черты полумашинной технологии.

Перспективу создания систем двуязычных универсальных электронных словарей нового поколения в развитых и развивающихся странах с обеспечением сетевого межнационального взаимодействия этих систем на унифицированной форматной основе можно с уверенностью рассматривать как кардинальный лексикографический прорыв начала будущего тысячелетия. Активное участие основных славяноязычных звеньев (особенно русско- и украиноязычного) в разработке этих уникальных систем является вопросом национального престижа.

Попытаемся кратко проанализировать практические цели, преимущества, источники формирования, лингвистические и программные принципы построения, например, славяноязычно-английских (русско-английского украинско-английского и др.) универсальных электронных словарей. Практические цели заключаются в создании мощной формализованной лексикографической базы для оперативного перекодирования текстовой информации с русского (украинского) языка на английский и в коммерческом использовании базы как на основе информационного обслуживания потребителей, так и путем рыночной реализации целевых распечаток отдельных блоков информационного массива (например, в виде отраслевых словарей) на электронных или бумажных носителях. Преимущества разрабатываемого словаря состоят в лексической универсальности информации, в увеличении ее объема (в идеале с приближением к реальному лексическому узусу), оптимальном режиме актуализации лексической информации, возможности сетевого сопряжения с ее зарубежными электронными источниками, структурной простоте формата и поэлементном использовании отраслевых помет, ориентирующем потребителя на фактографические источники детализации узкоспециальных слов. К источникам формирования словаря относятся отраслевые электронные словари, произведения традиционной двуязычной лексикографии, фактографические источники, содержащие элементы перевода, толкования и номинации понятий, продукты переводческого процесса и дедуктивное расширение лексической информации.

Основными лингвистическими принципами построения данного электронного словаря являются: двуязычность, универсальность, семантическая структура, маркирование и актуализация. Попытаемся кратко их охарактеризовать.

Очевидно, лексикографическая стратегия должна включать в себя естественное движение от простого к сложному, которое принципиально сводится к следующему генерационному ряду: одноязычность → двуязычность → многоязычность. В данном случае это принципиальное положение предусматривает предварительную тщательную подготовку лексикографических баз славяно- и англоязычного звеньев. На практике это означает: наличие солидной машинной базы входного языка (в России, как указывалось, она именуется машинным фондом русского языка), так как формализованная англоязычная лингвистическая база развита достаточно хорошо; последующее сопряжение лексикографических массивов обеих баз. Заметим попутно, что формирование многоязычных универсальных электронных словарей на указанной основе, очевидно, представляет собой еще более сложную задачу.

Универсальность словаря обусловливается оригинальной разработкой входного славяноязычного словника, объединяющего общую (общеупотребительную, общелитературную) лексику и специальную: преимущественно термины, а также значительную часть номенклатурных названий, торговых (фирменных) названий, специальных сокращений и других знаковых единиц, употребляемых в специальных текстах и имеющих лексическую или частично лексическую природу (лексические ктематонимы и т.п.).

Гнездовая описательная интерпретация терминов, характерная для традиционных специальных словарей, не предусмотрена, однако отсутствие таких обычно семантически прозрачных лексикографических описаний компенсируется несвойственной прежним словарям повышенной полиэквивалентностью лексикографической интерпретации входных слов.

Семантической структуре данного словаря будут присущи две взаимосвязанные особенности: наличие повышенной полиэквивалентности описания входных слов и диктуемое этим обстоятельством применение семантических ключей многозначных слов. В результате формируемая общая структура словаря будет отличаться существенной новизной.

Повышенная лексическая насыщенность словаря будет достигнута путем разработки развитой системы лексикографического маркирования в виде многочисленных специальных отраслевых помет и обычных лингвистических помет, которые могут быть сведены в общий вводный алфавитный список.

Машинная природа словаря обеспечивает возможность его оперативной текущей актуализации по источникам формирования. При этом особую генеративную ценность представляют выборки эквивалентов новой специальной лексики, формируемые в результате переводческого процесса. Они характеризуются исследователями в области лексикографии и переводоведения как "билингвистическая неологическая специальная лексическая информация" [3].

Уточненная интерпретация программных принципов построения двуязычного универсального электронного словаря в значительной мере будет зависеть от реального процесса увязывания его формата с форматом развитых сетевых лексикографических систем, функционирующих по каналам Интернет и Термнет или иным специальным каналам. Естественно, этот процесс может возникнуть лишь после окончания разработки словаря. Здесь можно указать лишь на общие положения, касающиеся программного обеспечения словаря. Идеальный словарь должен быть большим по объему, но давать возможность достаточно быстрого поиска слова. Следует обеспечить наличие простого и понятного интерфейса, но в то же время допускать широкие возможности варьирования форм представления запросов. Из различных требований к организации словаря с помощью ЭВМ особое значение приобретают защита данных от несанкционированного доступа, т.е. от возможностей разрушения и искажения записей. Необходимо также позаботиться о разработке механизма расширения словаря за счет новых терминов, причем новые сведения следует извлекать из источников, заслуживающих доверия, в связи с чем можно ввести индекс качества вводимой информации. Особое внимание нужно обратить на качество подачи словарной информации, которая должна отражать связи между словами и их значениями. Требуется наличие развитой справочной системы для пользователя словаря. Чтобы облегчить разработку программного обеспечения для универсального электронного словаря, можно воспользоваться анализом структуры уже существующих словарей — таких, как Контекст 3.0, Мультилекс 1.0, RusLan 1.0, POLYGLOSSUM, Lingvo.

Список литературы: 1. *Караулов Ю.Н.* Машинный фонд русского языка: идеи и суждения. М.: Наука, 1986. 240 с. 2. *Убин И.И.* Перевод с помощью ЭВМ. М.: Всесоюз. центр пер. (ВЦП). 1986. 80 с. (Обзор. информ. / ВЦП; Вып. 11). 3. *Михайлов В.И.* О системе текущего отслеживания эквивалентов неологизмов специальной лексики из неопубликованных научно-технических переводов. Х., 1985. 51с. Деп. в ИНИОН РАН 19.05.86, № 25281 52/86000 2587 Деп. 25281.

Поступила в редколлегию 29.12.97

Проблемы бионики
1998. Вып. 49. С. 130—134

УДК 518.5

А.Н. ШЕВЧЕНКО, О.В. ТОНИЦА

МОДЕЛИРОВАНИЕ ГЕОМЕТРИЧЕСКИХ ОБЪЕКТОВ В СИСТЕМАХ АНАЛИЗА ФИЗИЧЕСКИХ ПОЛЕЙ

При построении систем исследования задач расчета физико-механических полей важен учет допусков на геометрические характеристики, так как большинство реальных процессов являются стохастическими, например варьирование конкретных технических данных в пределах допусков, обусловливаемых технологиями. В данной работе рассматриваются методы и алгоритмы построения описаний геометрических областей, задаваемых с допусками, и их анализа при решении краевых задач математической физики.

Основные источники нечеткости в моделях физических полей таковы: на чертежах инженерной документации размеры деталей задаются с допусками, связанными с закономерностью проектирования и технологиями производства; в условиях реального производства возникает необходимость в изменении допусков; в задаче синтеза заданы поле или его характеристики (например, топология, исходные размеры области) в виде функции поля или ее дифференциальных характеристик, которые имеют допуски.

Приведем основные понятия теории нечетких множеств [1; 2]. Формализация нечетких понятий предполагает отказ от основного утверждения классической теории множеств о том, что некоторый элемент может либо принадлежать, либо не принадлежать множеству. При этом вводится специальная характеристическая функция принадлежности, которая принимает значения из интервала [0, 1].

Пусть E — универсальное множество, A — подмножество E: $A \subset E$.

Математический объект, определяемый выражением

$$A = \{x_1 \mid \mu_A(x_1), x_2 \mid \mu_A(x_2), \ldots, x_n \mid \mu_A(x_n)\},$$

где x_i — элемент E, $\mu_A(x_i)$ — степень принадлежности x_i к A, называют нечетким подмножеством множества E и обозначают $A \subset E$.

Рассмотрим наиболее употребительные операции над нечеткими множествами [2]. Пусть A и B — два нечетких подмножества E, а M = [0, 1] — множество принадлежностей. Будем говорить, что A включается в B, т.е.

$$(A \subset B), \text{ если } \forall x \in E : \mu_A(x) \geq \mu_B(x).$$

Подмножества A и B равны ($A = B$) тогда и только тогда, когда

$$\forall x \in E : \mu_A(x) = \mu_B(x).$$

Определим объединение $A \cup B$ как наименьшее нечеткое множество, которое содержит как A, так и B:

$$\forall x \in E : \mu_{A \cup B}(x) = \max(\mu_A(x), \mu_B(x)).$$

Пересечение $A \cap B$ определяют как наибольшее нечеткое подмножество, содержащееся одновременно в A и B:

$$\forall x \in E : \mu_{A \cap B}(x) = \min(\mu_A(x), \mu_B(x)).$$

Подмножества A и B дополняют друг друга $A = \bar{B}$, если

$$\forall x \in E : \mu_B(x) = 1 - \mu_A(x).$$

В алгебре нечетких множеств имеют место все тождества алгебры множеств [2], кроме закона исключения третьего и закона противоречия.

Проанализируем построение описаний геометрических объектов в классе функций нечеткой логики. Сложное нечеткое множество можно рассматривать как множество, построенное с помощью операций $\cap, \cup, \neg$ и простых нечетких множеств, функции принадлежности которых известны. Пусть построено теоретико-множественное описание D сложного нечеткого множества из исходных нечетких множеств D_i. Если в таком описании выполнить замену множеств D_i на соответствующие функции $\mu_{D_i}(x)$, знаков $\cap$ на min, знаков $\cup$ на max, $\neg D_i$ на $(1 - \mu_D(x))$, то получим функцию принадлежности сложного нечеткого множества. Рассмотрим область, представляющую собой прямоугольник с двумя непересекающимися окружностями внутри него. Разобьем границы объекта на части с одинаковыми допусками. Будем рассматривать сложное нечеткое множество как построенное из простых: D_1 — внутренность прямоугольника, D_2, D_3 — внешности окружностей. Теоретико-множественное описание данного множества имеет вид

$$D = D_1 \cap D_2 \cap D_3 \cdot$$

Следовательно, функция принадлежности для D записывается в форме

$$\forall x \in E : \mu_D(x) = \mu_{D_1 \cap D_2 \cap D_3}(x) =$$
$$= \min(\mu_{D_1}(x), \mu_{D_2}(x), \mu_{D_3}(x)).$$

Функции принадлежности чаще всего задаются вычислимыми функциями, т.е. алгоритмами вычислений. Проведенные исследования показали, что для построения функции принадлежности рационально использовать следующие классы алгоритмов: 1) характеристик по дискретному множеству характерных точек; 2) интегральных характеристик.

Проанализируем первый класс алгоритмов. Анализ по дискретному множеству характерных точек имеет место для кусочно-непрерывных границ. Характерные точки контура находятся следующим образом. Будем рассматривать участок некоторого контура, заданный упорядоченным множеством точек. Пусть $f_{i,i+1}(x)$, $x \in R^2$ — левая часть уравнения прямой, проведенной через точки B_i, B_{i+n}. Тогда, если имеет место равенство $\operatorname{sign} f_{i+n}(x_{i+2n}) = \operatorname{sign} f_{i,i+2n}(x_{i+3n})$, точки B_i, B_{i+n}, B_{i+2n} принадлежат выпуклому (вогнутому) участку контура; в противном случае в объединении окрестностей точек B_{i+2n}, B_{i+3n} радиуса n содержится точка перегиба. Эта окрестность сужается к точке перегиба путем более детального ее анализа по той же системе правил при некотором $n_1 < n$. Значения n, n_1 выбираются экспериментально. Последовательно обходя контур, алгоритм определяет точки перегиба, разбивая тем самым контур на выпуклые (вогнутые) его участки, характеризующиеся постоянным знаком кривизны. В частном случае, для класса многоугольных областей, характерными точками являются вершины.

Опишем алгоритм вычисления функции принадлежности для нечеткого множества, заданного произвольным многоугольником. Пусть задан эталонный n-угольник и пусть $f_i = 0$ — уравнения прямых, участки которых составляют его границу, а ε — заданный допуск по всей границе. Пусть $f = f(\wedge_1, \vee_1, f_i)$ — аналитическое описание n-угольника, в котором f_i — левые части формообразующих прямых, а $\wedge_1$, $\vee_1$, — R-операции [3]. Выражения

$$f + \varepsilon = 0, f - \varepsilon = 0$$

описывают зоны размытости. Описание зоны размытости построим в виде

$$f^* = (f - \varepsilon) \wedge_1 (f + \varepsilon) \cdot$$

Функция f^* положительна в области размытости границы, равна нулю на границах размытости и отрицательна в остальной части пространства.

Примем такую последовательность правил в качестве вычислимой функции принадлежности:

1. Конкретизируем исходные данные: задан проверяемый n-угольник, для которого необходимо вычислить функцию принадлежности нечеткому множеству, заданному описанным эталонным n-угольником с допуском ε.

2. Координаты каждой вершины проверяемого многоугольника подставим в выражение описания зоны размытости, вычислим f^*.

3. Подсчитаем количество вершин, в которых $f^* \geq 0$. Пусть таких вершин будет m, $m>n$. Вычислим отношение m/n. Оно и будет значением функции принадлежности для проверяемого n-угольника.

Рассмотрим класс алгоритмов интегральных характеристик объекта по контуру и по площади. Криволинейный интеграл по замкнутому контуру l можно найти путем вычисления функции f^* для точек границы с заданным шагом [4]: $\oint f(x,y)dl$. В качестве вычислимой функции принадлежности можно принять такую последовательность действий:

1. Конкретизируем исходные данные: задан n-угольник, для которого необходимо вычислить функцию принадлежности нечеткому множеству, заданному описанным эталонным n-угольником с допуском ε.

2. .Вычислим криволинейный интеграл по контуру границы проверяемого (текущего) n-угольника :

$$L_T = \oint\limits_{l_T} f^*(x,y)dl \cdot$$

3. Найдем криволинейный интеграл по контуру границы эталонного n-угольника:

$$L_Э = \oint\limits_{l_Э} f^*(x,y)dl \cdot$$

4. Вычислим функцию принадлежности текущего n-угольника эталонному:

$$\mu_{L_Э}(L_T) = \frac{\oint\limits_{l_T} f^*(x,y)dl}{\oint\limits_{l_Э} f^*(x,y)dl}.$$

Возможен также следующий алгоритм вычисления функции принадлежности:

1. Конкретизируем исходные данные: задан текущий n-угольник и эталонный с допуском ε.

2. Найдем площадь проверяемой многоугольной области D_T путем вычисления двойного интеграла [4]:

$$S_T = \iint\limits_{D_N} f(x,y)dxdy.$$

3. Найдем интеграл по области, представляющей собой пересечение эталонной и текущей областей:

$$\mu_{D_Э}(D_T) = \frac{\iint\limits_{D_Э \cap D_T} f(x,y)dxdy}{\iint\limits_{D_T} f(x,y)dxdy}.$$

Последний алгоритм имеет существенный недостаток: отсутствует учет топологии. Для достижения повышенной точности и обеспечения устойчивости вычисления функции принадлежности целесообразно использовать комбинации данных методов[5; 6].

При решении краевых задач математической физики по предлагаемым алгоритмам объем вычислений значительно возрастает. Поэтому актуальны вопросы параллельных реализаций. Разработанная методика может быть использована в диалоговых системах проектирования и экспертных системах.

Список литературы: 1. *Заде Л.* Понятие лингвистической переменной и его применение к принятию приближенных решений: Пер. с фр. М.: Мир, 1976. 165 с. 2. *Кофман А.* Введение в теорию нечетких множеств: Пер. с фр. М.: Радио и связь, 1982. 432 с. 3. *Рвачев В.Л.* Теория R-функций и некоторые ее приложения. К.: Наук. думка, 1982. 550 с. 4. *Будак Б.М., Фомин С.В.* Кратные интегралы и ряды. М.: Наука, 1965. 607 с. 5. *Шевченко А.Н., Тоница О.В.* Моделирование физических полей с использованием теории R-функций и нечеткой логики // Методы оптимизации технических и информационных систем. К., 1995. С. 64–70. 6. *Шевченко А.Н., Тоница О.В.* Моделирование физико-механических полей методами теории R-функций и нечеткой логики // Междунар. конф. «Математические модели и численные методы механики сплошных сред», Новосибирск, 1996 г.: Тез. докл. Новосибирск, 1996. С. 114–115.

Поступила в редколлегию 06.04.98

Проблемы бионики
1998 Вып. 49. С. 135—140

ХТУРЭ

УДК 518.5

А.Н. ШЕВЧЕНКО, О.В.ТОНИЦА

РАЗВИТИЕ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ АНАЛИЗА ФИЗИЧЕСКИХ ПОЛЕЙ СЕРИИ «ПОЛЕ»

Существуют производственные задачи, при решении которых необходимо учитывать и совместно перерабатывать сложную геометрическую, логическую и аналитическую информацию. К таким задачам относятся проблемы математической физики, связанные с инженерными расчетами физико-механических полей, определяющих основные качественные характеристики изделий. Возрастающий интерес к расчетам физических полей объясняется тем, что они необходимы в теплофизике, теории упругости и пластичности, магнитной гидродинамике и других областях науки, достижения которых имеют первостепенное значение для научно-технического прогресса [1;2]. Данная работа посвящена развитию интеллектуальных систем исследования физических полей серии «Поле» для учета технологических допусков и методических погрешностей.

Математические модели физико-механических полей могут быть представлены в виде краевых задач для уравнений с частными производными при определенных краевых условиях. Специфическая особенность поля как объекта моделирования — его зависимость не только от характера физических законов, учитываемых соответствующими уравнениями, но и от формы, взаимного расположения тел, в которых возбуждаются поля, от конфигурации площадок их взаимодействия и других геометрических и физических факторов. При исследовании и решении перечисленных проблем весьма важно создать эффективные вычислительные методы, которые позволили бы решить поставленные задачи и учесть при этом геометрическую, логическую и аналитическую информацию. Такую информацию необходимо приводить к единому аналитическому виду, позволяющему включать ее в разрешающий алгоритм. Особенно актуальна разработка методов решения задач, которые имели бы универсальный характер и не требовали бы от исследователя глубокого знания теории. Универсальность позволяет использовать методы системного программирования для автоматизации научных исследований в математической физике.

К широко известным универсальным методам относятся вариационные и проекционные методы, которые принято называть прямыми [1]. Их характерная особенность — сведение краевых задач к системам линейных алгебраических уравнений. Приближенное решение задачи отыскивается в виде линейных комбинаций так называемых координатных функций, которые удовлетворяют, если это необходимо, краевым условиям рассматриваемой задачи, а также требованиям полноты и условиям аппроксимационной универсальности. Построение координатных функций, удовлетворяющих перечисленным условиям для областей практически произвольной формы, долгое время оставалось проблематичным. Рассмотреть данный вопрос с общих позиций удалось после создания конструктивного математического аппарата теории R-функций [1; 2], позволившего получить решение соответствующей математической задачи в виде формулы, называемой структурой решения и содержащей явную зависимость от геометрических и физических параметров. Полученные теоретические результаты использовались при создании современной технологии программирования в математической физике, реализации проблемно-ориентированных языков и специализированных систем серии "Поле". Пользователи последних указывают формулировку задачи, исходные данные, требуемую форму выдачи результатов на языке высокого уровня, который максимально приближен к общепринятому языку описания постановки задачи и алгоритма ее решения. По заданию пользователя система "Поле" создает вычислительную схему, а затем автоматически синтезирует рабочую программу решения задачи. Опыт применения проблемно-ориентированных языков и специализированных систем серии «Поле» показывает, что они существенно упрощают и ускоряют наиболее трудоемкие этапы вычислительных экспериментов: программирование и отладку модулей, решение задач и анализ результатов.

При конструировании на аналитическом уровне решений краевых задач для сложных областей со сложным характером условий применяется метод R-функций в сочетании с вариационными и проекционными методами. Благодаря тому что метод R-функций позволяет автоматизировать процесс преобразования геометрической информации в аналитическую, результат решения краевой задачи можно рассматривать как некоторое соотношение, включающее всю информацию о краевой задаче. Однако применение операций дифференцирования делает такие соотношения весьма громоздкими, труднообозримыми и неудобными в обращении. Их использование при решении краевой задачи невозможно без специального аппарата формализации, позволяющего свести алгоритм решения к последовательности простых операций. Сложные математические объекты объединяются средствами алгебры дифференциальных кортежей [2], а затем применяются обычные алгебраические методы преобразования этих соотношений.

При решении краевых задач математической физики используется традиционная схема реализации прямых методов, которая сводится к последовательному выполнению следующих этапов: учет геометрической информации; формирование разрешающих систем линейных алгебраических уравнений; решение систем уравнений или проблемы собственных чисел и векторов; обработка результатов вычислений.

Системы серии «Поле» являются сложными комплексами программ, предназначенными для программирования и решения краевых задач, сформулированных для уравнений с частными производными при произвольных краевых условиях и сложной геометрии области. Организационно данные системы состоят из двух частей — функционального и системного наполнения. Инструментальная система «Поле» может быть использована для исследования и решения краевых задач математической физики. Проблемной ориентацией служит решение краевых задач для дифференциальных уравнений с переменными коэффициентами без ограничений на характер краевых и начальных условий, форму области и участков границы. Входные языки системы «Поле» позволяют вводить всю необходимую информацию о краевых задачах математической физики.

В настоящее время автоматизация программирования в области математической физики достигла значительного прогресса. Эффект от использования проблемно-ориентированных языков и специализированных систем состоит в сокращении времени решения многих научно-технических задач, а также в создании базы для перехода к индустриальным методам и новой технологии программирования. Такие языки и системы являются инструментальной базой для проведения вычислительных экспериментов, освобождающей математиков и инженеров от рутинной и не соответствующей их специальности работы по составлению и отладке громоздких программ. Для проведения вычислительных экспериментов в области математической физики во многих случаях необходимо глубокое изучение моделируемого процесса или явления, познание законов природы процессов и их проявлений в сложном взаимодействии. Кроме того, при решении задач математической физики приходится учитывать вопросы сходимости, устойчивости вычислительного процесса, точности вычислений, эффективности применяемых методов и т.д. Эти обстоятельства создают дополнительные трудности при разработке математического обеспечения для решения задач математической физики. С помощью системы «Поле» решено много научно-технических и инженерных задач. Например, произведены расчет и оптимизация различных конструкций, для которых определяются основные качественные характеристики (прочность, добротность, долговечность, надежность и т.д.). При решении таких сложных задач возникает необходимость в проведении все более тонких расчетов температурных, деформационных, силовых и других физико-механических полей с учетом самых разных факторов физического и геометрического характера.

Для решения перечисленных задач используются методы и конструктивные средства теории R-функций, позволяющие с единых позиций решать вопросы учета и совместной переработки сложной геометрической, логической и аналитической информации, а также системы серии «Поле», характерной особенностью которых является возможность явного задания на проблемно-ориентированном языке необходимых физических и геометрических параметров. Это позволяет проводить многовариантные вычислительные эксперименты, переходить от решения одной задачи к другой, изменяя при этом форму рассматриваемых объектов и различные физические и математические параметры. Анализируя методы теории R-функций и ее программное обеспечение, следует отметить, что они представляют собой достаточно эффективную совокупность средств для проведения вычислительных экспериментов по расчету различных физико-механических полей в объектах сложной формы. Различные тестовые примеры, сравнение результатов с точными решениями, согласование решений реальных задач с данными физических экспериментов подтверждают достоверность результатов и свидетельствуют о целесообразности применения описанных программных и языковых средств для инженерных расчетов. Основное направление развития теории R-функций связано с созданием новых, более простых и эффективных конструктивных средств и с расширением предметной области системы «Поле».

Рассмотрим подходы к усовершенствованию систем анализа физических полей. При построении систем исследования задач расчета полей важен учет стохастического характера погрешностей измерений, допусков на геометрическую и физическую информацию и погрешностей округления. В связи с этим возникает необходимость в развитии существующих систем расчета полей для многовариантных задач с целью получить допуски на решение и последующее экспертное заключение. Вычисления в системах расчета полей, как правило, носят детерминированный характер, в то время как реальные процессы в определенной степени являются стохастическими, содержат в себе некоторую нечеткость. Для учета последней нужно так преобразовать существующую схему исследования физических полей, чтобы в результате многовариантного счета получить более точное «нечеткое» решение, которое будет ближе к реальности. Целесообразно ввести в схему решения учет допусков, т. е. источников нечеткости, наиболее сильно влияющих на результирующее решение. Практика показывает, что таких источников, как правило, три: допуски модели (на геометрические и физические характеристики), ошибки метода («усечение» ряда, ошибки интегрирования, решения систем линейных уравнений) и погрешности округления[3]. Необходимо установить комплексное влияние варьирования величин в пределах допусков и исследовать возможности построения допусков на решение. В связи с этим большой интерес представляет разработка систем исследования полей, ориентированных на многовариантное решение краевых задач с целью учесть варьирование определенных величин в пределах заданных допусков.

Для иллюстрации нечеткости реальной задачи моделирования рассмотрим задачу Дирихле для дифференциального уравнения общего вида. Пусть заданы область D и краевые условия с учетом допусков:

$$Au = f;$$

$$U\big|_{\Gamma_{щ}} = \varphi_i^* \Leftrightarrow \varphi \pm \Delta_\varphi;$$

$$D \Leftrightarrow D \pm \Delta_D.$$

Изменяя допуски на геометрию и краевые условия в заданных пределах, получаем допуски на U, которым должно удовлетворять решение реальной краевой задачи.

Предлагаемая методика моделирования для задачи анализа включает в себя следующие этапы: формирование допусков на решение; решение реальной краевой задачи; формулирование экспертного заключения о приемлемости найденного решения.

Рассмотрим модель физического поля, например задачу Дирихле[1]. Четкую модель поля будем обозначать через m:

$$Au = f;$$

$$U\big|_{\Gamma_i} = \varphi_i;$$

$$D.$$

Ее решение структурным методом строится в виде

$$U_n = \sum_{i=1}^{n} C_i \omega P_i + \Psi,$$

где ω — аналитическое описание области D; Ψ — функция, продолжающая краевые условия внутрь области; C_i — неопределенные коэффициенты.

В соответствии с описанными выше источниками нечеткости построим нечеткую модель поля[2]:

$$Au = f;$$
$$U\big|_{\Gamma_i} = \varphi_i \Leftrightarrow (\varphi \pm \Delta_\varphi, \mu_{\varphi^*}(\varphi^m));$$
$$D \Leftrightarrow (D \pm \Delta_D, \mu_{D^*}(D^*)),$$

которую будем обозначать через M. Для выборки $\{m\} \subset M$ нужно получить нечеткое решение

$$U_n^* = \sum_{i=1}^{n} C_i \omega^m P_i + \Psi_i(\varphi_i), \mu_{U_n^*}(U_n^m).$$

Через M_1 обозначим модель, учитывающую варьирование физических величин в пределах заданных допусков, а через M_2 — учитывающую варьирование геометрических характеристик.

Модель M_1 будет иметь вид

$$Au = f;$$
$$U\big|_{\Gamma_i} = \varphi_i \Leftrightarrow (\varphi \pm \Delta_\varphi, \mu_{\varphi}\ (\varphi^m));$$
$$D.$$

Модель M_2 будет выглядеть следующим образом:

$$Au = f;$$
$$U\big|_{\Gamma_i} = \varphi;$$
$$D \Leftrightarrow (D \pm \Delta_D, \mu_{D^*}(D)).$$

Для выборки $\{m\} \subset M$ получим нечеткое решение

$$\widetilde{U_n^*} = \sum_{i=1}^{n} \widetilde{C}_i \omega^m P_i + \widetilde{\Psi}_i(\varphi_i), \mu_{U_n^*}(U_n^m).$$

Функцию принадлежности текущей области по отношению к эталонной находим с помощью вычисления поверхностного интеграла по исследуемой области.

Разработанная методика применяется для развития систем исследования физических полей.

Список литературы: 1. *Рвачев В.Л.* Теория R-функций и некоторые ее приложения. К.: Наук. думка, 1982. 550 с. 2. *Рвачев В.Л., Шевченко А.Н.* Проблемно-ориентированные языки и системы для инженерных расчетов. К.: Тэхника, 1988. 199 с. 3. *Шевченко А.Н., Тоница О.В.* Моделирование физических полей с использованием теории R-функций и нечеткой логики // Методы оптимизации технических и информационных систем. К., 1995. С. 64-70.

Поступила в редколлегию 06.04.98

Проблемы бионики
1998. Вып. 49. С. 141—146

ХТУРЭ

УДК 519.23-25

Н.И. ЧИСЛИН

ЭРГОДИЧЕСКИЕ СВОЙСТВА ЦЕПЕЙ МАРКОВА С ПЕРЕМЕННЫМ ЧИСЛОМ СОСТОЯНИЙ И МОДЕЛЬ УСТОЙЧИВОГО РАЗВИТИЯ ПСИХИКИ

Психика человека характеризуется последовательной сменой состояний, происходящей вследствие воздействия внешних факторов (которые можно считать случайными) и в зависимости от состояния, в котором она до этого находилась. Очевидно, что для математического описания процесса смены психических состояний можно использовать аппарат цепей Маркова. Устойчивость психики можно охарактеризовать как примерное постоянство отношения долей времени нахождения в различных психических состояниях, или, на языке описания марковских процессов, стационарность распределения цепи. Условия стационарности распределений цепей Маркова достаточно хорошо разработаны в классической теории, но следует заметить, что число возможных психических состояний, особенно на начальных этапах развития, непрерывно меняется, т.е. стационарность в общепринятом понимании невозможна. Однако если использовать модель цепей Маркова с переменным числом состояний, то в этом случае можно выделить классы распределений, аналогичные по своему значению стационарному распределению классической модели. Такие цепи рассматривались нами ранее, причем были получены условия эргодичности для специального класса таких цепей [1]. Здесь рассматриваются более общие условия эргодичности, аналогичные условиям, полученным Дж. Хайналом [2] для неоднородных цепей Маркова с постоянным числом состояний.

Прежде чем анализировать условия эргодичности для цепей Маркова с переменным числом состояний, заметим, что используемые далее понятия согласованного класса и представителя класса определены в [1]. Потребуются также следующие определения.

Определение 1. *Пусть вероятности переходов в цепи с переменным числом состояний, принимающей в моменты времени* k ($k = 0, 1, 2, \ldots$) *значения из множества состояний* I_k, *задаются последовательностью прямоугольных стохастических матриц* $\{P_k\}_{k=1}^{\infty}$. *Если для всех* $k \geq 0$, $n \geq 1$ *элементы* $h_{ij}^{(k,n)}$ *матриц*

$$H_{k,n} = \prod_{j=1}^{k+n} P_j$$

удовлетворяют предельным соотношениям

$$\lim_{n\to\infty} \max_{j\in I_{k+n+1}} \max_{i,l\in I_k} \left| h_{ij}^{(k,n)} - h_{lj}^{(k,n)} \right| = 0 ,$$

то такую цепь будем называть эргодической в слабом смысле.

Определение 2. *Пусть цепь Маркова с переменным числом состояний, в которой вероятности переходов в моменты времени $k = 1, 2, \ldots$, между множествами состояний I_k ($k = 0, 1, 2, \ldots$) задаются последовательностью прямоугольных стохастических матриц $\{P_k\}_{k=1}^{\infty}$, является эргодической в слабом смысле. Пусть также существует согласованный класс распределений π, представители которого π_k заданы на соответствующих множествах состояний I_k. Если для распределений вероятностей состояний цепи p_k в моменты времени k ($p_{k+n+1} = p_k H_{k,n}, k \geq 0$), независимо от характера начального распределения p_0, выполняются предельные соотношения*

$$\lim_{k\to\infty} \max_{i\in I_k} |p_{ki} - \pi_{ki}| = 0,$$

где p_{ki}, π_{ki} – i-е компоненты векторов распределений p_k и π_k соответственно, то такую цепь будем называть эргодической в сильном смысле с предельным распределением из согласованного класса распределений π.

Рассмотрим условия, при которых марковская цепь с переменным числом состояний будет эргодической в слабом смысле. Для этого необходимо выделить классы прямоугольных стохастических матриц, которые удовлетворяли бы следующим требованиям: 1) произведение двух матриц из которых хотя бы одна принадлежит такому классу, также будет принадлежать этому классу; 2) наличие в последовательности матриц переходных вероятностей за единицу времени некоторой цепи Маркова с изменяющимся числом состояний бесконечного числа матриц, принадлежащих такому классу, гарантирует слабую эргодичность этой цепи.

Определение 3. *Пусть в прямоугольной стохастической матрице $P = \|p_{ij}\|_{i\in I', j\in I''}$ для любых двух строк $l, m \in I'$ существует по крайней мере один столбец $n \in I''$ такой, что одновременно выполняются* *неравенства* $p_{ln} > 0$, $p_{mn} > 0$. *Такие матрицы будем называть перемешивающими.*

В качестве характеристики "перемешивающей силы" матриц P_k и $H_{k,n}$ используем меру

$$\mu_S(P) = \min_{l,m \in I'} \sum_{n \in I''} \min(p_{ln}, p_{mn}).$$

Мера $\mu_S(\cdot)$ обладает такими свойствами: 1) $0 \le \mu_S(P) \le 1$ для любой стохастической матрицы P; 2) $\mu_S(P) = 0$ только в том случае, когда матрица P – не перемешивающая; 3) $\mu_S(P) = 1$ тогда и только тогда, когда все строки матрицы P идентичны.

Следующие леммы показывают, что класс перемешивающих матриц замкнут по умножению.

Лемма 1. *Если L и M – прямоугольные стохастические матрицы такие, что для них определено произведение $Q = LM$, то $\mu_S(Q) \ge \mu_S(L)$.*

Лемма 2. *Если L и M – прямоугольные стохастические матрицы такие, что для них определено произведение $Q = LM$, и хотя бы одна из них – перемешивающая, то перемешивающей является и матрица Q.*

Далее потребуется не только мера "перемешивающей силы" отдельной матрицы, но и мера "аккумулированной перемешивающей силы" произведения матриц. Для этого воспользуемся мерой $\mu_D(\cdot)$, которая определяется следующим образом: для всякой матрицы $P = \|p_{ij}\|_{i \in I', j \in I''}$

$$\mu_D(P) = \max_{j \in I''} \max_{i,k \in I'} |p_{ij} - p_{kj}|.$$

Заметим, что для любых k и n $\mu_D(H_{k,n}) = 0$ только тогда, когда все строки матрицы $H_{k,n}$ идентичны, т.е. $\mu_D(H_{k,n})$ является грубым аналогом для $1 - \mu_S(H_{k,n})$. Но, если $1 - \mu_S(H_{k,n})$ с ростом n может только убывать, $\mu_D(H_{k,n})$ может как убывать, так и возрастать.

Лемма 3. *Если $F = PG$, где P и G – прямоугольные стохастические матрицы, то*

$$\mu_D(F) \le (1 - \mu_S(P))\mu_D(G).$$

Доказательства лемм 1–3 строятся по схеме доказательств соответствующих лемм в [2].

Чтобы сформулировать общие условия слабой эргодичности, необходимо рассмотреть следующие теоремы.

Теорема 1. *Для любой цепи Маркова с переменным числом состояний и для любых k и n справедливы неравенства*

$$\mu_D(H_{k,n}) \le \prod_{j=1}^{n-1}(1-\mu_S(P_{k+j})\mu_D(P_{k+n}).$$

Доказательство следует из леммы 3.

Теорема 2. *Цепь Маркова с переменным числом состояний является эргодической в слабом смысле в том и только в том случае, когда из последовательности матриц переходных вероятностей за один шаг $\{P_k\}_{k=1}^{\infty}$ можно выделить такую подпоследовательность $\{P_{k_i}\}_{i=1}^{\infty}$, что ряд $\sum_{i=1}^{\infty}\mu_S(H_{k_i,k_{i+1}-k_i})$ расходится.*

Доказательство. Известно, что ряд $\sum_{i=1}^{\infty}\mu_S(H_{k_i,k_{i+1}-k_i})$ расходится в том и только в том случае, когда

$$\prod_{i=1}^{\infty}(1-\mu_S(H_{k_i,k_{i+1}-k_i})) = 0.$$

Отсюда, исходя из теоремы 1 и из того, что для всякого k справедливо неравенство $\mu_D(P_k) \le 1$, следует достаточность условия теоремы.

Чтобы доказать его необходимость, предположим, что цепь эргодична. Тогда, по определению, $\mu_S(H_{i,n}) \to 1$ при $n \to \infty$ для всех i. Пусть $\delta > 0$ – произвольная малая постоянная. Можно выбрать k_1, k_2, ... таким образом, что

$$\mu_S(H_{k_1,k_2-k_1}) \ge \delta,\ \mu_S(H_{k_2,k_3-k_2}) > \delta,\ \mu_S(H_{k_3,k_4-k_3}) > \delta,\ \dots,$$

т.е. условие теоремы выполняется, что и завершает доказательство.

Теорема 3. *Если последовательность матриц переходных вероятностей за один шаг $\{P_k\}_{k=1}^{\infty}$ некоторой цепи Маркова с переменным числом состояний содержит подпоследовательность перемешивающих матриц $\{P_{k_i}\}_{i=1}^{\infty}$, то такая цепь является эргодической в слабом смысле.*

Доказательство теоремы 3 следует из теоремы 2 и лемм 2 и 3 очевидным образом.

Из определения 2 видно, что слабая эргодичность служит необходимым условием для сильной эргодичности цепей Маркова с переменным числом состояний. Рассмотренные выше теоремы позволяют определить условия, при которых такие цепи будут эргодическими в сильном смысле.

Теорема 4. *Пусть цепь Маркова, определенная на последовательности множеств состояний* $\{I_k\}_{k=0}^{\infty}$, *вероятности переходов между которыми задаются прямоугольными стохастическими матрицами из последовательности* $\{P_k\}_{k=1}^{\infty}$, *удовлетворяет следующим условиям:*

1) существует последовательность векторов-распределений $\{\overline{p}_k\}_{k=0}^{\infty}$, $\overline{p}_k = \|\overline{p}_i(k)\|_{i \in I_k}$, *такая, что:*

а) $\sum\limits_{k=1}^{\infty} \max\limits_{j \in I_k} \left| \overline{p}_j(k) - \sum\limits_{i \in I_{k-1}} \overline{p}_i(k-1) p_{ij}(k) \right| < \infty$;

б) $\lim\limits_{k \to \infty} \max\limits_{i \in I_k} |\overline{p}_i(k) - \pi_{ki}| = 0$, *где* π_{ki} *– i-я компонента вектора-распределения* π_k *– представителя согласованного класса распределений* π;

2) цепь эргодична в слабом смысле.

Тогда такая цепь является эргодической в сильном смысле с предельным распределением из согласованного класса π.

Доказательство. В силу слабой эргодичности $\mu_S(H_{i,n}) \to 1$ при $n \to \infty$ для всех i, т.е. при достаточно больших n матрица $H_{i,n}$ достаточно близка к некоторой матрице с идентичными строками $S_{i,n}$, или, что то же самое, матрица $V_{i,n}H_{i,n} = H_{i,n} - S_{i,n}$ стремится к нулевой матрице. С другой стороны, каждую матрицу переходных вероятностей P_k можно представить в виде $P_k = S_k + V_k$, где S_k – матрица с идентичными строками, которые совпадают с соответствующим вектором-распределением $\overline{p}_k$. Исходя из этого, для $H_{i,n}$ получим выражение

$$H_{i,n} = \prod_{k=1}^{n} S_{i+k} + \sum_{j=1}^{n-1} \prod_{k=1}^{j} S_{i+k} \prod_{l=j+1}^{n} V_{i+l} + \prod_{j=1}^{n} V_{i+j} \,. \qquad (1)$$

Заметим, что в (1) из всех слагаемых только $\prod_{k=1}^{n} S_{i+k}$ является стохастической матрицей, а у остальных слагаемых суммы элементов каждой строки равны нулю. Возвратившись к представлению $H_{i,n} = S_{i,n} + V_{i,n}$, отметим, что здесь стохастической является матрица $S_{i,n}$, а у матрицы $V_{i,n}$ суммы элементов каждой строки равны нулю. Кроме того, норма матрицы $V_{i,n}$ (т.е. величина, равная наибольшему из абсолютных значений ее элементов) и норма матрицы $\sum_{j=1}^{n-1} \prod_{k=1}^{j} S_{i+k} \prod_{l=j+1}^{n} V_{i+l} + \prod_{j=1}^{n} V_{i+j}$ с ростом n стремятся к нулю, поскольку из условия 1 теоремы следует, что при $k \to \infty$ к нулю стремится норма матрицы V_k, а значит, матрицы $S_{i,n}$ и $\prod_{k=1}^{n} S_{i+k}$ при $n \to \infty$ стремятся к одной и той же матрице. Поэтому при достаточно больших n можно положить

$$S_{i,n} = \prod_{k=1}^{n} S_{i+k} . \tag{2}$$

Таким образом, при всяком i распределение цепи p_{i+n} в случае $n \to \infty$ стремится к распределению $\bar{p}_{i+n}$, а в силу условия 1, б теоремы – к распределению π_{i+n} из согласованного класса распределений π .

Рассмотренная модель, благодаря ее эргодическим свойствам, может быть, в частности, использована для задания правил относительно устойчивого поведения систем искусственного интеллекта при меняющемся составе множества возможных альтернатив.

Автор выражает признательность проф. В. А. Дикареву за постановку задачи и обсуждение полученных результатов.

Список литературы: 1. *Методы* и алгоритмы фокусировки распределений марковских процессов / А.Е. Веприк, С.Н. Герасин, В.А. Дикарев, А.А. Родзинский, Н.И. Числин. Х.: Харьк. техн. ун-т радиоэлектроники, 1997. 160 с. 2. *Hajnal J.* Weak ergodicity in non-homogeneous Markov chains // Proc. Cambridge Philos. Soc. 1957. Vol. 52. P. 239–265.

Поступила в редколлегию 03.04.98

Проблемы бионики
1998. Вып. 49. С. 147—150

ХТУРЭ

УДК 519.7

Е.В. ЖУРАВОК

МОДЕЛИРОВАНИЕ ПРОЦЕССА МОРФОЛОГИЧЕСКОЙ ОБРАБОТКИ СЛОВОФОРМ МЕТОДОМ РЕШЕНИЯ КВАНТОРНЫХ ПРЕДИКАТНЫХ УРАВНЕНИЙ

Для реализации математической модели процесса словоизменения удобно использовать метод решения кванторных предикатных уравнений, который основан на теории линейных логических операторов [1]. Найдем решение предикатного уравнения

$$P_1(Y) = \exists X(P_2(X) \wedge P_3(Y,X), \qquad (1)$$

где предикаты $P_1(Y)$ и $P_2(X)$ заданы на области $U = (u_1, ..., u_n)$, состоящей из n элементов, бинарный предикат $P_3(Y, X)$ задан на области UxU. Требуется определить предикат $P_2(X)$, считая известным предикаты $P_1(Y)$ и $P_3(Y, X)$. Учитывая, что предикатная переменная X связана квантором существования, уравнение (1) перепишем в виде

$$P_1(Y) = \bigvee_{j=1}^{n} (P_2(u_j) \wedge P_3(Y,u_j)). \qquad (2)$$

Равенство (2) выполняется только в том случае, когда оно верно для любого значения предикатной переменной Y, пробегающей множество U. Таким образом, имеем следующие n равенств:

$$P_1(u_i) = \bigvee_{j=1}^{n} (P_2(u_j) \wedge P_3(u_i,u_j)) \qquad (3)$$

для любого $j \in 1, ..., n$.

Значения предикатов $P_1(u_i)$ и $P_2(u_j)$ обозначим через y_i и x_j, где $y_i, x_j \in \{0, 1\}$ и $i, j \in 1, ..., n$. Значение бинарного предиката $P_2(u_i, u_j)$ обозначим через $p_{ij} \in \{0, 1\}$, $i, j \in 1, ..., n$. С учетом введенных обозначений равенство (3) примет вид

$$y_i = \bigvee_{j=1}^{n} x_j \wedge p_{ij}. \qquad (4)$$

Согласно (3) получим операторное выражение вида

$$P_3(X) = Y, \qquad (5)$$

где P_3 – линейный логический оператор, заданный на пространстве E, с матрицей оператора

$$P=\begin{bmatrix} p_{11}, & \dots & \dots & \dots, & p_{1n} \\ \dots & \dots & \dots & \dots & \dots \\ p_{i1}, & \dots & p_{ij} & \dots, & p_{in} \\ \dots & \dots & \dots & \dots & \dots \\ p_{n1}, & \dots & \dots & \dots, & p_{nn} \end{bmatrix} \qquad (6)$$

Таким образом, предикатное уравнение (1) эквивалентно операторному уравнению (5), определенному на логическом пространстве Е. Согласно утверждению об общем виде матрицы линейного регулярного оператора, действующего из пространства Е в себя [1], для обратимости необходимо и достаточно, чтобы в каждых строке и столбце матрицы такого оператора находился один и только один элемент, равный единице . Если матрица (6) удовлетворяет указанным выше условиям, то решение уравнения (5) имеет вид

$$X = P_3^{-1}(Y). \qquad (7)$$

Матрица обратного оператора совпадает с транспонированной матрицей оператора P_3 [1], поэтому решение операторного уравнения (7) в матричном виде следующее:

$$X = P^t \bullet Y. \qquad (8)$$

В результате решение предикатного уравнения (1) можно записать в виде

$$P_2(X) = \exists Y(P_1(Y) \wedge P_3(X,Y)). \qquad (9)$$

Если оператор P_3 не является регулярным, дать решение в виде (9) нельзя. Однако, воспользовавшись алгебраической записью предикатного уравнения (1), будем искать решение уравнения следующим образом. Операторное уравнение (5) выпишем в виде системы логических уравнений:

$$\begin{cases} \bigvee\limits_{j=1}^{n} (p_{1j} \wedge x_j) = y_1; \\ \dots\dots\dots\dots\dots\dots \\ \bigvee\limits_{j=1}^{n} (p_{ij} \wedge x_j) = y_i; \\ \dots\dots\dots\dots\dots\dots \\ \bigvee\limits_{j=1}^{n} (p_{nj} \wedge x_j) = y_n . \end{cases} \qquad (10)$$

Пусть единицы стоят в Y на местах $(d_1, \ldots, d_{y(0)}) = D$, а нули – на местах $(z_1, \ldots, z_{н(0)}) = Z$; $D \cap Z = \varnothing$; $D \cup Z = N = (1, \ldots, n)$. Множество мест, на которых стоят нули вектора X, будем обозначать как $L = (l_1, \ldots, l_{x(0)})$. Символом * будем обозначать места, на которых могут стоять нули либо единицы. Символы * стоят на местах $(m_1, \ldots, m_{x(*)}) = M$.

Сформулируем соответствующий алгоритм:

1. Производим инициализацию: $i := z_1$.
2. Формируем множество, состоящее из равных нулю координат вектора X.
   2.1. Присваиваем $j := 1$.
   2.2. Если $P_3 [i,j] = 1$, то считаем $X [j] := i_1$.
   2.3. Организуем перебор индексов j от 0 до n.
3. Индексу i присваиваем следующий элемент из множества z и повторяем переход к п. 2.1 до тех пор, пока не будут выбраны все элементы множества z.
4. Формируем множество M. Получаем некоторый логический вектор X*, состоящий из нулей и символов *.
5. Проверяем систему (10) на непротиворечивость.
   5.1. Подставляем найденный вектор в систему.
   5.2. Организуем решение полученной системы согласно формуле

$$\bigvee_{j=1}^{n} x_j \wedge p_{ij} = y_i .$$

   5.3. Проверяем совместность системы; если она несовместна, то вектор не является ее решением .
. Формируем решение системы.
   6.1. В вектор X* подставляем вместо первого символа * единицу, а вместо остальных — нули.
   6.2. Переходим к п. 5.
   6.3. Если сформированный логический вектор является решением системы, обеспечиваем его запоминание в массиве решений.
   6.4. Организуем различные подстановки нулей и единиц вместо символов * с переходом при каждом новом сочетании к п. 5.
7. Выписываем все полученные решения системы, если массив решений непуст. В противном случае производим вывод сообщения о противоречивости системы.

Программа RPY предназначена для решения кванторных предикатных уравнений описанным выше методом. Программа реализована на объектно-ориентированном языке программирования высокого уровня Borland Pascal 7.0 с использованием библиотеки TurboVision, что обеспечивает простоту ее использования и наглядность. Объем исходного кода 15 Кбайт, объем исполняемого EXE-файла 60 Кбайт. Для работы необходимы: минимум – IBM PC XT, 512 К памяти, DOS 3.3.

Метод решения кванторного предикатного уравнения удобно использовать для реализации математической модели процессов словоизменения [2]. Отношение "признаки – буквы окончания" при морфологической обработке по описанному методу имеет следующий вид:

$$P_3(x,y) = \begin{cases} 1, \text{если признак } x \text{ определяет окончание } y; \\ 0, \text{ в противном случае,} \end{cases}$$

где $x \in \{x_1, \ldots, x_n\}$ и $y \in \{y_1, \ldots, y_m\}$. Таким образом, чтобы получить информацию о том, какие признаки определяют окончание y_1, строим предикат:

$$P_1(y) = \begin{cases} 1, \text{если окончание } y = y_1, \\ 0, \text{ в противном случае.} \end{cases}$$

В результате предикат $P_2(x)$, соответствующий искомой информации, определяется кванторным уравнением вида (9) и задает следующее отношение:

$$P_2(x) = \begin{cases} 1, \text{ если признак } x \text{ определяет окончание } y_1; \\ 0, \text{ в противном случае,} \end{cases}$$

Решение данного кванторного предикатного уравнения получается из решения соответствующего операторного уравнения A^X=У в линейном логическом пространстве. Таким образом, операция поиска интересующей информации заменяется операцией операторного умножения.

Список литературы: 1. *Шабанов -Кушнаренко Ю.П.* Теория интеллекта. Математические средства. Х.: Выща шк. Изд-во при Харьк. ун-те, 1984. 144 с. 2. *Зализняк А.А.* Грамматический словарь русского языка. М.: Рус. яз., 1977. 878 с.

Поступила в редколлегию 09.04.98

роблемы бионики
998. Вып. 49. С. 151—161

УДК 681.324

М.Ю. ГАЦКИЙ

ФОРМАЛИЗАЦИЯ НЕЧЕТКИХ ЗНАНИЙ НА ОСНОВЕ МОДЕЛЕЙ НЕЧЕТКИХ АЛГОРИТМИЧЕСКИХ КВАНТОВ ЗНАНИЙ (НАКЗ-МОДЕЛЕЙ) ДЛЯ КОМПЬЮТЕРНОГО ПРИНЯТИЯ РЕШЕНИЙ В УСЛОВИЯХ НЕОПРЕДЕЛЕННОСТИ

С точки зрения бионики интеллекта актуальна проблема моделирования человеческого знаниеориентированного принятия решений в условиях неполноты данных, т.е. при некоторой неопределенности. Представляется целесообразным адаптировать метод многоуровневых алгоритмических квантов знаний (МАКЗ-метод) [1], эффективный при знаниеориентированном принятии решений на основе четких данных, к принятию решений в условиях неопределенности. Для этого требуется: а) определить векторно-матричное представление нечетких знаний; б) построить базу нечетких знаний (БНЗ) в форме системы закономерностей предметной области с помощью оператора индуктивного вывода нечетких знаний; в) синтезировать алгоритм дедуктивного вывода решения исходя из наблюдаемого нечеткого кванта знаний, с использзованием построенной БНЗ.

Адаптированный метод назовем методом нечетких алгоритмических квантов знаний (НАКЗ-методом).

Алгоритмическая формализация нечетких $\tilde{k}$ -знаний и их векторно-матричное представление

Рассмотрим основные понятия и определим исходные кванты. Согласно МАКЗ-методу [1] множество объектов принятия решений (ОПР) различной природы из заданного набора классов К обозначим как $\Omega = \{\omega\}$. Для наблюдений за объектами используется заданное множество SC шкал измерений разнотипных признаков объектов из Ω .

Приближенным неточным знаниям поставим в соответствие нечеткие множества. Обозначим через $U = \{u\}$ универсальное множество.

Определение 1. *Нечетким множеством* $\tilde{M}$ *на множестве* U *называется совокупность пар вида*

$$\tilde{M} = \left\{u | \hbar_{\tilde{M}}(u)\right\}, \qquad (1)$$

где $\hbar_{\tilde{M}} : U \to [\,0,\,1\,]$ *— отображение множества* U *на отрезок* $[\,0,\,1\,]$, *называемое нечеткой характеристической функцией достоверности (НХФД) нечеткого множества* $\tilde{M}$. Значение функции $\hbar_{\tilde{M}}(u) = \mu(u)$ для конкретного элемента $u \in U$ назовем показателем достоверности (ПД), характеризующим степень достоверности определенных знаний. Во всех случаях полагаем, что ПД $\mu(u)$ является некоторой объективной мерой того, насколько элемент $u \in U$ соответствует понятию, смысл которого формализуется нечетким множеством $\tilde{M}$.

Определение 2. *Носителем нечеткого множества* $\tilde{M}$ *называется множество*

$$Z_{\tilde{M}} = \left\{ u | u \in U \wedge \hbar_{\tilde{M}}(u) \in [\,0,\,1\,] \right\}, \tag{2}$$

т. е. подмножество $Z_{\tilde{M}}$ *универсального множества* U, *для элементов которого НХФД* $\hbar_{\tilde{M}}$ *принимает значения строго на отрезке* $[\,0,\,1\,]$.

Пусть наблюдаемые объекты характеризуются числом конечных разнотипных характеристик (признаков) $x_1, x_2, \ldots, x_n$ (в том числе и целевых признаков классов объектов), которые принимают значения из конечных нечетких множеств:

$$\begin{gathered} \tilde{X}^{(1)} = \left\{\tilde{\alpha}_1^{(1)}, \tilde{\alpha}_2^{(1)}, \ldots, \tilde{\alpha}_{p_1}^{(1)}\right\} \\ \tilde{X}^{(2)} = \left\{\tilde{\alpha}_1^{(2)}, \tilde{\alpha}_2^{(2)}, \ldots, \tilde{\alpha}_{p_2}^{(2)}\right\} \ldots \\ \ldots,\ \tilde{X}^{(n)} = \left\{\tilde{\alpha}_1^{(n)}, \tilde{\alpha}_2^{(n)}, \ldots, \tilde{\alpha}_{p_n}^{(n)}\right\}. \end{gathered} \tag{3}$$

Множествам значений признаков $\tilde{X}^{(j)}$ $\left(j = \overline{1,n}\right)$ поставим в соответствие одномерные числовые массивы $\tilde{d}^{(j)}$, называемые доменами. Таким образом, информация о том, что некоторый класс наблюдаемых объектов $\omega \in \Omega$ характеризуется n признаками со значениями из множеств $\tilde{X}^{(j)}$ векторов (3), представляется доменизированным числовым вектором. Последний назовем векторным квантом нечетких знаний 0-го уровня $\tilde{k}_0 y$:

$$\begin{aligned} \tilde{k}_0 y &= \left(\tilde{d}^1 : \tilde{d}^2 : \ldots : \tilde{d}^n : \tilde{d}^{n+1}\right) = \\ &= \left(\tilde{\alpha}_1^{(1)}, \ldots, \tilde{\alpha}_{p_1}^{(1)} : \tilde{\alpha}_1^{(2)}, \ldots, \tilde{\alpha}_{p_2}^{(2)} : \ldots : \tilde{\alpha}_1^{(n)}, \ldots, \tilde{\alpha}_{p_n}^{(n)} : \tilde{r}_0\right). \end{aligned} \tag{4}$$

Так же, как и при МАКЗ-методе, определим множество зарегистрированных наблюдений значений j-й характеристики x_j $\left(j=\overline{1,n}\right)$ как $\tilde{Y}^{(j)}=\left\{\tilde{\alpha}_k^{(j)}\right\}$. Применив НХФД $\hbar_{\tilde{Y}(j)}$ этого множества, получим характеристический квант нечетких знаний 1-го уровня (характеристи-ческий f-квант 1-го уровня) вида

$$\tilde{k}_1 f=\hbar_{\tilde{Y}(j)}\left(\tilde{\alpha}_k^{(j)}\right)=\mu\left(\tilde{\alpha}_k^{(j)}\right), \tag{5}$$

где $\mu\left(\alpha_k^{(j)}\right)$ — ПД наблюдаемого k-го признака j-й характеристики, принимающий значения в интервале $[\,0,\,1\,]$ и, таким образом, указывающий степень достоверности описываемого наблюдения. Этот ПД является аналогом используемой в теории нечетких подмножеств [2] характеристической функции принадлежности элемента $x\in X$ — $\mu(x)$. Смысл значения, которое может принимать ПД, состоит в следующем.

Интервал $[\,0,\,1\,]$ разбивается на ряд непересекающихся подынтервалов, каждый из которых выражает различную степень уверенности наблюдателя в достоверности описываемого наблюдения. Например, может быть образовано пять подынтервалов, описывающих достоверность наблюдения k-го признака j-й характеристики (таблица).

Интервал	Достоверность наблюдения признака
[0; 0,1 [	Не наблюдается
[0,1; 0,4 [	Возможно, не наблюдается
[0,4; 0,6]	Неизвестно, наблюдается или нет
] 0,6; 0,9 [	Возможно, наблюдается
[0,9; 1]	Точно наблюдается

Естественно, что при необходимости количество подынтервалов может быть увеличено.

Нечеткая характеристическая функция задается экспертом или инженером по знаниям; она может быть получена и в результате автоматической обработки сигналов датчиков.

В отличие от МАКЗ-метода, при котором аналогичный четкий квант имеет n доменов, при НАКЗ-методе f-квант имеет n+1 домен, содержащий только один элемент $\tilde{r}_0$. Последний назовем показателем достоверности f-кванта и определим как значение нечеткой функции достоверности $\tilde{h}(f)$.

Пример 1. Пусть имеется следующая информация, описывающая людей различных рас: рост — маленький (1 — 1,5 м), средний (1,5 — 2 м), большой (более 2 м); регион обитания — Африка, Азия, Южная Америка, Северная Америка, Европа, Австралия; цвет кожи — белый, желтый, красный, черный.

Эта информация может быть представлена в виде следующего f-кванта 0-го уровня, имеющего n = 3 домена (рост — $\tilde{X}_1$, регион обитания — $\tilde{X}_2$, цвет кожи — $\tilde{X}_3$), каждый из которых включает соответственно $p_1 = 3$, $p_2 = 6$, $p_3 = 4$ компонента:

$$\tilde{k}_0 y = \left[\tilde{\alpha}_1^{(1)}, \tilde{\alpha}_2^{(1)}, \tilde{\alpha}_3^{(1)} : \tilde{\alpha}_1^{(2)}, \tilde{\alpha}_2^{(2)}, \tilde{\alpha}_3^{(2)}, \tilde{\alpha}_4^{(2)}, \tilde{\alpha}_5^{(2)}, \tilde{\alpha}_6^{(2)} : \right.$$
$$\left. : \tilde{\alpha}_1^{(3)}, \tilde{\alpha}_2^{(3)}, \tilde{\alpha}_3^{(3)}, \tilde{\alpha}_4^{(3)} : \tilde{r}_0 \right]. \qquad (6)$$

Тогда, например, факт "в Южной Америке зарегистрированы люди, рост которых не превышает 1,5 м, с красным цветом кожи" со значением достоверности 0,72 может быть выражен следующим f-квантом с $\tilde{r}_0$ =0,72:

$$\tilde{k}_0 y = \left[\overbrace{0{,}95;\ 0{,}0;\ 0{,}0}^{\tilde{X}_1} : \overbrace{0{,}0;\ 0{,}8;\ 0{,}0;\ 0{,}0;\ 0{,}0;\ 0{,}0}^{\tilde{X}_2} : \right.$$
$$\left. : \overbrace{0{,}0;\ 0{,}0;\ 0{,}85;\ 0{,}0}^{\tilde{X}_3} : \overbrace{0{,}72}^{\tilde{r}_0} \right] \qquad (7)$$

Аналогично МАКЗ-методу при НАКЗ-методе определяется выбирающий квант нечетких знаний 0-го уровня :

$$\tilde{k}_0 \alpha = V_k^{(p)}\left(\tilde{\alpha}_1, \tilde{\alpha}_2, \ldots, \tilde{\alpha}_p\right) = \tilde{\alpha}_k . \qquad (8)$$

Он позволяет установить, что, например, из p-мерной совокупности наблюдаемых значений характеристик выбрано (зарегистрировано) значение $\tilde{\alpha}_k$.

Таким образом, определены исходные алгоритмические конструкции, которые будут использоваться для построения нечетких квантов знаний различных уровней с помощью описанных в исходном МАКЗ-методе операторов строчной и матричной конкатенации $con\langle\circ\rangle$ и $con[\circ]$, а также оператора суперпозиции [1] (в МАКЗ-методе — П-оператор).

Далее описано алгоритмическое определение и конструирование многоуровневых нечетких $\tilde{k}$-знаний. Определим кванты нечетких знаний по аналогии с МАКЗ-методом следующим образом.

Определение 3. *Квантами нечетких $\tilde{k}$-знаний 0-го уровня (f-квантами 0-го уровня) называются алгоритмические структуры, получаемые из терминальных квантов $\tilde{k}_0 y$ (3) и $\tilde{k}_0 \alpha$ (4) путем конечного числа применений* П*-оператора.*

Определение 4. *Квантами нечетких $\tilde{k}$-знаний 1-го уровня (*f*-квантами 1-го уровня) называются алгоритмические структуры, получаемые из терминальных квантов $\tilde{k}_0 y$ (3) и $\tilde{k}_0 \alpha$ (4) путем конечного числа применений П-оператора и $con\langle\circ\rangle$-оператора.*

Пример 2. По определению 3 можно получить структуру, например, следующего вида:

$$\begin{aligned}\tilde{k}_1 d_э = \tilde{d}_э^{(1)} = con \mathop{<}_{k=1}^{p_1=3} \hbar_{\tilde{Y}(1)} \left(V_k^{p_1=3}\left(V_k^{n=2}\left(\tilde{k}_0 y\right)\right)\right) >= \\ = con \mathop{<}_{k=1}^{p_1=3} \tilde{k}_1 f_k^{(1)} >= [\,0{,}0; 0{,}95; 0{,}0:\,],\end{aligned} \tag{9}$$

которая представляет собой элементный доменный нечеткий квант 1-го уровня. Этим квантом описывается следующая информация: "в первом домене зафиксировано только одно значение — второе значение ($\tilde{\alpha}_2^{(1)}$) первой характеристики наблюдаемого объекта с показателем достоверности 0,95".

Если в таком кванте содержатся все значения компонент характеристики, имеющие значение больше нуля, то, как и при использовании

МАКЗ-метода, это свидетельствует о неполноте информации о характеристиках объекта. Такой квант называется интервальным доменным f-квантом 1-го уровня.

Пример 3. По определению 4 для f-квантов $\tilde{k}_0 y$ и $\tilde{k}_0 d_э$ можно получить структуру следующего вида:

$$\begin{aligned}\tilde{k}_1 Y_э &= con \underset{j=1}{\overset{n=2}{<}} \tilde{k}_1 d_э^{(j)} >= \\ &= con \underset{j=1}{\overset{n=2}{<}} con \underset{k=1}{\overset{p_j}{<}} \hbar_{\tilde{\gamma}(j)} (V_k^{(p_j)} \left(V_j^{(n)} \left(\tilde{k}_0 y\right)\right)) >= \\ &= \left[\ 0{,}0;\ 0{,}95;\ 0{,}0 : 0{,}0;\ 0{,}79 : 0{,}71\ \right].\end{aligned} \tag{10}$$

Эта структура называется элементным f-квантом 1-го уровня и отражает следующую информацию: "для наблюдаемого объекта зафиксированы конкретные значения всех известных характеристик $x_1 = \tilde{\alpha}_2^{(1)}$, $x_2 = \tilde{\alpha}_2^{(2)}$, и показатель достоверности данного наблюдения равен 0,71".

Пример 4. Пусть по условиям примера 3 аналогично построен интервальный f-квант 1-го уровня:

$$\tilde{k}_1 Y = \left[\ 0{,}0;\ 0{,}95;\ 0{,}0 : 0{,}83;\ 0{,}79 : 0{,}98 : 0{,}71\right]. \tag{11}$$

Он имеет следующий смысл: "у наблюдаемого объекта в данный момент характеристика x_1 определена частично, а характеристика x_2 полностью не определена, и показатель достоверности данного наблюдения равен 0,71".

Определение 5. *Квантами нечетких $\tilde{k}$ -знаний 2-го уровня (f-квантами 2-го уровня) называются алгоритмические структуры, получаемые из терминальных квантов $\tilde{k}_0 y$ (3) и $\tilde{k}_0 \alpha$ (4) путем конечного числа применений П-оператора и $con[\circ]$-оператора.*

Таким образом, матричный f-квант 2-го уровня представляется структурой следующего вида:

$$\tilde{k}_2\|Y\| = con\,\underset{i=1}{\overset{m}{[}}\,\tilde{k}_0 y] = \begin{bmatrix} \tilde{k}_0 y_1 \\ \tilde{k}_0 y_2 \\ \dots \\ \tilde{k}_0 y_m \end{bmatrix}. \qquad (12)$$

По определению 5 аналогично строятся: матрично-элементный f-квант 2-го уровня вида

$$\tilde{k}_2\|Y_э\| = con\,\underset{i=1}{\overset{n}{[}}\,\tilde{k}_0 Y_{эi}] = \begin{bmatrix} \tilde{k}_0 y_{э1} \\ \tilde{k}_0 y_{э2} \\ \dots \\ \tilde{k}_0 y_{эn} \end{bmatrix} \qquad (13)$$

и матрично-интервальный f-квант 2-го уровня вида

$$\tilde{k}_2\|Y\| = con\,\underset{i=1}{\overset{s}{[}}\,\tilde{k}_0 Y_l] = \begin{bmatrix} \tilde{k}_0 y_1 \\ \tilde{k}_0 y_2 \\ \dots \\ \tilde{k}_0 y_s \end{bmatrix}. \qquad (14)$$

Представление нечетких k-знаний в виде предикатов

Остается справедливым данное для МАКЗ-метода определение интервала пространства моделей объектов принятия решений как декартова произведения подмножеств $\chi \subseteq \tilde{X}^{(j)}$, $j = \overline{1,n}$, выбранных по одному из множеств $\tilde{X}^{(1)},\dots,\tilde{X}^{(n)}$. Поэтому так же, как и для МАКЗ-метода, произвольный интервальный квант $\tilde{k}_1 Y$ алгебраически можно описать конечным предикатом в форме элементарной конъюнкции (ЭК) либо элементарной дизъюнкции (ЭД) выражений

$$x_j \in \tilde{\chi}_j \subseteq \tilde{X}^{(j)}, \; j = \overline{1,n}. \qquad (15)$$

И так же, как и при использовании МАКЗ-метода, указанный предикат принимает значение 1 только на элементах соответствующего ему интервала пространства моделей объектов принятия решений.

Аналогично определим предикат, заданный в форме ЭК, как конъюнкт $\tilde{k}_1^{\&} Y$, а в форме ЭД— как дизъюнкт $\tilde{k}_1^{V} Y$.

Пример 5. Если нечеткий векторный квант знаний

$$\tilde{k}_1^{\&} Y = [0,6;\ 0,0;\ 0,0;\ 0,8 :\ 0,95;\ 0,0;\ 0,0;\ 0,74 :\ 0,54] \qquad (16)$$

представляет собой конъюнкт, то ему соответствует ЭК вида

$$\left(\left(x_1 = \tilde{\alpha}_1^{(1)}\right) \vee \left(x_1 = \tilde{\alpha}_4^{(1)}\right)\right) \wedge \left(\left(x_2 = \tilde{\alpha}_1^{(2)}\right) \vee \left(x_2 = \tilde{\alpha}_3^{(2)}\right)\right) = \\ = 1 | \tilde{r}_0 \qquad (17)$$

или, после упрощений,

$$\tilde{k}_1^{\&} Y = (x_1 = 1 | 0,6; 4 | 0,8)(x_2 = 1 | 0,95; 3 | 0,74) = 1 | 0,54. \qquad (18)$$

Таким образом, k-знания, описывающие различные факты, могут быть выражены в виде доменизированных векторов или предикатных уравнений.

Функция достоверности фактов и закономерностей

МАКЗ-методом предусматривается, что f-кванты, как и четкие кванты, могут описывать и функциональные, и импликативные закономерности.

Пример 6. Пусть имеется конъюнктный f-квант вида

$$\tilde{k}_1^{\&} Y = [0,0; 0,84, 0,0 : 0,76; 0,98; 0,0 : 0,0; 0,9 : 0,68] = \\ = 1 | 0,68. \qquad (19)$$

Этим квантом можно описать функциональную закономерность

$$\tilde{k}_1^{\&} Z = (x_1 = 2 | 0,84)(x_2 = 1 | 0,76; 2 | 0,98)(x_3 = 2 | 0,9) = \\ = 1 | 0,68, \qquad (20)$$

означающую: "рассматриваемый объект должен обладать 2-м значением 1-й характеристики с ПД не менее 0,84, 1-м значением 2-й характеристики с ПД не менее 0,76 или 2-м значением 2-й характеристики с ПД не менее 0,98 и 2-м значением 3-й характеристики с ПД не менее 0,9 и ПД этого наблюдения 0,68".

Предикат (19) может быть представлен в форме следующих импликаций:

$$\tilde{k}_1^{\&} Z_1 = (x_1 = 2 \mid 0{,}84)(x_2 = 1 \mid 0{,}76, 2 \mid 0{,}98) = \neg(x_3 = 2 \mid 0{,}9) = \\ = 1 \mid 0{,}68; \tag{21}$$

$$\tilde{k}_1^{\&} Z_2 = \left(x_1 = 2|0{,}84\right)\left(x_2 = 1|0{,}76,\ 2|0{,}98\right)\left(x_3 = 2|0{,}9\right) = \\ = 1|0{,}68 \tag{22}$$

и т.д.

ПД подобной импликации будем определять по формуле [3]

ПД заключения = ПД посылки * ПД импликации. (23)

ПД заключения в данном случае является ПД *f*-кванта, описывающего факт либо закономерность. Это значение ПД принимает единственный элемент $\tilde{r}_0$ n+1-го домена *f*-кванта.

ПД посылки определяется на основе ПД элементов доменов исходя из следующих определений.

Определение 6. *Дизъюнкция двух элементов домена кванта представляет собой результат следующей операции над ПД этих элементов* :

$$\tilde{\alpha}_r^{(j)} \vee \tilde{\alpha}_s^{(j)} = MAX\left(\mu\left(\tilde{\alpha}_r^{(j)}\right), \mu\left(\tilde{\alpha}_s^{(j)}\right)\right), \tag{24}$$

$$j = \overline{1,n}\,,\ r = \overline{1,p_j}\,,\ s = \overline{1,p_j}\,,\ r \neq s\,,$$

где p — число элементов домена.

Эту операцию можно применить и к доменам *f*-кванта.

Определение 7. Конъюнкция двух элементов домена f-кванта представляет собой результат следующей операции над ПД этих элементов:

$$\tilde{\alpha}_r^{(j)} \vee \tilde{\alpha}_s^{(j)} = MIN\left(\mu\left(\tilde{\alpha}_r^{(j)}\right), \mu\tilde{\alpha}_s^{(j)}\right), \tag{25}$$

$$j = \overline{1,n}\,,\ r = \overline{1,p_j}\,,\ s = \overline{1,p_j}\,,\ r \neq s\,,$$

где p — число элементов домена.

Эту операцию можно применить и к доменам *f*-кванта.

Определение 8. *ПД j-го домена f-кванта определяется выражением*

$$\mu\left(\tilde{d}^{(j)}\right)=\tilde{\alpha}_1^{(j)}\vee\tilde{\alpha}_2^{(j)}\vee\ \ldots\ \vee\tilde{\alpha}_{p_i}^{(j)},\ j=\overline{1,n},\ i=\overline{1,n}. \tag{26}$$

Определение 9. *ПД f-кванта, представленного в виде конъюнкта $\tilde{k}_1^{\&}Y$, определяется выражением вида*

$$\mu\left(\tilde{k}_1^{\&}Y\right)=\mu\left(\tilde{d}^{(1)}\right)\wedge\mu\left(\tilde{d}^{(2)}\right)\wedge\ \ldots\ \wedge\mu\left(\tilde{d}^{(n)}\right)=\tilde{r}_0, \tag{27}$$

где n — число доменов f-кванта.

Определение 10. *ПД f-кванта, представленного в виде дизъюнкта $\tilde{k}_1^{V}Y$, определяется выражением вида*

$$\mu\left(\tilde{k}_1^{V}Y\right)=\mu\left(\tilde{d}^{(1)}\right)\vee\mu\left(\tilde{d}^{(2)}\right)\vee\ \ldots\ \vee\mu\left(\tilde{d}^{(n)}\right)=\tilde{r}_0, \tag{28}$$

где n — число доменов f-кванта.

Определение 11. *Функция достоверности факта или закономерности, описываемой f-квантом в виде конъюнкта $\tilde{k}_1^{\&}Y$, представляет собой выражение вида*

$$\tilde{h}(f)=ft\left(\tilde{k}_1^{\&}Y\right)=\mu\left(\tilde{k}_1^{\&}Y\right)*\mu(\to), \tag{29}$$

а факта или закономерности, описываемой f-квантом в виде дизъюнкта $\tilde{k}_1^{V}Y$, представляет собой выражение вида

$$\tilde{h}(f)=ft\left(\tilde{k}_1^{V}Y\right)=\mu\left(\tilde{k}_1^{V}Y\right)*\mu(\to), \tag{30}$$

где $\mu(\to)$ — ПД импликации, равный 0,9 [3].

Таким образом, решена первая часть поставленной задачи — определена стандартная форма представления нечеткой информации при использовании НАКЗ-метода. Кроме того, определены многоуровневые нечеткие $\tilde{k}$-знания, которые представляют собой алгоритмические конструкции, получаемые из исходных квантов $\tilde{k}_0 y$ (3), $\tilde{k}_0\alpha$ (4) и $\tilde{k}_0 f$ путем конечного числа применений П-оператора, $con\langle\circ\rangle$-оператора и $con[\circ]$-оператора. Определена также функция достоверности факта (закономерности), позволяющая описать степень уверенности наблюдателя в достоверности наблюдаемого факта (закономерности).

В работе [4] показан механизм индуктивного вывода закономерностей из f-квантов, описывающих факты и закономерности, построения БНЗ, а также механизм дедуктивного вывода нечетких закономерностей из имеющейся БНЗ.

Список литературы: 1. *Сироджа И.Б.* Математическое и программное обеспечение интеллектуальных компьютерных систем. Х.: Харьк. авиац. ин-т, 1992. 101 с. 2. *Кофман А.* Введение в теорию нечетких множеств: Пер. с фр. М.: Радио и связь, 1982. 432 с. 3. *Марселус Д.* Программирование экспертных систем на ТУРБО ПРОЛОГе: Пер. с англ. М.: Финансы и статистика, 1994. 256 с. 4. *Гацкий М.Ю.* Операторы индуктивного и дедуктивного вывода $\tilde{k}$ -знаний в классе моделей нечетких алгоритмических квантов знаний (НАКЗ-моделей). – См. статью в настоящем сборнике.

Поступила в редколлегию 23.01.98

Проблемы бионики
1998. Вып. 48. С. 162—167

ХТУРЭ

УДК 681.324

М.Ю. ГАЦКИЙ

ОПЕРАТОРЫ ИНДУКТИВНОГО И ДЕДУКТИВНОГО ВЫВОДА $\tilde{k}$-ЗНАНИЙ В КЛАССЕ МОДЕЛЕЙ НЕЧЕТКИХ АЛГОРИТМИЧЕСКИХ КВАНТОВ ЗНАНИЙ (НАКЗ-МОДЕЛЕЙ)

В работе [1] определена стандартная форма представления нечеткой информации при использовании метода нечетких алгоритмических квантов знаний (НАКЗ-метода). Следующим этапом является: разработка механизма индуктивного вывода нечетких $\tilde{k}$ -знаний, создание базы нечетких знаний (БНЗ) и разработка механизма дедуктивного вывода нечетких знаний из БНЗ.

Основные определения

При использовании НАКЗ-метода, как и МАКЗ-метода [2], интервальный f-квант может быть представлен в виде предиката, заданного в форме элементарной конъюнкции (конъюнкта) $\tilde{k}_1^{\&} Y$ [1]. Значения элементов доменов, при которых этот предикат является истинным, назовем характеристическим множеством данного предиката.

Определение 1. *Пусть A — множество элементов доменов* $\tilde{\alpha}$, $\tilde{A}$ *и* $\tilde{B}$ *— два нечетких подмножества A. Будем считать, что* $\tilde{B}$ *содержится в* $\tilde{A}$ [3], *если*

$$\forall \tilde{\alpha} \in A : \mu_A(\tilde{\alpha}) \leq \mu_B(\tilde{\alpha}), \tag{1}$$

где $\mu_A(\tilde{\alpha})$ *— показатель достоверности (ПД)* [1] *элемента нечеткого подмножества A;* $\mu_B(\tilde{\alpha})$ *— ПД элемента нечеткого подмножества B. Будем этот факт обозначать как*

$$\tilde{B} \subset \tilde{A}. \tag{2}$$

Определение 2. *Из f-кванта* $\tilde{k}_1^{\&} A$ *логически следует f-квант* $\tilde{k}_1^{\&} B$,

$$\tilde{k}_1^{\&} A \rightarrow \tilde{k}_1^{\&} B, \tag{3}$$

тогда и только тогда, когда характеристическое множество f-кванта $\tilde{k}_1^{\&} B$ является нечетким подмножеством одного из характеристических множеств f-кванта $\tilde{k}_1^{\&} A$. Будем этот факт выражать так:

$$E_f\left(\tilde{B}\right) \subseteq E_f\left(\tilde{A}\right). \tag{4}$$

Аналогично,

$$\tilde{k}_2^{\&}\|S\| \to \tilde{k}_1^{\&}C, \quad \text{если} \quad E_f\left(\tilde{C}\right) \subseteq E_f\left(\|\tilde{S}\|\right); \tag{5}$$

$$\tilde{k}_2^{\&}\|S\| \to \tilde{k}_2^{\&}\|P\|, \quad \text{если} \quad E_f\left(\|\tilde{P}\|\right) \subseteq E_f\left(\|\tilde{S}\|\right), \tag{6}$$

где $\tilde{k}_1^{\&} A$, $\tilde{k}_1^{\&} B$ — некоторые конъюнкты; $\tilde{k}_2^{\&}\|S\|$, $\tilde{k}_2^{\&}\|P\|$ — матрицы конъюнктов, представленные в виде предикатных дизъюнктивных нормальных форм [2]; $E_f\left(\tilde{A}\right)$, $E_f\left(\tilde{B}\right)$, $E_f\|\tilde{S}\|$, $E_f\|\tilde{P}\|$ *— их характеристические множества.*

Пример. Пусть имеются f-кванты вида

$$\begin{aligned} \tilde{k}_1^{\&} A &= [0{,}6;\ 0{,}7;\ 0{,}0 : 0{,}0;\ 0{,}75;\ 0{,}0 : 0{,}0;\ 0{,}8 : 0{,}54]; \\ \tilde{k}_1^{\&} B &= [0{,}8;\ 0{,}0;\ 0{,}0 : 0{,}0;\ 0{,}9;\ 0{,}0 : 0{,}0;\ 0{,}95 : 0{,}72]. \end{aligned} \tag{7}$$

Тогда на основании определения 2 можно утверждать, что

$$\tilde{k}_1^{\&} A \to \tilde{k}_1^{\&} B, \tag{8}$$

так как

$$\begin{aligned} E_f\left(\tilde{A}\right) = \{ & [0{,}6; 0{,}0; 0{,}0 : 0{,}0; 0{,}75; 0{,}0 : 0{,}0; 0{,}8], \\ & [0{,}0; 0{,}7; 0{,}0 : 0{,}0; 0{,}75; 0{,}0 : 0{,}0; 0{,}8] \}; \end{aligned} \tag{9}$$

$$E_f\left(\tilde{B}\right) = [\,0{,}8; 0{,}0; 0{,}0 : 0{,}0; 0{,}9; 0{,}0 : 0{,}0; 0{,}95\,]$$

и, следовательно,

$$E_f\left(\tilde{B}\right) \subseteq E_f\left(\tilde{A}\right).$$

О п р е д е л е н и е 3. *Назовем* $\tilde{0}$ *-квантом f-квант, имеющий значение показателя достоверности (ПД)*[1], *равное 0 во всех компонентах, и будем его обозначать как* $0\tilde{k}_1^{\&}$.

О п р е д е л е н и е 4. *Назовем* $\tilde{\delta}$ *-квантом f-квант, содержащий хотя бы один домен, в котором все элементы имеют значение ПД, равное 0, и будем его обозначать как* $\delta\tilde{k}_1^{\&}$.

О п р е д е л е н и е 5. *Назовем f-кванты* $\tilde{k}_1^{\&}A$ *и* $\tilde{k}_1^{\&}B$ *условно смежными по j-й характеристике* $\left(j=\overline{1,n}\right)$, *если выполняются такие условия:*

$$\begin{aligned} &\tilde{d}_A^{(j)} \vee \tilde{d}_B^{(j)} \neq \tilde{d}_A^{(j)} \neq \tilde{d}_B^{(j)} \neq \varnothing; \\ &\forall i=\overline{1,n} \quad \tilde{d}_A^{(i)} \neq \varnothing, \quad \tilde{d}_B^{(i)} \neq \varnothing, \quad i \neq j. \end{aligned} \tag{10}$$

О п р е д е л е н и е 6. *Условной дизъюнкцией f-квантов* $\tilde{k}_1^{\&}A$ *и* $\tilde{k}_1^{\&}B$ *по характеристике* $\tilde{X}^r$ $\left(r=\overline{1,n}\right)$ *называется f-квант*

$$\tilde{k}_1^{\&}R = \tilde{k}_1^{\&}A \vee_{\tilde{X}^r} \tilde{k}_1^{\&}B, \tag{11}$$

доменные строки которого определяются по формуле

$$\forall j=\overline{1,n}, \quad \tilde{d}_R^{(j)} = \begin{cases} \tilde{d}_A^{(j)} \vee \tilde{d}_B^{(j)}, & \text{если } j=r; \\ \tilde{d}_A^{(j)} \wedge \tilde{d}_B^{(j)}, & \text{если } j\neq r. \end{cases} \tag{12}$$

Оператор индуктивного вывода $\tilde{k}$ -знаний

При использовании НАКЗ-метода сохраняется предусмотренное МАКЗ-методом определение функциональной закономерности как связи между некоторыми величинами, позволяющей «по значениям одних величин, называемых аргументами, однозначно определить значения других — функций» [2].

На основании определений 1—6 разработан алгоритм F1 построения БНЗ из имеющейся базы фактов.

Алгоритм F1

Вход: $\tilde{\Sigma}_0 = \tilde{k}_2^{\&}\|x_0\|$ — база нечетких фактов, представленная в виде матрицы $\tilde{k}$ -знаний размером $m \times n$, где m — число строк матрицы, n — число доменов (характеристик объекта).

Выход: $\tilde{\Sigma}_{ff} = \tilde{k}_2^{\&}\|Y_f\|$ — база функциональных $\tilde{k}$ -знаний как модель исследуемого мира.

Действия:

1. Преобразовать матрицу $\tilde{\Sigma}_0$ в матрицу $\tilde{\Sigma}_1$ путем попарного выполнения операции условной дизъюнкции над строками матрицы $\tilde{\Sigma}_0$.

2. Добавить из матрицы $\tilde{\Sigma}_0$ в матрицу $\tilde{\Sigma}_1$ все строки, представленные интервальными f-квантами 1-го уровня.

3. Получить матрицу $\tilde{\Sigma}_2$, удалив из матрицы $\tilde{\Sigma}_1$ все строки, соответствующие $\tilde{0}$ - или $\tilde{\delta}$ -квантам.

4. Получить матрицу $\tilde{\Sigma}_3$, удалив из матрицы $\tilde{\Sigma}_2$ все столбцы, имеющие во всех строках значение ПД, большее или равное 0.

5. Получить матрицу $\tilde{\Sigma}_4$, удалив из матрицы $\tilde{\Sigma}_3$ все полностью одноэлементные домены. Полученную матрицу $\tilde{\Sigma}_4$ считать базой нечетких функциональных $\tilde{k}$ -знаний (БНЗ) и обозначать как $\tilde{\Sigma}_{ff} = \tilde{k}_2^{\&}\|Y_f\|$.

Определение 7. *Алгоритмическая процедура*

$$INDFF\left(\tilde{\Sigma}_0;\ F1;\ \tilde{\Sigma}_{ff}\right) = \tilde{k}_2^{\&}\|x_0\| \underset{F1}{\overset{INDFF}{\Rightarrow}} \tilde{k}_2^{\&}\|Y_f\|, \tag{13}$$

реализующая формирование f-кванта $\tilde{k}_2^{\&}\|Y_f\|$ базовых функциональных нечетких закономерностей с помощью алгоритма F1, называется оператором индуктивного вывода функциональных $\tilde{k}$ -знаний (INDFF-оператором).

Оператор дедуктивного вывода $\tilde{k}$ -знаний

На основании определений 1—6 разработан алгоритм F2 дедуктивного вывода нечетких закономерностей из БНЗ.

Алгоритм F2

Вход: $\tilde{k}$ -знания $\tilde{\Sigma}_{ff}$ и $\tilde{k}_1^{\&} Y_{\omega}$.

Выход: $\tilde{k}$ -знания $\tilde{k}_2^{\&}\|Y_\omega^*\|$ о возможном состоянии объекта ω, судя по наблюдениям $\tilde{k}_1^{\&}Y_\omega$.

Действия:

1. Преобразовать f-квант $\tilde{k}_1^{\&}Y_\omega$ в f-квант $\tilde{k}_1^{\&}F$, заменив значения ПД в параметрах характеристик, требующих уточнения, на значение ПД, равное 0,5.

2. Из матрицы $\tilde{\Sigma}_{ff}$ получить матрицу $\tilde{\Sigma}_1$ путем выполнения операции условной дизъюнкции над f-квантом $\tilde{k}_1^{\&}F$ и каждой строкой матрицы $\tilde{\Sigma}_{ff}$.

3. Получить матрицу $\tilde{\Sigma}_2$, удалив из матрицы $\tilde{\Sigma}_1$ все строки, соответствующие $\tilde{0}$- или $\tilde{\delta}$-квантам.

4. Получить матрицу $\tilde{\Sigma}_3$, удалив из матрицы $\tilde{\Sigma}_2$ все столбцы, имеющие во всех строках значение ПД, большее или равное 0.

5. Получить матрицу $\tilde{\Sigma}_4$, удалив из матрицы $\tilde{\Sigma}_3$ все полностью одноэлементные домены. Полученную матрицу $\tilde{\Sigma}_4$ считать $\tilde{k}$-знаниями о возможном состоянии объекта ω, судя по наблюдениям $\tilde{k}_1^{\&}Y_\omega$, и обозначать как $\tilde{\Sigma}_\omega^* = \tilde{k}_2^{\&}\|Y_\omega^*\|$.

О п р е д е л е н и е 8. *Алгоритмическая процедура*

$$DEDF\left(\tilde{\Sigma}_{ff},\, \tilde{k}_1^{\&}Y_\omega;\; F2;\; \tilde{\Sigma}_\omega^*\right) = \tilde{k}_2^{\&}\|Y_f\| \overset{DEDF}{\underset{F2}{\Rightarrow}} \tilde{k}_2^{\&}\|Y_\omega^*\|, \tag{14}$$

реализующая получение f-кванта $\tilde{k}_2^{\&}\|Y_\omega^*\|$, который описывает возможное состояние объекта ω исходя из базовых ($\tilde{\Sigma}_{ff}$) и наблюдаемых ($\tilde{k}_1^{\&}Y_\omega$) $\tilde{k}$-знаний с помощью алгоритма F2, называется оператором дедуктивного вывода нечетких $\tilde{k}$-знаний (DEDF-оператором).

Таким образом, определен механизм индуктивного вывода нечетких $\tilde{k}$-знаний о рассматриваемых объектах в виде алгоритма F1 и построения БНЗ. Кроме того, описан механизм дедуктивного вывода нечетких $\tilde{k}$-знаний из БНЗ.

Тем самым решена вторая и третья части задачи, поставленной в [1], и, следовательно, задача адаптации МАКЗ-метода к принятию решений в условиях неопределенности.

В Харьковском авиационном университете разрабатывается прототип программы, выполняющей индуктивный и дедуктивный выводы нечетких $\tilde{k}$ -знаний.

Список литературы: 1. *Гацкий М.Ю.* Формализация нечетких знаний на основе моделей нечетких алгоритмических квантов знаний (НАКЗ-моделей) для компьютерного принятия решений в условиях неопределенности. — См. статью в настоящем сборнике. 2. *Сироджа И.Б.* Математическое и программное обеспечение интеллектуальных компьютерных систем. Х.: Харьк. авиац. ин-т, 1992. 101 с. 3 *Кофман А.* Введение в теорию нечетких множеств: Пер. с фр. М.: Радио и связь, 1982. 432 с.

Поступила в редколлегию 23.01.98

Проблемы бионики
1998. Вып. 49. С. 168—174

ХТУРЭ

УДК 658.012:007

А.П. РОТШТЕЙН, Ю.И. МИТЮШКИН

ИДЕНТИФИКАЦИЯ НЕЛИНЕЙНЫХ ЗАВИСИМОСТЕЙ НЕЙРОННЫМИ СЕТЯМИ

Использование нейронных сетей [1], являющихся обобщением идеи персептрона Розенблатта и моделирующих работу мозга, в последнее время считается одной из перспективных технологий исследований в области искусственного интеллекта [2]. С формальной точки зрения нейронная сеть представляет собой универсальную модель-аппроксиматор в виде текущего графа, которая приобретает адекватность реальному объекту вследстие модификации весов дуг. Определение необходимых весов дуг осуществляется в процессе "обучения" нейронной сети [3; 4].

Большинство известных результатов применения нейронных сетей касается проблемы распознавания образов [5]. Задача распознавания образов тождественна задаче идентификаци нелинейных объектов [6], которую часто приходится решать не только при разработке теории управления, но и при обработке данных в экономике, медицине, социологии и т. д. Возникает вопрос: а возможно ли применить нейронные сети для идентификации нелинейных объектов? Поиску ответа на этот вопрос и посвящена данная работа.

Нейронная сеть как универсальный аппроксиматор. Покажем, что с помощью нейронной сети можно формально описать нелинейный объект вида

$$y = f(x_1, x_2, ..., x_n), \tag{1}$$

где y — выход; $x_1, x_2, ..., x_n$ — входы.

Формальная модель нейронной сети представляет собой совокупность формальных нейронов, определенным образом соединенных между собой и внешней средой [1; 2]. Модель формального нейрона Розенблатта изображена на рис. 1.

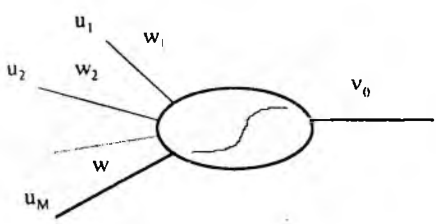


Рис. 1

Нейрон в простейшем случае представляет собой элемент с несколькими входами и одним выходом, выполняющий нелинейное параметрическое преобразование суммарного взвешенного входного сигнала в скалярную величину:

$$v_0 = \psi\left(\sum_{i=1}^{M} u_i w_i\right). \tag{2}$$

Здесь v_0 — выходной сигнал; $\psi(\bullet)$ — характеристическая функция (функция активации) нейрона; M — количество входов; u_i — входной i-й сигнал; w_i — вес i-го входа.

В качестве $\psi(\bullet)$, как правило, используется сигмоидная функция вида

$$\psi(\tau) = \frac{1}{1+\exp(-k\,\tau)}, \tag{3}$$

где k — коэффициент сжатия-растяжения функции вдоль оси абсцисс.

Особенность данной функции заключается в том, что ее производную можно представить в аналитическом виде:

$$\psi'(\tau) = \psi(\tau)\cdot(1-\psi(\tau)). \tag{4}$$

Для моделирования нелинейных объектов можно применить нейронную сеть типа многослойного персептрона [2—4]. Модель многослойного персептрона с тремя слоями представлена на рис. 2. Первый слой нейронов служит для ввода, последний — для вывода, а внутренний — для хранения информации. Каждый нейрон может соединяться с любым нейроном из соседнего слоя, однако между нейронами одного слоя связи отсутствуют. Все нейроны могут посылать сигналы только в следующие слои и принимать сигналы только из предыдущих слоев. Для организации взаимодействия нейронной сети с внешней средой служат блоки кодирования (БК) и декодирования (БДК) информации. В рассматриваемом случае эти блоки выполняют преобразование чисел соответственно из десятичной системы счисления в двоичную и наоборот.

Веса межнейронных связей представляются в виде матриц:

$$W^k = \{w_{ij}^k\},$$

$$k = \overline{1,\ k},\quad i = \overline{1, M_{вх}},\quad j = \overline{1, M_{вых}}, \tag{5}$$

где k— количество матриц межнейронных связей; w_{ij}^k — вес связи между j-м нейроном выходного слоя и i-м нейроном входного слоя; $M_{вх}$, $M_{вых}$— количества нейронов входного и выходного слоев.

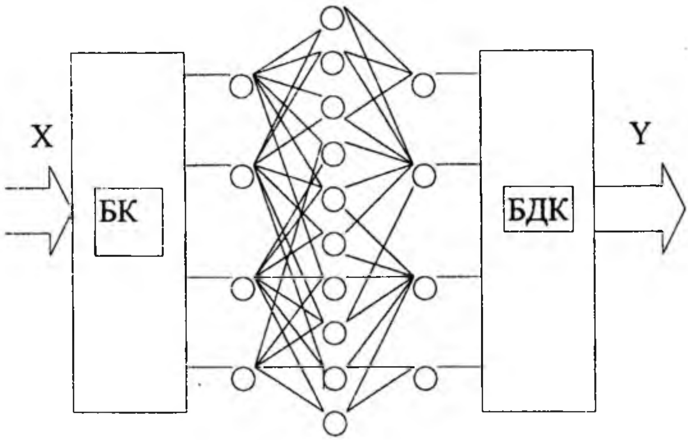


Рис. 2

Многослойный персептрон функционирует следующим образом Вектор входных данных $X=\{x_1, x_2, \ldots, x_n\}$ кодируется и подается на входной слой нейронной сети. После этого осуществляется поочередное вычисление выходных сигналов для нейронов каждого следующего слоя с использованием уже известных выходов нейронов предыдущего слоя и значений весов межнейронных связей. В итоге информация, полученная из выходного слоя сети, декодируется и выдается как выход нелинейного объекта:

$$Y = F(W^1, W^2, \ldots, W^3, x_1, x_2, \ldots, x_n). \tag{6}$$

Задача идентификации нелинейных объектов нейронными сетями может быть представлена в таком виде. Дано: (X^p, a^p), $p=\overline{1, P}$ — обучающая выборка в виде P пар вход-выход, где $X^p=\{x_1^p, x_2^p, \ldots, x_n^p\}$ — входной вектор; a^p — выходной параметр.

Необходимо определить такие параметры нейронной сети (5), при которых обеспечивается минимальное отклонение модели от объекта:

$$\sum_{p=1}^{P}(F(W^1, W^2, \ldots, W^k, x_1^p, x_2^p, \ldots, x_n^p) - a^p)^2 \to \min_{W^k}. \tag{7}$$

Обучение нейронной сети. Для поиска неизвестных параметров, обеспечивающих адекватность нейронной сети данным эксперимента, был использован метод обратного распространения ошибки (back-propagation method) [3; 4]. Каждая итерация процедуры обучения состоит из двух этапов — прямого и обратного хода. Приведем соответствующие алгоритмы.

Алгоритм прямого хода:

1. Определить суммарный взвешенный входной сигнал u_j каждого нейрона текущего слоя:

$$u_j = \sum_i \nu_i \, w_{ij}. \tag{8}$$

2. Определить выходной сигнал ν_j каждого нейрона текущего слоя:

$$\nu_j = \frac{1}{1 + e^{-k u_j}}. \tag{9}$$

3. Если текущий слой — не выходной, то перейти к следующему слою и повторить процедуры начиная с шага 1.

4. Вычислить погрешность Q нейронной сети:

$$Q = 0{,}5 \sum_j (\nu_j - \nu_j^{\circ})^2. \tag{10}$$

Алгоритм обратного хода:

1. Вычислить скорость EA изменения погрешности при изменении выходного сигнала для каждого нейрона выходного слоя:

$$EA_j = \frac{\partial Q}{\partial \nu_j} = (\nu_j^{\circ} - \nu_j). \tag{11}$$

2. Вычислить скорость EI изменения погрешности при изменении суммарного входного сигнала каждого нейрона текущего слоя:

$$EI_j = \frac{\partial Q}{\partial u_j} = \frac{\partial Q}{\partial \nu_j} \cdot \frac{\partial \nu_j}{\partial u_j} = EA_j \, \nu_j \, (1 - \nu_j). \tag{12}$$

3. Вычислить скорость EW изменения погрешности при изменении веса на входной связи каждого нейрона текущего слоя:

$$EW_{ij} = \frac{\partial Q}{\partial w_{ij}} = \frac{\partial Q}{\partial u_j} \cdot \frac{\partial u_j}{\partial w_{ij}} = EI_j \, \nu_i. \tag{13}$$

4. Вычислить скорость EA изменения погрешности при изменении активности нейрона предыдущего слоя:

$$EA_i = \frac{\partial Q}{\partial v_i} = \sum_j \frac{\partial Q}{\partial u_j} \cdot \frac{\partial u_j}{\partial v_i} = \sum_j EI_j \, w_{ij}. \qquad (14)$$

5. Провести модификацию межнейронных связей по правилу градиента:

$$w_{ij}[t+1] = w_{ij}[t] + \gamma \, EW_{ij}, \qquad (15)$$

где t — номер шага обучения; γ — скорость обучения (шаг итерации).

Перейти к следующему слою.

6. Если данный слой не является входным, все процедуры повторить начиная с шага 2.

Обучение продолжается до тех пор, пока погрешность не снизится до приемлемого уровня.

Компьютерный эксперимент. При исследовании возможности идентификации нелинейных объектов нейронными сетями обучающая выборка генерировалась из априорно заданной нелинейной модели-эталона. С помощью обучающей выборки определялись параметры нейронной сети. Полученная модель нелинейного объекта сравнивалась с эталонной моделью.

Нелинейный объект-эталон представляется в следующем виде:

$$y(x) = \frac{3\,(4x-2)\,(4x-3)\,(4x-3{,}7)\,(4x-1{,}3)\,(4x-0{,}2)+20}{40}, \qquad (16)$$

$$x \in [0, 1],$$

Его форма изображена на рис. 3. Нейронная сеть-аппроксиматор, соответствовавшая данному объекту, аналогична показанной на рис. 2, с той лишь разницей, что она являлась полносвязанной сетью [3], т. е. каждый нейрон ее текущего слоя был связан с каждым нейроном следующего слоя. Вид исследуемой модели до и после обучения сети-аппроксиматора представлен на рис. 3.

Рис. 3

На рис. 4 показана динамика обучения нейронной сети — зависимость погрешности сети Q от количества N циклов обучения (предъявления обучающих пар вход-выход). Обучающая выборка состояла из 51 пары вход-выход. Обучение нейронной сети было проведено за 11000 обучающих циклов.

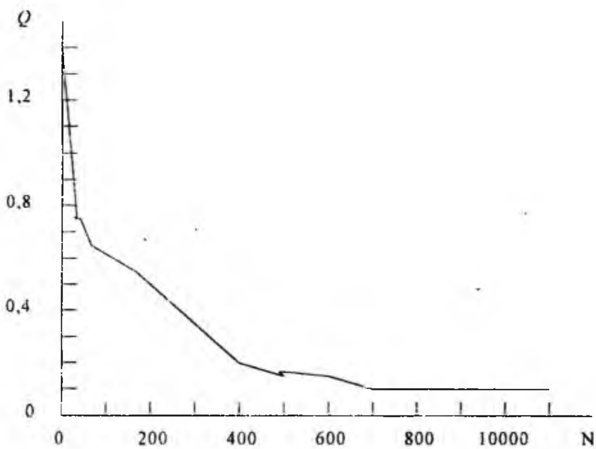


Рис. 4

Выводы. В данной работе предложена идея использования нейронных сетей для идентификации нелинейных объектов. Для проверки этой идеи разработаны методика использования нейронной сети в качестве универсальной модели-аппроксиматора и алгоритм ее обучения на основе известного метода обратного распространения ошибки. Компьютерные исследования показали высокую сходимость модельных и эталонных результатов. Поэтому предложенная методика может быть использована для обработки экспериментальных данных в случаях сильной нелинейности объектов исследования.

На наш взгляд, перспективным направлением дальнейших исследований является создание идентификаторов нелинейных объектов путем композиции нейронных сетей с нечеткой логикой. Это даст возможность уменьшить объем обучающей выборки и длительность обучения посредством использования экспертных знаний об объекте идентификации, представленных в виде высказываний "если — то" [7].

Список литературы: 1. *Кузин Л.Т.* Основы кибернетики: В 2 т. М.: Энергия, 1979. Т. 2. 584 с. 2. *Нейрокомпьютеры* и интеллектуальные роботы / Под ред. Н.М. Амосова. К.: Наук. думка, 1991. 272 с. 3. *Аведьян Э.Д.* Алгоритмы настройки многослойных нейронных сетей // Автоматика и телемеханика. 1995. № 4. С. 106–118. 4. *Джеффри Е. Хинтон.* Как обучаются нейронные сети // В мире науки. 1992. № 11–12. С. 103–110. 5. *Ту Дж., Гонсалес Р.* Принципы распознавания образов: Пер. с англ. М.: Мир, 1978. 411 с. 6. *Цыпкин Я.З.* Основы информационной теории идентификации. М.: Наука, 1984. 320 с. 7. *Ротштейн О.П., Катєльніков Д.І.* Ідентифікація нелінійних об'єктів нечіткими базами знань // Вісн. Вінниц. політехн. ін-ту. 1997. № 4. С. 98–103.

Поступила в редколлегию 29.06.98

Проблемы бионики
1998. Вып. 49. С. 175—180

УДК 519.712

П.Л. САМОФАЛОВ

ИСПОЛЬЗОВАНИЕ ТАБЛИЦ РЕШЕНИЙ В ОПИСАНИИ ИНТЕЛЛЕКТУАЛЬНЫХ ЗАДАЧ

Для повышения эффективности программных систем с элементами искусственного интеллекта нами предлагаются методы представления знаний таблицами решений. Особенность подхода заключается в замене логического вывода выводом в аналоге семантической сети из продукций, представленных таблицами решений.

Как известно, представлением знаний называется описание некоторых истинных утверждений с помощью формализованного языка в целях дальнейшего использования знаний в программных системах. Довольно популярной формой представления знаний является логическая. Язык представления формул исчисления предикатов позволяет формулировать описания в виде, близком как к естественному языку, так и к языку программирования. Система программирования ПРОЛОГ дает возможность формулировать базы знаний (БЗ) и запросы к ним в виде логических формул — хорновских дизъюнктов [1]. Система ПРОЛОГ весьма эффективна для представления БЗ и запросов, так как обладает математической строгостью, ясностью и локальностью формулировок. По ее выводу можно проследить цепочку принимаемых решений. Однако системы, базирующиеся на логической форме представления знаний и логической формулировке запросов, весьма неэффективны в реализации. Кроме того, запрос к таким системам формулируется на специальном языке, что требует от пользователя специальных знаний.

Для улучшения представления БЗ предлагается система, запросы к которой выполняются на языке пользователя некоторой предметной области. Языковой процессор, находящийся в составе системы, распознает исходную языковую конструкцию-запрос. На основе распознанной языковой конструкции формулируются временная БЗ (предикаты-факты, предикаты-цели) из соответствующих предикатов, а также предикаты-правила из постоянной БЗ и стратегия вывода. На первый взгляд, такой подход позволяет существенно повысить эффективность логического вывода. Однако при более внимательном рассмотрении оказывается, что время, сэкономленное на логическом выводе, теряется при формулировке временной БЗ и стратегии вывода. Возможно даже, что эффект улучшения будет минимальным.

Более целесообразными представляются отказ от логического вывода в предикатных формулах и замена логического вывода выводом в аналоге семантической сети из продукций, представленных таблицами решений. В пользу такого представления свидетельствует тот факт, что переформулировка естественно-языкового сообщения на языке предикатов возможна лишь при БЗ, аналогичной предикатной. Как показал опыт использования предлагаемого подхода, большая часть знаний из постоянной БЗ оказывается необходимой для формирования временной БЗ.

Отказ от предикатной БЗ означает, что она не реализуется нами в виде программной системы. Но это не означает, что от нее следует полностью отказаться. Такая БЗ весьма полезна при составлении БЗ в виде сети таблиц решений и выборе соответствующих понятий — нетерминальных символов при конструировании грамматики для языкового процессора, который функционирует на основе специальных типов грамматик, использующих атрибуты и специальные семантические функции. Результатом функционирования является синтаксическое атрибутное дерево, которое можно интерпретировать как фрейм или семантическую сеть. На этом дереве с помощью стандартных операций обработки информации, содержащейся в дереве, можно сформулировать продукционную систему, выраженную системой связанных таблиц решений и аналогичную предикатной БЗ.

Таким образом, исходная языковая конструкция с помощью языкового процессора преобразуется в дерево, которое можно рассматривать как некоторый эквивалент языковой конструкции, задающий сущности и отношения между ними, свойства сущностей и отношений. БЗ в виде таблиц решений позволяет проверить все условия, выполняющиеся на дереве, и принять соответствующие решения-действия, чтобы дать ответ. БЗ в виде таблиц решений обладает всеми теми преимуществами, что и предикатная БЗ, но превосходит ее в эффективности реализации.

Продемонстрируем на конкретном примере методику построения БЗ в виде таблиц решений. Пусть предметной областью является некоторый раздел элементарной алгебры, а именно нахождение экстремумов функции. Здесь приходится отвечать на вопросы об экстремумах самой функции, а также о значениях аргументов, для которых функция имеет экстремумы. Знания, необходимые для ответов на эти вопросы, зависят от определенных функций. В частности, если функция определена как квадратный трехчлен ax^2+bx+c $(a \neq 0)$, знания выражаются равенством

$$ax^2 + bx + c = a(x + b/2a)^2 - (b^2 - 4ac)/(4a). \qquad (1)$$

Из (1) можно установить значение аргумента, при котором функция имеет экстремум ($x = -b/a$), или значение экстремума ($f(x) = (-b^2 - 4ac) / (4a)$).

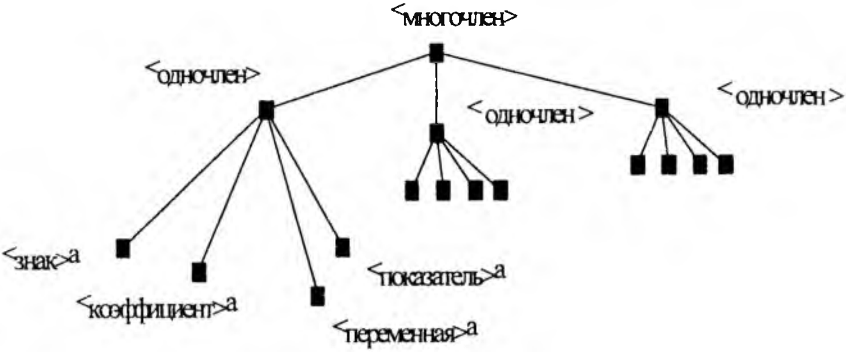
Теперь можно сформулировать в предикатной (дизъюнктивной) форме правила определения максимума, аргумента функции, где имеется экстремум, для квадратного трехчлена:

$$\neg F(x) \vee \neg КВТР(a,b,c) \vee \neg УСЛ(a,<) \vee MAXF(f_1(a,b,c)); \qquad (2)$$

$$\neg F(x) \vee \neg\ КВТР(a,b,c) \vee \neg\ UMAX \vee \neg\ УСЛ(a,<) \vee$$
$$\vee ЗНАЧХ(f_2(a,b)). \qquad (3)$$

Здесь $F(x)$ — определение функции от одного аргумента; *КВТР* — квадратный трехчлен с коэффициентами a, b, c; *УСЛ* — условие, наложенное на коэффициент a; *MAXF* — максимальное значение функции; *UMAX* — условие максимума; *ЗНАЧ* — значение аргумента; $f_1(a,b,c) = -(b^2 - 4ac)/(4a)$; $f_2(a,b) = -b/(2a)$. Формулы (2), (3) являются фрагментом БЗ, которая позволяет отвечать на вопросы об экстремальных значениях функций и их аргументах (возможно, о значениях функций при заданных аргументах и о значениях аргументов при заданных значениях функций).

Покажем, как переформулировать знания (2), (3) в связанную систему таблиц решений. Основой для формирования таких таблиц служат атрибутное синтаксическое дерево D, полученное в результате работы языкового процессора, и предикатная БЗ. Каждая таблица решений в верхней части в первом столбце содержит предикаты-условия; значения, которые они принимают, определяют ситуацию. При совпадении ситуации со столбцом R_i выполняются отмеченные в этом столбце действия, задаваемые процедурными операторами [2; 4]. Условия таблиц решений формируются с помощью априорно определенных операций на D, таких как: включение вершины, обозначенной некоторым символом, в D (<многочлен> $\in$ D); определение метки корня дерева (корень (D)=<уравнение>); установление существования вершины, помеченной символом X ,такой что ($X \Rightarrow Y$); установление того факта, что вхождение некоторого символа в поддерево дерева не превышает заданного числа: (вхождение (<одночлен>) ≤ 3) $\in$ *ПОДДЕР(X)*, здесь *ПОДДЕР*(X) — поддерево дерева D с корнем в вершине, помеченной символом X; фиксация некоторой вершины в виде переменной (одночлен ~Y).



Использование формул (2) и (3) возможно лишь при выполнении таких условий: “определение функции одной переменной”, “наличие квадратного трехчлена”, “наличие запроса на вычисление максимума или значения переменной”.

Выполнение всех этих условий должно быть проверено на дереве D, поскольку <трехчлен> — частный случай категории <многочлен> и его не имеет смысла включать в синтаксическое дерево, а проверять это условие нужно только по необходимости. Поэтому категория <трехчлен>$_{a,b,c}$ с атрибутами a, b, c в качестве нетерминала должна быть включена в грамматику. Ему будет соответствовать трехчлен с атрибутами a, b, c, являющимися значениями коэффициентов трехчлена. Однако в дерево вывода эта категория включена не будет, вместо нее там будет более общая категория <многочлен>, из которой и должна быть выведена категория <трехчлен>.

Таблица 1

	TP1	R_1	R_2	R_3	R_4	R_5	E
1	Корень(D) =(‘найти’ $\vee$ ‘вычислить’ $\vee$ $\vee$‘определить’)	0	1	1	1	0	
2	<определение функции одной переменной> $\in$ D	1	1	1	1	—	
3	(<многочлен> ~ X_1) $\in$ D	1	1	0	0	—	
4	<экстремум> $\in$ D	1	0	1	0	—	
		TP2	TP3	TP4	TP5	TP6	TP7

Для подобного вывода необходимо обладать знаниями о структуре <многочлен>, т.е. о сущностях, являющихся составными частями <многочлена>. Такой сущностью будет категория <одночлен>, состоящая, в свою очередь, из категорий <знак>, <коэффициент>, <переменная>, <показатель>. Соотношения между перечисленными категориями представлены на рисунке (заметим, что это не имеет отношения к синтаксическому дереву — там может быть сформировано совершенно другое дерево, зависящее от продукций грамматики).

Таблица 2

	ТР2	R_1	R_2	R_3	R_4	R_5	E
1	(Вхождение (<одночлен> ≤ 3) ∈ (ПОДДЕР(ТР1.X1) ~ Y1)	1	1	1	1	0	
2	((<одночлен> ~ Y2) ⇒ (<показатель>.a = 2)) ∈ Y1	1	1	1	1	—	
3	((<одночлен> ~ Y3) ⇒ (<показатель>.a = 1)) ∈ Y1	1	0	0	1	—	
4	((<одночлен> ~ Y4) ⇒ (<показатель>.a = 0)) ∈ Y1	1	0	1	0	—	
1	<трехчлен>.a : = conc(Y2 ⇒ <знак>.a, Y2 ⇒ <коэффициент>.a)	x	x	X	X		
2	<трехчлен>.b : = conc(Y3 ⇒ <знак>.a, Y3 ⇒ <коэффициент>.a)	x			X		
3	<трехчлен>.c : = conc(Y4 ⇒ <знак>.a, Y4 ⇒ <коэффициент>.a)	x		x			
4	<трехчлен>.b : = 0		x	x			
5	<трехчлен>.c : = 0		x		X		
6	ТР2.2	x	x	x	X		
7	ТР2.1					X	
8	ТР7						x

Символы <знак>$_a$, <коэффициент>$_a$, <переменная>$_a$, <показатель>$_a$, имеют в качестве значений атрибута строки, выражающие знак, коэффициент, имя переменой и ее показатель степени в соответствующем одночлене. Все нетерминалы, указанные на рисунке, должны присутствовать в грамматиках для языкового процессора в продукциях, задающих структуру определенных понятий. Кроме того, языковой процессор должен воспринимать слова типа “известно, что”, “дано”, “пусть”, “имеем” в качестве задания функции, а слова “найти”, “вычислить”, “определить” — в качестве последовательности действий, определенных лингвистической БЗ. От этой последовательности действий будет зависеть выбор необходимой части БЗ в виде таблиц решений.

С учетом изложенного построим фрагмент БЗ в виде таблиц решений. Он соответствует формулам (2), (3) в виде системы связанных таблиц решений. Эти таблицы имеют имена, которые далее приводятся в ссылках.

В ТР1 (табл. 1) зафиксирована, с помощью переменной X1, вершина дерева, помеченная как <многочлен>, поскольку в ТР2 (табл. 2) решается вопрос, является ли данный многочлен трехчленом. В многочлене отсутствуют подобные члены (этому условию легко удовлетворить, выполнив при синтаксическом анализе неявную операцию “приведение подобных”).

Таблица 3

	ТР2.1	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8
1	<трехчлен>.a < 0	1	1	1	0	0	0	1	0
2	<max> ∈ D	1	1	0	0	1	1	0	0
3	<знач x> ∈ D	1	0	1	1	0	1	0	0
1	F_1(<трехчлен>.a, <трехчлен>.b, <трехчлен>.c)	x	x						
2	F_2(<трехчлен>.a, <трехчлен>.b)	x							
3	Max . = '∞'					x	x		
4	Перем : = '±∞'						x		
	ТР 2.1.1			x	x			x	x
6	Выход с ответами на вопрос	x	x			x	x		

В ТР2 использована функция conc — конкатенация двух строк (знака и коэффициента). В ТР1, ТР2, ТР2.1 (табл. 3) переменные *Xi*, *Yi* представляют собой рабочие переменные; они фиксируют некоторые вершины дерева и упрощают, кроме того, запись условий. Проанализировав таблицы решений, можно отметить их легкую расширяемость, модифицируемость. Можно утверждать, что приобретение знаний в различных БЗ в виде таблиц решений является систематическим, структурированным процессом, базирующимся на предикатной БЗ и структуре сущностей, выраженных в предикатной БЗ.

Таблица решений, превращенная в схему программы (или, вернее, в программу), работает более эффективно, чем продукционная БЗ. Аналогом резолюции в поиске по дереву решений становится операция проверки некоторого условия в дереве решений. При этом дерево решений может быть подвергнуто оптимизации чисто формальными методами [3; 4].

Список литературы: 1. *Логический* подход к искусственному интеллекту: от классической логики к логическому программированию: Пер. с фр./ А. Тейз, П. Грибомон, Ж. Луи и др. М.: Мир, 1990. 432 с. 2. *Введение* в технику работы с таблицами решений: Пер. с нем. / Г. Фрайтаг, Х. Годе, Т. Буш и др. М.: Энергия, 1977. 88 с. 3. *Hoover D. N., Chen Z.* A mathematical tool for analyzing decision tables // Nat. aeronautics and space administration (NASA) contractor rep. 195027, Nov. 1994. P. 10–17. 4. *Дюбко Г. Ф., Самофалов П. Л.* Таблицы решений и их применение при разработке программного обеспечения. Х.,1996. 26 с. Деп. в ГНТБ Украины 12.08.96, № 1686-Ук96.

Поступила в редколлегию 25.04.97

Проблемы бионики
1998. Вып. 49. С. 181—185

ХТУРЭ

УДК 371.315.7: 681.513.6

В.В. ТИЩЕНКО

ОБ ОДНОЙ МЕТОДОЛОГИИ РАЗРАБОТКИ ПРОГРАММНЫХ МОДЕЛЕЙ

Рассмотрим методологию построения программных моделей, основанную на причинно-следственных отношениях. В предлагаемой методологии не разделяются средства анализа и методы моделирования. Они соединены в единое целое в поисковой структуре, которая включает в себя базу данных, содержащую операционную основу отношений, подходов, способов и методов моделирования.

Методология основана на аксиоме эмерджентности, существо которой состоит в следующем [1]. Пусть имеют место: S (x, y, z)-система с параметрами; x, y, z – объекты, свойства, связи, представляющие конкретные множества; ψ(a, b, c) – программная модель с атрибутами; a, b, c – предметная область, план эксперимента, критериальные оценки. Отметим, что S-система обладает традиционными параметрами, а ψ первоначально имеет: «a» – структуру вида $(A,\leq)$, «b» – функциональное отношение вида ($\cup$ или $\cap$), «c» – функциональные оценки по критерию. Тогда, если существует «проект» вида $\varphi = \varphi[S(x,y,z), \psi(a,b,c)]$, то

$$P(S, \psi) = \begin{cases} 1, \text{если} & S = \psi; \\ 0, \text{если} & S \neq \psi. \end{cases}$$

Используем предикат эмерджентности; отсюда вытекает цепь преобразований:

$$S\,(x, y, z) \vdash S_1(x_1, y_1, z1) \vdash S_2(x_2, y_2, z_2) \vdash \ldots \vdash \psi(a, b, c).$$

Следовательно, если P(S, ψ) = 1, то возможна цепь преобразований, которые направлены на построение программной структуры, реализующей искомый алгоритм. Практически в этой методологии учитываются и используются:

1. Основные свойства системы, состоящие в том, что «целое больше суммы элементов искомой опорной информации с основными отношениями внутри элементов. Для восприятия целого его необходимо квантифицировать на элементы опорной информации с основными отношениями внутри их; затем собрать их, композировать с учетом связей между ними и углубленным пониманием целого в эксперименте» [2].

2. Свойства системного анализа, заключающиеся в том, что при анализе нужно иметь «стандартный универсум» с элементами стандартной

опорной информации и отношениями на них и конструктивное множество аналогичной, но определенной в процессе анализа информации для модели.

3. Инструментарий преобразований: перенос свойств элементов определенной опорной информации в стандартный универсум путем достижения одинаковой истины в нем и конструктивном множестве; расширение свойств элементов стандартного универсума на конструктивное множество путем компактификации и пополнения последнего. Отметим, что объекты, свойства и связи группируются в компактных областях полного признакового n-мерного гиперпространства с использованием известного логического принципа «reductio ad absurdum» (доведение до абсурда) по отношению к элементам внутреннего и внешнего множеств определенной опорной информации [3].

Конструкция модели причинно-следственной методологии представляется опрокинутым гиперконусом, который разделен на страты (послойные группировки точечных моделей), кластеризованные в отношения признаков графа близости (политетические классы). Форма гиперконуса наглядно отражает основное свойство моделирования – поэтапное понижение неопределенности информации при конструировании моделей [4]. В основании гиперконуса лежит множество гипотетических моделей (ГМ), структуры которых определены, с одной стороны, заданием, а с другой – базой знаний.

В вершине гиперконуса находится программная модель (ПМ), а промежуточные страты обусловлены множеством концептуальных моделей (КМ), индуцированных из страты ГМ совместно с информацией базы умения и схематологических моделей (СМ), агрегированных с предшествующей стратой и информацией базы навыков.

На первой стадии модель на своем уровне представляет некоторое подпространство признаков, сгруппированных в операционные таксономические единицы (ОТЕ); на второй — системы редуцируемые комбинаторными единицами моделирования (КЕМ); на третьей — специфицированные программно-схематологическими единицами системы (технологии).

Тогда под моделью причинно-следственной методологии понимается некоторый кортеж (триплет) атрибутов виртуальной системы:

$$M(A, R, Q),$$

где $A = \{(a_1, a_2, ..., a_n), (b_1, b_2, ..., b_m)\}$ – множество (ассоциация) подмножеств, конструктивных объектов и отношений между множествами (опорная информация); $R = \{(r_1, r_2, ..., r_n), (s_1, s_2, ..., s_m), (c_1, c_2, ..., c_k)\}$ — множество r_i-состояний, s_i-связей, c_i-ограничений некоторого функционального пространства преобразования; Q — оценки компактности, целостности в форме d или μ–функции принадлежности и т. д.

Тогда в постадийном пространстве преобразований на гипотетической стадии под моделью понимают

ГМ(A1, R1, Q1),

где A1 — множество конструктивных объектов и отношений на множестве; R1 — область определения модели; Q1 — функция нормы, определяющая расстояние между областями и ошибку, по отношению к БЗ.

На концептуальной стадии моделью является

КМ(A2, R2, Q2),

где A2 — множество КЕМ и связей между ними; R2 — область значений модели; Q2 — функция принадлежности модели к стандартным ОТЕ или БУ.

На схематической стадии модель выражается следующим образом:

СМ(A3, R3, Q3),

где A3 — структуры модели как совокупности объектов и связей; R3 — среда (внутренняя, внешняя) как множество способов представления; Q3 — оценки модели по отношению к заданию и база навыков (БН) в форме эквивалентности, подобия и адекватности.

И, наконец, программная стадия, которая определяет модель таким образом:

ПМ(A4,R3,Q4),

где A4 — рабочая область модели как множество программных объектов и структур; R4 — план эксперимента и множество текстов; Q4 —оценки сложности, надежности, мобильности и репрезентативности.

Программная модель не является окончательным программным продуктом, но может являться объектом исследования и постановкой задачи в процессе получения окончательного продукта.

Наше субъективное представление о системе всегда носит форму некоторого «Задания», имеющего следующее представление:

ЗД(U,T,L),

где U — множество результатов, трактуемых как множество целей; T — множество требований к отношениям, фигурирующее в форме признаков; L — множество ограничений, накладываемых на кортежи признаков. Такой подход способствует преобразованию субъективных параметров ЗД в атрибуты «Задачи» (ЗЗ), т.е. в программную модель, отработанную в процессе исследования с последующим кодированием, документированием и получением программного продукта. Основу причинно-следственной методологии образуют стандартные универсумы, представленные в форме баз данных: базы знаний (БЗ), умений (БУ) и БН.

БЗ – это гиперпространство признаков (П-пространство), или ассоциация подходов. Оно расчленено на монотетические классы «стандартных» пространств (области определения дескриптивных, динамических моделей, конечных автоматов и т.д. или пространства, выраженные в форме графов близости с поуровневыми и послойными группировками). Пространство рассматривается как ассоциация множеств, отношений и аксиом, трактуемых как операционные таксономические единицы.

БУ – это ассоциация методов конструктивного пространства, представленного в форме объектов КЕМ, связей и аксиом, отражающих полномочия КЕМ. В качестве объектов (КЕМ) используют операторы, события, процессы, транзакты, агрегаты и т. д. Для представления связей могут применяться операторные методы, событийные, компилятивные, интерпретативные и др.

Конструктивное пространство БУ является областью значений той области определения, которая сконструирована в БЗ.

БН – это ассоциация способов представления программных единиц в конструктивном пространстве, специфицированных в форме подпрограмм, модулей и т. п., фундаментальных отношений и аксиом, отображенных в формальных моделях типа структурного графа встроенных систем. Первоначально «Цепочка преобразований» на гиперконусе причинно-следственной методологии представляет собой, в общем случае, поуровневые и послойные преобразования графа близости путем редукции политетического признакового класса пространства в монотетические признаковые классы на основе разработки проектов стадии.

На первой стадии (разработка гипотетической модели) создается проект φ1[ЗД, ГМ, Р1(ЗД, ГМ)], ставящий целью абстрагирование информации ЗД путем формализации и квантификации информации в ГМ с последующей оценкой целостности:

$$\text{P1(З.0, ГМ)} = \begin{cases} 1, \text{С.О.Ц.ЗД} = \text{О.О.ГМ}; \\ 0, \text{С.О.Ц.ЗД} \neq \text{О.О.ГМ}, \end{cases}$$

где С. О. Ц. ЗД – субъективная область цели задания; О.О.ГМ – область определения гипотетической модели.

На второй стадии (разработка концептуальной модели) создается проект φ2 [ГМ, КМ, Р2(ГМ,КМ)], цель которого – композиция информации ГМ путем редукции и выбора с последующей оценкой целостности:

$$\text{P2(ГМ, КМ),О.О.Г} = \begin{cases} 1, \text{О.О.Г} = \text{О.З.КМ.}; \\ 0, \text{О.О.ГМО.} \neq \text{О.З.КМ.}, \end{cases}$$

где О. З. КМ. – область значений КМ.

На этапе преобразования стадийных моделей в схематологическую модель разрабатывается проект этапа $\psi\{(\text{ГМ, КМ}), \text{СМ}, P[(\text{ГМ, КМ}), \text{СМ}]\}$, чтобы обеспечить конструирование СМ путем агрегирования информации ГМ, КМ с последующей оценкой целостности:

$$P[(\text{ГМ,К М})] = \begin{cases} 1, \text{О.О. и З.М.} = \text{О.П.М.}; \\ 0, \text{О.О. и З.М.} \neq \text{О.П.М.}, \end{cases}$$

где О.О., З.М. – области определения и значения моделей; О.П.М. – область представления модели.

На этапе преобразования стадийных моделей в схематологическую модель создается проект этапа

$$\psi\{(\text{ГМ,КМ}),\text{СМ},P[(\text{ГМ,КМ}),\text{СМ}]\}$$

для конструирования СМ путем агрегирования информации ГМ, КМ с последующей оценкой целостности:

$$P[(\text{ГМ,КМ}),\text{СМ}] = \begin{cases} 1, \text{О.О. и З.М.} = \text{О.П.М.}; \\ 0, \text{О.О. и З.М.} \neq \text{О.П.М.}, \end{cases}$$

где О.О., З.М. – области определения и значения моделей; О.П.М. – область представления моделей.

Сравнения производятся на основе критериальных оценок: эквивалентности, подобия и адекватности.

Аналогичным образом создается и программный проект на основе спецификации и определения целостности исходя из критериальных оценок структуры модели и редуцирования.

Предложенная методология имеет широкий спектр применения, в частности в биологии и технике, благодаря накопленному опыту конструирования моделей и инструментария сервисных программ.

Список литературы: 1. *Горбатов В. А.* Теория частично-упорядоченных систем. М.: Сов. радио, 1976. 336 с. 2. *Пешель М.* Моделирование сигналов и систем: Пер. с англ. М.: Мир, 1981. 300 с. 3. *Классификация* и кластер / Под ред. Дж. Вэн Райзина: Пер. с англ. М.: Мир, 1980. 390 с. 4. *Моисеев Н. Н.* Математические задачи системного анализа. М.: Наука, 1981. 48 с.

Поступила в редколлегию 11.02.98

Проблемы бионики
1998. Вып. 49. С. 186--195

УДК 681.142.1.01

В.Е. ХОДАКОВ, В.Г. ШЕРСТЮК, К.Г. СТЕПАНСКИЙ, А.А. ДИДЫК, А.Н. МАРТЫНОВ

МЕТОДЫ ОЦЕНКИ РЕЛЕВАНТНОСТИ ИНФОРМАЦИОННЫХ СТРУКТУР В БАЗАХ ЗНАНИЙ

При построении интеллектуальных систем информацию, имеющуюся в базах знаний, желательно рассматривать не как отдельные факты или правила, а как единую структуру. Важное значение при построении модели такой структуры имеет реализация механизма оценки релевантности одних элементов знаний системы другим. Наличие релевантности между элементами знаний подразумевает совместимость этих элементов в одном и том же контексте. Знание того, насколько элементы структуры релевантны друг другу, позволяет не только проводить более полный анализ уже сложившейся ситуации в этой структуре, но и, что более важно, прогнозировать последствия для всей структуры при изменении ее элементов.

Зачастую релевантность трактуется как мера, однозначно определяющая, релевантен данный элемент другому или нет, и не допускающая никаких промежуточных значений. Однако подобный подход применим только при построении частных моделей ряда структур, а при построении обобщенной модели можно воспользоваться математическим аппаратом теории нечетких множеств и теории возможностей [1; 2].

Рассмотрим организованную [3] и находящуюся в стабильном состоянии структуру из элементов, связанных множеством отношений. Задача состоит в определении релевантности элементов структуры по отношению друг к другу.

Пусть X – множество всех элементов, находящихся в структуре S, т.е. $\forall x_i \in X$, причем все элементы множества X организованы в структуру S путем определения множества бинарных отношений R между элементами X. Множество R состоит из всех возможных на множестве пар $X \times X$ видов отношений, с помощью которых могут быть организованы элементы X.

Предположим, что множества X и R конечны и, кроме того, все отношения, принадлежащие множеству R, независимы друг от друга [4]. Предположим также, что структура находится в стабильном состоянии, т.е. все связи между элементами уже установлены и имеют некоторое стабильное значение. Иными словами, структура S полностью организована.

Определимся со значениями, которые принимают отношения на конкретной паре элементов структуры. Предположим, что все отношения, принадлежащие множеству R, являются нечеткими отношениями [2]. Заметим, что удобнее использовать более общую структуру, чем отрезок [0, 1], а под нечетким отношением понимать функцию $R: X_1 \times X_2 \times ... \times X_n \to L$, отображающую декартово произведение множеств $X_1, ..., X_n$ в L. В качестве L могут быть выбраны множество вещественных чисел, множество лингвистических переменных, множество векторов различной арности, цепи, псевдобулевы алгебры, дистрибутивные решетки и т.д. Такой подход к определению понятия нечеткого отношения дает возможность обобщить это понятие, что может быть использовано в теории моделей, и интерпретировать функции со значениями из L как нечеткие, используя для изучения свойств этих функций аппарат теории нечетких множеств. Однако отношения с различными областями определения будет сложно сопоставлять между собой; между тем это необходимо при установлении полной релевантности. Поэтому будем считать, что все области определений отношений, принадлежащих множеству R, приведены к единому масштабу, т.е. к промежутку [0, 1]:

$$\forall r_i \in R: X \times X \to [0,1]. \quad (1)$$

Ввиду того что отношения, принадлежащие множеству отношений R, могут иметь различный характер, следует для каждого отношения из множества R ввести специфическую именно для данного отношения операцию композиции [2]. Обозначим эту операцию через γ_{r_i}. Операция композиции для каждого отношения однозначно характеризует его. Поэтому элемент множества R можно представить так:

$$r_i = \{\gamma_{r_i}\}, \; r_i \in R \quad (2)$$

Для определения релевантности двух элементов x и y необходимо выделить все возможные связи между ними. Эти связи могут быть двух типов:

- прямая связь, когда нечеткое отношение определено именно на паре (x, y);
- косвенная связь, когда элементы связаны транзитивной цепочкой.

Под транзитивной цепочкой будем понимать путь [5; 6] от одного элемента к другому в нечетком графе [2], соответствующем нечеткому отношению. Процесс определения транзитивных цепочек можно показать на следующем примере.

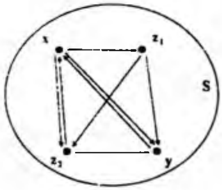


Рис. 1

На рис. 1 представлена структура S с множеством элементов $X = \{x, y, z_1, z_2\}$ и множеством отношений, состоящим из одного элемента $R = \{r\}$, причем r – транзитивное отношение, а иначе в структуре S были бы возможны только прямые связи.

В структуре S между ее элементами x и y существуют такие цепочки связей: одна прямая связь $\{x, y\}$; $\{x, z_1\}$, $\{z_1, y\}$; $\{x, z_1\}$, $\{z_1, z_2\}$, $\{z_2, y\}$; $\{x, z_2\}$, $\{z_2, y\}$. Выборка всех возможных цепочек функционально соответствует полному множеству путей из вершины x в вершину y в нечетком графе, отвечающем отношению r. Для упрощения эти цепочки можно обозначать таким образом: xy, xz_1y, xz_1z_2y, xz_2y.

Для решения поставленной задачи существенно, какие значения отношение, по которому строится транзитивная цепочка, принимает на конкретной паре элементов. Поэтому вместо наименований элементов, входящих в цепочку, элемент транзитивной цепочки должен содержать значение отношения [5; 6], по которому строится эта цепочка, определенного на паре, состоящей из предыдущего и данного элементов. Таким образом, цепочки по отношению r в структуре, представленной на рис. 1, имеют вид $r(x, y)$; $r(x, z_1)$, $r(z_1, y)$; $r(x, z_1)$, $r(z_1, z_2)$, $r(z_2, y)$; $r(x, z_2)$, $r(z_2, y)$.

Множество цепочек между элементами x и y по всем отношениям из множества R в структуре S обозначим через C. Множество транзитивных цепочек по одному отношению обозначим как C^r, где r – некоторое отношение. Каждый элемент множества C^r представляет собой линейно упорядоченное множество, характеризующее цепочку. Ясно, что множество всех цепочек является объединением множеств цепочек, соответствующих всем отношениям множества R (т.е. $C = \bigcup\limits_{r_i \in R} C^{r_i}$).

Далее определим отношение релевантности:

$$\mathrm{Rel}: X \times X \to [0, \infty). \qquad (3)$$

Это бинарное нечеткое отношение [2], определенное на множестве пар элементов, принадлежащих множеству X. Областью значений данного отношения является множество всех действительных чисел на промежутке от нуля до бесконечности. Если отношение релевантности принимает нулевое значение, это означает, что элементы x и y не имеют каких-либо связей между собой и никакого влияния друг на друга оказывать не могут. Бесконечно большое значение отношение релевантности принимает только в том случае, когда x и y – один и тот же элемент структуры (т.е. $\mathrm{Rel}(x,\ x)$).

У отношения релевантности имеются следующие свойства:

1. Строгая рефлексивность относительно бесконечности:

$$\forall x, y \in X\,\mathrm{Rel}(x,\ y) = \infty,\ \text{если}\ x = y. \tag{4}$$

2. Асимметричность:

$$\forall x, y \in X\,\mathrm{Rel}(x,\ y) \neq \mathrm{Rel}(y,\ x). \tag{5}$$

3. Транзитивность:

$$\mathrm{Rel}(x,\ z) = \mathrm{Rel}(x,\ y) * \mathrm{Rel}(y,\ z), \tag{6}$$

где * – операция композиции, определенная на отношении релевантности. Выбор операции композиции зависит от решаемых задач, но результат композиции отношений релевантности всегда находится в обратно пропорциональной зависимости от длины транзитивной цепочки.

Формализуем это отношение на структуре S. Несомненно, наибольшее влияние на степень релевантности некоторого элемента будут иметь те элементы структуры S, которые напрямую связаны с данным элементом, т.е. для которых $\left|C_i^r\right| = 1$. При увеличении длины транзитивной цепочки ее влияние на степень релевантности должно уменьшаться согласно некоторому закону. Последним является операция композиции γ_r, определенная для отношения r. Причем различия между операциями композиции отношений, принадлежащих множеству R, состоят в степени кривизны графика зависимости релевантности от длины цепочки. Чем длиннее транзитивная цепочка, тем меньше взаимосвязь элементов [5; 7].

Теперь становится возможным формально описать отношение релевантности по некоторому отношению r. Для начала определим релевантность по цепочке C_j. Значение релевантности по одной цепочке отношения образуется путем последовательной композиции элементов цепочки C_j согласно операции композиции (2), соответствующей отношению r. Исходя из этого значение релевантности по одной цепочке имеет вид

$$\mathrm{Rel}_r^{C_j^r}(x,\ y) = \frac{1}{\left|C_j^r\right|} \mathop{\gamma_r}_{k=1}^{\left|C_j^r\right|} c_k,\ c_k \in C_j^r. \tag{7}$$

Множитель $\frac{1}{\left|C_j^r\right|}$ является нормализующим для приведения формулы (7) в соответствие с условием (4) и обеспечивает обратно пропорциональную зависимость релевантности от длины цепочки. Однако она должна соответствовать условию строгой рефлексивности отношения релевантности. Это условие выполняется в (7) только при $r(x,\ x) \neq 0$, а иначе получается неопределенность (0/0). Чтобы избежать этого, можно в формулу (7) ввести некоторый параметр ε, который принимает нулевое значение тогда и только тогда, когда длина цепочки больше нуля:

$$\varepsilon = \begin{cases} 0, \text{ если} \left|C_j^r\right| > 0; \\ > 0, \text{ если} \left|C_j^r\right| = 0. \end{cases}$$

С добавлением этого параметра формула (7) примет вид

$$\mathrm{Rel}_r^{C_j^r}(x,\ y) = \frac{1}{\left|C_j^r\right|} \cdot \left(\prod_{k=1}^{\left|C_j^r\right|} \gamma_r\, c_k + \varepsilon \right),\ c_k \in C_j^r. \qquad (8)$$

Значение параметра ε при нулевой длине цепочки не играет важной роли, поскольку значение релевантности в этом случае равно бесконечности. Получив значение релевантности по одной цепочке, можно определить отношение релевантности по всем цепочкам.

Каждая цепочка является независимой от всех других, т.е. релевантность по одной цепочке не зависит от существования других. Поэтому общая релевантность по отношению r получается путем наложения релевантностей всех цепочек друг на друга. Оценка общей релевантности по отношению r

$$\mathrm{Rel}_r(x,\ y) = \sum_{j=1}^{\left|C^r\right|} \mathrm{Rel}_r^{C_j^r}(x,\ y). \qquad (9)$$

Как видно, выражение (9) соответствует условию строгой рефлексивности по отношению к ∞. Аналогичным образом определяется и релевантность между элементами x и y:

$$\mathrm{Rel}(x,\ y) = \sum_{i=1}^{|R|} \mathrm{Rel}_{r_i}(x,\ y). \tag{10}$$

Развернув выражение (10) с помощью формул (9) и (8), получим формулу оценки общей релевантности:

$$\mathrm{Rel}(x,\ y) = \sum_{i=1}^{|R|} \sum_{j=1}^{|C^{r_i}|} \left(\frac{1}{|C_j^{r_i}|} \cdot \left(\gamma_{r_i} {}_{k=1}^{|C_j^{r_i}|} c_k + \varepsilon \right) \right),\ c_k \in C_j^{r_i}. \tag{11}$$

До настоящего момента предполагалось, что все отношения, принадлежащие множеству R, являются равноправными в структуре S и одинаково влияют на релевантность одного элемента структуры другому. Однако чаще всего возникают ситуации, когда одни отношения больше влияют на релевантность элементов, а другие меньше, т.е. одни отношения организуют более сильную взаимосвязь элементов структуры, другие – менее сильную взаимосвязь, а некоторые, несмотря на наличие этих отношений между элементами, вообще не оказывают влияния на степень релевантности между элементами. Для формализации влияния значимости отношений на степень релевантности добавим к определению (2) дополнительную характеристику μ_{r_i}, которая будет характеризовать степень значимости [1; 2]. Тогда элементы множества R могут быть представлены в виде пары:

$$R = \{\gamma, \mu\}, \tag{12}$$

где γ – операция композиции отношения; μ – степень значимости отношения.

Областью определения степени значимости μ будем считать промежуток $[0,\ +\infty)$. Если $\mu > 1$, это значит, что отношение, которому соответствует μ, будет увеличивать релевантность связанных им элементов структуры. Если $\mu = 1$, то никакого влияния значимость данного отношения на общую релевантность оказывать не будет. Если $0 < \mu < 1$, то данное отношение будет уменьшать релевантность между связываемыми элементами. И если $\mu = 0$, то по данному отношению релевантности между двумя элементами не существует. Тогда оценка релевантности в условиях неравнозначности отношений может быть представлена так:

$$\mathrm{Rel}'(x,\ y)=\sum_{i=1}^{|R|}\left(\mu_{r_i}\cdot\sum_{j=1}^{\left|C^{r_i}\right|}\left(\frac{1}{\left|C_j^{r_i}\right|}\cdot\left(\mathop{\vphantom{\sum}}_{k=1}^{\left|C_j^{r_i}\right|}\gamma_{r_i}\,c_k+\varepsilon\right)\right)\right),\ c_k\in C_j^{r_i}. \qquad (13)$$

Учет приоритетности отношений в исчислении релевантности позволяет строить модели, приближенные к реальным структурам.

Во многих случаях необходимо знать, как элемент некоторой структуры интегрирован в нее и как изменение его характеристик повлияет на данную структуру. Для определения степени интеграции элемента в данную структуру можно использовать некоторую меру [1]. Поскольку релевантность носит аддитивный характер, т.е. общее значение релевантности складывается из релевантностей по независимым отношениям, которые, в свою очередь, складываются из релевантностей по независимым транзитивным цепочкам, можно предположить, что релевантность данного элемента определенной структуре слагается из всех релевантностей, существующих между этим элементом и всеми другими элементами структуры. Правомерна следующая запись:

$$G'^{S}_{\mathrm{Rel}}(x)=\sum_{x_i\in X,\,x_i\neq x}\mathrm{Rel}(x,\ x_i),\ x\in X. \qquad (14)$$

Функцию $G'^{S}_{\mathrm{Rel}}(x)$ можно назвать мерой влияния элемента x на структуру S. Данная мера может принимать значения на промежутке $[0,+\infty)$. Если $G'^{S}_{\mathrm{Rel}}(x)=0$, то изменение характеристик элемента x никак не повлияет на характеристики структуры S, однако несимметричность отношения релевантности не позволяет сделать обратное утверждение. Поэтому необходимо ввести меру влияния структуры на элемент, которая выводится аналогично мере влияния элемента на структуру, с той лишь разницей, что меняется направление отношения релевантности:

$$G''^{S}_{\mathrm{Rel}}(x)=\sum_{x_i\in X,\,x_i\neq x}\mathrm{Rel}(x_i,\ x),\ x\in X. \qquad (15)$$

Далее можно получить меру интеграции элемента в структуру:

$$G^{S}_{\mathrm{Rel}}(x)=\frac{G'^{S}_{\mathrm{Rel}}(x)+G''^{S}_{\mathrm{Rel}}(x)}{2}; \qquad (16)$$

подставим в (16) формулы (14) и (15):

$$G_{\mathrm{Rel}}^{S}(x)=\sum_{x_i\in X,\,x_i=x}\frac{\mathrm{Rel}(x,\,x_i)+\mathrm{Rel}(x_i,\,x)}{2}. \tag{17}$$

Введенную меру интеграции можно трактовать как меру значимости элемента для структуры.

Можно построить меру релевантности для структуры S. Эта мера должна показывать, насколько организованна структура S:

$$G_{\mathrm{Rel}}^{S}=\frac{\sum\limits_{i=1}^{|X|}\sum\limits_{j=1}^{|X|}\mathrm{Rel}(x_i,\,x_j)}{\left|X^2\right|-|X|}. \tag{18}$$

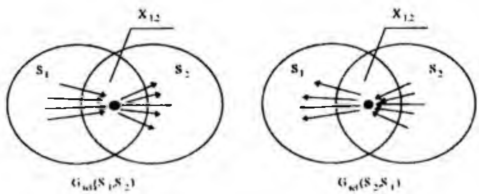
Из (18) исключены все вхождения отношений типа $\mathrm{Rel}(x,\,x)$. Эта мера показывает, насколько устойчива структура S и насколько она чувствительна к изменениям, которые могут происходить в ней. Таким образом, чем большим будет значение G_{Rel}^{S}, тем значительнее будут последствия для этой структуры от любого ее изменения. Эти последствия могут состоять в рассогласовании уже устоявшихся взаимосвязей и нарушении сбалансированности структуры. Другими словами, чем больше значение меры релевантности некоторой структуры, тем более сбалансированной системой является эта структура.

Рассмотрим вопросы взаимосвязи нескольких структур. Предположим, что имеются две структуры S_1 и S_2, состоящие из множеств X_1 и X_2, элементы которых связаны между собой множествами отношений R_1 и R_2. Для этих двух структур возможны такие ситуации:

1. Если $X_1\cap X_2=\varnothing$, то структуры не зависимы друг от друга.

2. Если $X_1\cap X_2\neq\varnothing$, релевантность между ними определяется релевантностью между элементами множества $X_{1,2}=X_1\cap X_2$ и элементами обеих структур, не входящих в $X_{1,2}$.

Элементы множества $X_{1,2}$ имеют некоторое количество связей со структурой S_1 и со структурой S_2. Так как множество является связующим звеном между структурами, релевантность одной структуры относительно другой зависит только от релевантности элементов, входящих во множество $X_{1,2}$. Схематично это показано на рис. 2.



а *б*

Рис. 2

Ввиду несимметричного характера отношения релевантности, релевантность структуры S_1 структуре S_2 (рис. 2, *а*) отличается от релевантности структуры S_2 структуре S_1 (рис. 2, *б*), т.е. $G_{\mathrm{Rel}}(S_1,\ S_2) \neq G_{\mathrm{Rel}}(S_2,\ S_1)$.

Пусть $x \in X_{1,2}$, где $X_{1,2} = X_1 \cap X_2$. Релевантность двух структур по одному элементу x будет состоять из двух слагаемых: релевантности этого элемента всем элементам структуры S_2, не входящим во множество $X_{1,2}$, и релевантности всех элементов S_2, не входящих в $X_{1,2}$, элементу x. Получаем:

$$G_{\mathrm{Rel}}^{(x)}(S_1,\ S_2) = \sum_{x_i \in X_2 - X_{1,2}} \mathrm{Rel}(x,\ x_i) + \sum_{x_i \in X_1 - X_{1,2}} \mathrm{Rel}(x_i,\ x)\,. \qquad (19)$$

Далее можно выразить релевантность структур по всем элементам:

$$G'_{\mathrm{Rel}}(S_1,\ S_2) = \sum_{x_i \in X_{1,2}} G_{\mathrm{Rel}}^{(x_i)}(S_1,\ S_2)\,. \qquad (20)$$

Однако отношение модуля множества $X_{1,2}$ к модулю множества X_1 прямо пропорционально релевантности между структурами, поскольку это отношение определяет степень включения множества X_1, во множество $X_{1,2}$:

$$G_{\mathrm{Rel}}(S_1, S_2) = \frac{|X_{1,2}|}{|X_1|} \cdot \sum_{x_i \in X_{1,2}} \left(\sum_{x_j \in X_2 - X_{1,2}} \mathrm{Rel}(x_i, x_j) + \sum_{x_j \in X_1 - X_{1,2}} \mathrm{Rel}(x_j, x_i) \right). \qquad (21)$$

Рассмотрим возможность применения оценки отношения релевантности при построении интеллектуальных систем.

Отношение релевантности может быть использовано при формализации знаний [6] и в процессе принятия решений [5; 7; 8]. Перечислим некоторые приложения отношения релевантности: прогнозирование изменений в структуре знаний; построение целеустремленных систем; контроль за полнотой знаний; оценка альтернативных путей решения задач; переоценка уже принятых решений и уменьшение вреда от неудачных решений.

Возникновение, устранение и избежание когнитивного диссонанса [9; 10] можно считать механизмами, позволяющими производить переоценку уже принятого решения и уменьшать отрицательные последствия этих решений. Диссонанс возникает между элементами знаний системы, однако диссонанс возможен только между релевантными друг другу элементами. Поэтому, не определив на структуре знаний отношение релевантности, невозможно определить и отношение диссонанса.

Список литературы: 1. *Дюбуа Д., Прад А.* Теория возможностей. Приложение к представлению знаний в информатике: Пер. с фр. М.: Радио и связь, 1990. 286 с. 2. *Нечеткие* множества в моделях управления и искусственного интеллекта / Под ред. Д. А. Поспелова. М.: Наука, 1986. 311 с. 3. *Лефевр В.А.* Конфликтующие структуры. М.: Сов. радио, 1974. 272 с. 4. *Гетманова А.Д.* Логика. М.: Высш. шк., 1986. 287 с. 5. *Гладун В.П.* Планирование решений. К.: Наук. думка, 1987. 168 с. 6. *Нильсон Н.* Искусственный интеллект. Методы поиска решений: Пер. с англ. М.: Радио и связь, 1985. 235 с. 7. *Акофф Р., Эмери Ф.* О целеустремленных системах: Пер. с англ. М.: Сов. радио, 1974. 272 с. 8. *Поспелов Д.А.* Ситуационное управление: теория и практика. М.: Наука, 1986. 197 с. 9. *Аронсон Э.* Теория диссонанса: прогресс и проблемы // Современная зарубежная социальная психология. М., 1984. С. 110–125. 10. *Фестингер Л.* Введение в теорию диссонанса // Современная зарубежная социальная психология. М., 1984. С. 97–110.

Поступила в редколлегию 09.04.98

Проблемы бионики
1998. Вып. 49. С. 196—202

ХТУРЭ

УДК 519.713

О.Ю. ГОЛОБРОДСКИЙ, И.Б. СИРОДЖА

МОДЕЛИ ПРИНЯТИЯ РЕШЕНИЙ НА ОСНОВЕ ИЕРАРХИЧЕСКИХ ВЕРОЯТНОСТНЫХ КВАНТОВ ЗНАНИЙ.

Целями проводимых исследований являются разработка и изучение класса знаниеориентированных моделей, алгоритмов и программных средств принятия решений в условиях неопределенности, пригодных для использования на имеющейся элементной базе, которая способна принимать разношкальные данные (обучаться по ним) и обрабатывать их (манипулировать ими). Особенность предлагаемого подхода заключается в разработке оператора индукции как основного инструмента построения закономерностей данной предметной области (Базы Знаний) в условиях неопределенности. Последняя в данном случае возникает как из-за субъективной оценки эксперта, так и за счет воздействия помех и погрешностей при измерениях числовых значений характеристик объекта принятия решений.

В работе [1] сформулирован и обоснован частный случай формирования знаний (к-знаний) в условиях полной информированности (четких знаний). Основываясь на результатах данной работы, обобщим методы формирования и структурирования знаний в условиях неопределенности, т.е. методику построения иерархических вероятностных квантов знаний (ИВК-знаний) и получения стохастических решений в условиях нечеткой информации на базе использования ИВК-метода. Опишем формально постановку и решение задачи представления знаний KR (Knowledge Representation) в общем виде. Исходные данные об объектах принятия решений (ОПР) из обучающей выборки представляются в виде Таблицы Эмпирических Данных (ТЭД) — Empirical Data Table (EDT). ТЭД содержит описания объектов, сделанные в разнотипных шкалах. В описании любого объекта можно выделить 3 группы признаков:

— группа N — признаки, значения которых измерены в шкале наименований; каждому из признаков поставлен в соответствие коэффициент доверия BdC;

— группа F — признаки, значения которых измерены в абсолютной шкале;

— группа INQ — "справочные" признаки, которые служат только для идентификации объекта, но не могут быть использованы для классификационной обработки.

Очевидно, что признаки группы INQ далее можно не рассматривать. Вполне ясно, что признаки группы N практически не требуют предобработки. Любой из признаков $A^{(j)}$, входящих в группу N, характеризуется набором значений $A^{(j)} = \{a^{(j)}_1, ..., a^{(j)}_k\}$ и соответствующих им коэффициентов доверия $P^{(j)} = \{p^{(j)}_1, ..., p^{(j)}_k\}$. Чтобы построить домен $d^{(j)}_i$ доменизированного вектора $d^{(j)}$ достаточно выполнить оператор векторной конкатенации. Преобразование признаков $A^{(j)}$ группы F в соответствующие домены $X^{(j)}$ доменизированного вектора dV в общем случае зависит от шкалы, в которой проведены измерения. Однако, исходя из того, что нами используется только абсолютная шкала, схема алгоритма предобработки следующая: инженером по знаниям или экспертом в данной предметной области, работающим с программной системой, задается набор интервалов $Int = \{Int^{(j)}_i\}$ для каждого признака $A^{(j)}$. Если для значения $a^{(j)}_k$ признака $A^{(j)}$ справедливо выражение $a^{(j)}_k \in Int^{(j)}_i$, то выполняются следующие действия:

1. Положим $X^{(j)}_i = 1$ и $X^{(j)}_l = 0$, $\forall\, l = 1, 2,..., i\text{-}1, i+1,...,k_j$.

2. Применив оператор векторной конкатенации к полученному домену $X^{(j)}$ с коэффициентом доверия $BdC^{(j)}$, получим доменизированный вектор.

3. Если измерения проведены с пренебрежимо малой погрешностью, то $BdC^{(j)} = 1{,}0$ и нечеткости не возникает.

4. Если установлено, что $(a^{(j)}_k \in Int^{(j)}_i) \oplus (a^{(j)}_k \in Int^{(j)}_r)$, где символ $\oplus$ обозначает операцию "исключающее ИЛИ", то необходимо оценить вероятность того, что $a^{(j)}_k \in Int^{(j)}_i$, т.е. однозначно определить $BdC^{(j)}_i$ и того, что $a^{(j)}_k \in Int^{(j)}_r$, т.е. однозначно определить $BdC^{(j)}_r$.

5. В домене $X^{(j)}$ присвоим $X^{(j)}_i = 1$, $X^{(j)}_r = 1$ и
$X^{(j)}_l = 0$, $\forall\, l = 1, 2,..., i\text{-}1, i+1,...,r\text{-}1,r+1,...,k_j$.

Классом ИВК-знаний назовем алгоритмическую структуру с определенной, наперед заданной семантикой, получаемую путем применения оператора порождения следующего вида:

$$ИВК=MCON\,(VCON\,(\{\,X^{(j)}, BdC^{(j)}\})), \qquad (1)$$

где: MCON, VCON – обозначения операторов матричной и векторной конкатенации; $X^{(j)}$ – конкретное значение j-й характеристики; $BdC^{(j)}$ – вероятность появления события $X^{(j)}$.

Отметим, что ИВК-знания несут в себе реальную информацию о некоторых, возможно даже частных, свойствах ОПР $\omega_i \in \Omega$. Например, ИВК-знания, полученные посредством оператора порождения (1), несут в себе закономерность типа «факт», т.е. отражают единичное эмпирически наблюдаемое явление, либо «запрет», т.е. указывают на невозможность такого явления.

Полученный ИВК-квант охватывает ряд возможных описаний или значений для наблюдаемого объекта или события, то оценкой вероятности $P^{(j)}$ того, что наблюдаемому объекту $\omega_i \in \Omega$ соответствует описание в виде набора значений из ИВК, с учетом измерений значений отдельных характеристик объекта, является коэффициент доверия $BdC^{(j)}$, т.е. $P^{(j)} = BdC^{(j)}$.

ИВК-моделью представления знаний назовем тройку вида

$$\langle \, ИВК\{A^{(j)}\}, Op \, \rangle. \tag{2}$$

Здесь ИВК — некоторый иерархический вероятностный квант знаний 2-го уровня (знания об ОПР в форме 2ИВК), отражающий объективно существующие закономерности исследуемой предметной области; $\{A^{(j)}\}$ — множество характеристик объектов предметной области Ω, свойства которых отражены ИВК-знаниями в форме 2ИВК, т.е. *семантика предметной области*; Ор — множество операций, допустимых в классе ИВК-знаний, т.е. средства манипуляции знаниями, представленными в форме ИВК-квантов 2ИВК.

ИВК-знания 2-го уровня dA называется *общезапретными*, если выполняется следующее условие:

$$\forall i, \forall j, \forall k : i = 1 \ldots \lim i, k = 1 \ldots \lim k, j = 1 \ldots \lim k, dA_j^{(ik)} \equiv 1. \tag{3}$$

Другими словами, квант dA будет общезапретным, если соответствующий ему конечный lim*j*-местный предикат конъюнктного вида тождественно равен 1. Последнее свидетельствует о наличии полностью единичного запретного кванта, покрывающего все остальные запреты, либо о наличии среди квантов, входящих в dA, полной комбинации простых запретных квантов, покрывающих все пространство возможных значений характеристик ОПР $\omega_i \in \Omega$. Исходя из сказанного выше, сформулируем следствие: полностью единичный иерархический вероятностный квант 1-го уровня (квант dA) является общезапретным, т.е.

$$\forall i, \forall k : i = 1 \ldots \lim i, k = 1 \ldots \lim k, dA^{(ik)} \equiv 1 \tag{4}$$

Оператором минимизации иерархического вероятностного кванта знаний 2-го уровня dA (*MINI-оператором*) назовем алгоритм MINIMUM, минимизирующий иерархический вероятностный квант знаний 2-го уровня dA в том смысле, что ни один иерархический вероятностный квант 1-го уровня dA_j из dA не следует ни из какого другого иерархического вероятностного кванта 1-го уровня dA_l из dA. *Оператором редукции (RED-оператором)* назовем реализацию алгоритма REDUCTION выделения из иерархического вероятностного кванта знаний 2-го уровня dA только тех иерархических вероятностных квантов знаний 1-го уровня из dA, которые связаны с иерархическим вероятностным квантом 1-го уровня da, с занесением полученного множества квантов в 2ИВК dA_{red}. Другими словами, задача RED-оператора — локализовать информацию об ОПР, описываемом квантом знаний da. *Оператором дедукции (DED-оператором)* назовем алгоритмическую процедуру DEDUCTION реализации процесса дедуктивного вывода с проверкой на общезапретность и редуцирования Базы Знаний.

Способ перехода от частного к общему, т.е. индуктивное рассуждение, неизбежно применяется в любом процессе принятия решений. В том случае, когда знаниеориентированная система поддержки принятия решений использует заданные экспертом готовые знания и правила, процесс индуктивного вывода проводится человеком. Если же система поддержки принятия решений способна к самообучению, т.е. может обучаться на каких-то эмпирических данных, то индуктивный вывод она проводит самостоятельно. Однако независимо от того, реализует система самообучение или нет, получение общих закономерностей, свойственных предметной области, из частных знаний всегда заложено в любой метод распознавания и классификации. Например, основу методов математической статистики составляет индуктивный принцип равномерной сходимости эмпирического риска к среднему риску. В ИВК-методе поддержки принятия решений *индуктивный принцип* состоит в построении базы закономерностей, свойственных данной предметной области, в виде материальных, импликативных запретных и функциональных связей, которые находятся по обобщениям экспертов и результата самообучения по справочным материалам и экспериментальным данным.

Разберем подробнее принцип индуктивного вывода в ИВК-методе. Пусть в результате опроса экспертов либо после обобщения информации от нескольких датчиков получены знания о некоторой части ξ пространства Ω ОПР $\omega_i \in \Omega$. Назовем ξ обучающей выборкой. Заведомо известно, что мощность множества ξ меньше множества φ допустимых значений характеристик ОПР $\omega_i \in \Omega$. В этой ситуации об общих закономерностях данной предметной области придется судить по немногочисленным выборочным знаниям $\omega_i \in \xi$. Суждения неизбежно будут носить характер гипотез и основываться на поиске запретных комбинаций (связей) между характеристиками ОПР. Такие связи будут равносильны утверждению о невозможности существования ОПР с некоторыми комбинациями свойств. Физической формой данных закономерностей будет являться материальная импликация, в дальнейшем для краткости будем называть эту форму импликативной закономерностью (связью). При усилении импликативной связи область запрета значений $\omega_i \in \xi$ расширяется относительно Ω и возрастает вероятность проявления закономерности, что увеличивает возможность ее обнаружения. Пусть заведомо известно, что некоторые из характеристик ОПР $\omega_i \in \Omega$ связаны между собой импликативными закономерностями разной силы. Тогда с большой долей уверенности можно сказать, что существует по крайней мере одна комбинация значений характеристик ОПР, которая будет запрещена. Значит, существует по крайней мере одна запретная закономерность, описывающая связи высказываниями типа: «Невозможно, чтобы ОПР $\omega_i \in \Omega$, обладающий *к*-м значением *i*-й характеристики с ненулевой вероятностью $P_i^{(ik)} \neq 0$, не обладал *l*-м значением *r*-й характеристики с ненулевой вероятностью $P_i^{(rl)} \neq 0$». Другими словами, в интервал, описываемый данной связью, не входит ни один элемент множества допустимых значений φ и, следовательно, ни один из элементов множества ξ. Таким образом, выдвигается гипотеза о существовании знаний, не пересекающихся с множеством ξ, т.е. о существовании множества τ запретных комбинаций, такого, что $\tau \cap \xi \equiv 0$. Формально выдвинутую гипотезу можно записать в следующем виде:

$$\begin{cases} 1ИВК - dA \cap \xi \equiv 0; \\ 1ИВК - dB \cap \xi \equiv 0, \xi \subseteq \varphi. \end{cases} \tag{5}$$

Возможна ситуация, когда ξ или даже φ вырождается из множества точек в точку. Тогда целесообразно использовать гипотезу о полноте и репрезентативности обучающей выборки ξ. Требование полноты и репрезентативности является справедливым, так как невозможно и нецелесообразно проводить обучение на примерах, неполно или неточно характеризующих область обучения. Большинство современных экспертных систем опирается на указанную гипотезу и полностью полагается на сообщенные им факты. Использовав данную гипотезу, сформулируем следующий *постулат*: существует ненулевая вероятность того, что в созданном множестве запретных закономерностей находятся знания, отражаемые закономерностями любого ранга. Значит, всегда имеется шанс обнаружить в описании ОПР в виде иерархического вероятностного кванта 2-го ранга интересующую нас закономерность и есть смысл проводить дальнейшие исследования.

Наряду с оператором индуктивного вывода вторым важнейшим моментом ИВК-метода является оператор дедуктивного вывода знаний, реализующий процесс распознавания и классификации, что в пространстве ОПР Ω означает определение неизвестных составляющих ИВК-вектора, представляющего исследуемый ОПР $\omega_i \in \Omega$, по его известным составляющим и общим знаниям о нем и о предметной области. Рассмотрим *дедуктивный принцип* ИВК-метода как процесс дедуктивного вывода ИВК-знаний, который в широком смысле можно понимать как принятие решения о значениях неизвестных характеристик ОПР исходя из частных (типа «факт», «запрет») и общих (иерархические вероятностные кванты 2-го уровня) знаний. Если рассмотреть традиционные модели представления знаний (фреймы, продукционные правила, семантические сети, логическая модель), то процесс принятия решения можно в общем виде описать следующим образом:

1. Составляется множество исходных ситуаций.

2. Формулируются все правила (узлы сети, заполнитель слотов фрейма и т.д.), относящиеся к указанному множеству ситуаций (т.е. правила, применимые в данных ситуациях).

3. Составляется множество следствий.

4. Шаги 1—3 повторяются циклически до тех пор, пока все из возможных применимых правил не будут найдены.

Зачастую процесс построения Базы Знаний требует большого количества времени и вычислительных ресурсов из-за многогранности и сложности заданной предметной области. Поэтому проблема минимизации объема Базы Знаний без потери информации достаточно остра. В терминах ИВК-метода под *минимизацией* Базы Знаний, представленной иерархическим вероятностным квантом знаний 2-го уровня, понимается приведение вероятностного кванта знаний в такое состояние, когда ни один из иерархических вероятностных квантов знаний 1-го уровня, входящих в Базу Знаний, логически не следует из любого другого кванта 1-го уровня, входящего в Базу Знаний [2].

Таким образом, получены следующие результаты:

1. Описан и обоснован класс ИВК-знаний как совокупность специальным образом структурированных данных.

2. Введено понятие ИВК-модели представления знаний и манипулирования ими, показаны механизм порождения ИВК-знаний и основные операции над ними.

3. Сформулирован индуктивный принцип построения Базы Знаний на основе применения ИВК-квантов, состоящий в необходимости выполнения заранее заданного критерия репрезентативности для индуктивно выводимых ИВК-знаний.

4. Сформулирован дедуктивный принцип ИВК-метода, заключающийся в процессе принятия решения о значениях неизвестных характеристик ОПР исходя из общих и частных знаний о нем и о заданной предметной области.

Список литературы: 1. *Сироджа И.Б.* Математическое и программное обеспечение интеллектуальных компьютерных систем. Х.: Харьк. авиац. ин-т, 1993. 101 с. 2. Golobrodsky O. Knowledgebase storage simplification and minimization and traduction conclusion drawing algorithm realisation using Method of multilevel algorihmical quanta // Proceedings. Signal and image processing.UKRObraz'94. Киев, 1994. С. 13—21.

Поступила в редколлегию 23.04.98

Проблемы бионики
1998. Вып. 49. С. 203—211

ХТУРЭ

УДК 371.385:681.3

Н.В. БЕЛОУС, А.П. ВЫРОДОВ, И.Ю.ШУБИН

МАТЕМАТИЧЕСКИЕ МОДЕЛИ ПОСТРОЕНИЯ ВИРТУАЛЬНЫХ ПРОСТРАНСТВ

Построение пространственных графических изображений, синтез звуков, обработка больших объемов информации в реальном масштабе времени — все эти возможности, появившиеся благодаря интенсивному развитию вычислительной техники, находят применение практически во всех отраслях человеческой деятельности. Системы построения виртуальных пространств уже используются при компьютерных презентациях, создании компьютерных обучающих систем и трехмерных Web-страниц (подробную информацию о последних можно найти в [1; 2]). Кроме того, следует учесть, что интерфейс операционных систем 2000 года будет также трехмерным. Все это вызывает повышенный интерес как к трехмерной компьютерной графике в целом, так и к способам построения виртуальных пространств в частности.

В [3] рассмотрено несколько различных способов построения виртуальных пространств, включая применение интерактивных средств разработки, а также приведена общая схема оптимизированного алгоритма трассировки луча, который является ядром системы проектирования виртуальных пространств. В данной статье предложена математическая модель оптимизированного алгоритма трассировки луча, рассмотрены возникающие при ее использовании искажения и указан способ их устранения. Кроме того, изложен метод программной оптимизации, позволяющий существенно повысить скорость выполнения программы, реализующей данный алгоритм.

Математические основы оптимизированного алгоритма трассировки луча. В оптимизированном алгоритме трассировки луча для определения положения точки пересечения текущего луча с вертикальной стороной пересеченного квадрата используется функция тангенс от текущего угла просмотра. Другими словами, вычисляется угловой коэффициент трассируемого луча. Обозначим угол наклона текущего луча через α, тогда угловой коэффициент K определяется по формуле

$$K = \operatorname{tg} \alpha = a / b, \qquad (1)$$

где a — противоположный катет; b — прилегающий катет.

Так как используется графический режим 320 х 200 и угол обзора в 60°, угол наклона лучей может меняться не непрерывно, а дискретно с шагом в 60/320 = 0,1875°. Чтобы повысить производительность, необходимо заранее просчитать значения тангенса для всех возможных углов наклона и занести их в таблицу. После этого для получения требуемого значения тангенса достаточно извлечь его из таблицы, что оказывается гораздо быстрее непосредственного вычисления арифметическим сопроцессором. Полученная таким образом таблица будет содержать 360/0,1875 = 1920 элементов. Как известно, функция тангенс стремится к бесконечности при значениях аргумента в 90° и 270°. Поэтому надо ввести дополнительные проверки угла наклона, чтобы избежать в указанных точках ошибки деления на нуль.

Остановимся подробнее на вычислении координат точки пересечения трассируемого луча со стороной первого встретившегося на пути квадрата. Через X_1 обозначим абсциссу точки пересечения луча с горизонтальной стороной квадрата, через Y_1 — ординату точки пересечения луча с вертикальной стороной квадрата, а через X и Y — текущие координаты пользователя. Тогда уравнение, описывающее трассируемый луч (уравнение прямой с угловым коэффициентом), представится в виде

$$(Y_1 - Y) / (X_1 - X) = K\,, \tag{2}$$

откуда

$$Y_1 = K(X_1 - X) + Y; \tag{3}$$

$$X_1 = K^{-1}(Y_1 - Y) + X. \tag{4}$$

Отметим, что каждое из полученных преобразований требует предварительного вычисления другого. Чтобы избежать такой зависимости, нужно в формулу (3) вместо абсциссы X_1 подставить абсциссу любой другой точки, лежащей на трассируемом луче (рис. 1). В качестве абсциссы удобно взять абсциссу первой граничной вертикальной линии X'. Аналогично, вместо Y_1 в формулу (4) подставим ординату Y' первой граничной горизонтальной линии (рис. 1). Теперь формулы (3) и (4) можно переписать в виде

$$Y_1 = K(X' - X) + Y; \tag{3a}$$

$$X_1 = K^{-1}(Y' - Y) + X. \tag{4a}$$

Формулы (3а) и (4а) полностью подходят для проведения вычислений, поскольку координаты X' и Y' могут быть легко определены исходя из следующих соображений.

Пусть координаты пользователя X и Y могут изменяться от 0 до 1023 и нужно найти абсциссу X' первой граничной вертикальной линии (см. рис. 1). Для этого разделим нацело координату X пользователя на размер стороны квадрата, который положим равным 64 единицам. Полученная координата представляет собой абсциссу левой граничной вертикальной линии, а так как на рис. 1 вектор направления взгляда указывает вправо, необходимо увеличить найденную координату на размер стороны квадрата. Если бы вектор направления взгляда указывал влево, то необходимо было бы уменьшить эту координату на размер стороны квадрата. Аналогично находится ордината Y' первой граничной горизонтальной линии.

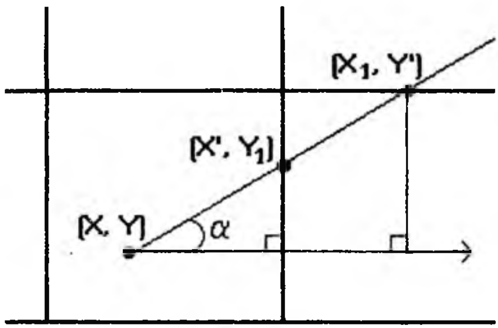


Рис. 1

Заметим, что в формулу (4а) коэффициент K входит в степени -1, а значит, требуется еще одна таблица — тангенсов всех углов в степени -1, т.е. таблица котангенсов всех углов.

После вычисления координат точки первого пересечения переходим к вычислению координат следующего возможного пересечения. Так как ширина каждого квадрата фиксирована (в данном случае — 64 единицы), определяем по формуле (6) следующую координату Y точки, с которой пересечется трассируемый луч, если увеличить координату X на 64. Тем самым сразу пропускаем 64 точки карты, поскольку из ее построения следует, что там стена находиться не может. Аналогично, увеличивая координату Y на 64, по рекуррентной формуле (5) определяем координату X следующего возможного пересечения. Итак, координаты точек возможного пересечения, начиная со второй, вычисляются по следующим рекуррентным формулам:

$$X_{i+1} = X_i + K^{-1}c; \qquad (5)$$

$$Y_{i+1} = Y_i + Kc, \qquad (6)$$

где с — сторона квадрата.

Благодаря тому что сторона квадрата имеет фиксированный размер в 64 единицы, становится возможным составление еще двух таблиц — результатов умножения значений тангенса и котангенса на 64, что позволит существенно повысить скорость выполнения программы.

Как только найдена X-стена, следует использовать координату Y, чтобы установить, какую колонку стены надо прорисовать, и координату X, чтобы определить расстояние до стены. Аналогично, если найдена Y-стена, то следует использовать координату X, чтобы установить, какую колонку стены надо прорисовать, и координату Y, чтобы определить расстояние до стены. Действительно, расстояние от точки с координатами (X,Y) до точки с координатами (X',Y_1) или от точки (X,Y) до точки (X_1,Y') (см. рис. 1) можно отыскать двумя способами: 1) по теореме Пифагора, использовав функцию извлечения корня; 2) с помощью известных соотношений между сторонами и тригонометрическими функциями углов прямоугольного треугольника. Как было указано в [3], извлечение корня неэффективно даже при предварительном составлении таблицы всех возможных значений корня. Поэтому целесообразно использовать тригонометрические функции от угла наклона α трассируемого луча. Расстояние до точки пересечения трассируемого луча с горизонтальной стороной квадрата
вычисляется по формуле

$$Dx = (X_i - X)\cos^{-1}\alpha, \tag{7}$$

где Dx — расстояние до точки пересечения трассируемого луча с горизонтальной стороной квадрата.

Расстояние до точки пересечения трассируемого луча с вертикальной стороной квадрата находится следующим образом:

$$Dy = (Y_i - Y)\sin^{-1}\alpha, \tag{8}$$

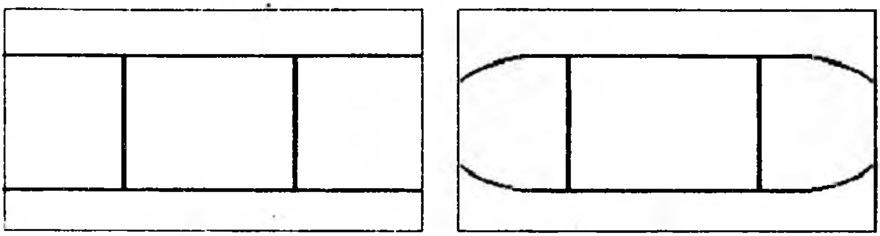
где Dy — расстояние до точки пересечения трассируемого луча с вертикальной стороной квадрата.

Угол α в формулах (7) и (8) является углом между трассируемым лучом и вектором направления взгляда. В программной реализации данного алгоритма угол α — это индекс, изменяющийся в диапазоне от 0 до 1920, для таблиц заранее вычисленных значений функций $\sin^{-1}\alpha$, $\cos^{-1}\alpha$, $\mathrm{tg}\,\alpha$, $\mathrm{ctg}\,\alpha$, $64\mathrm{tg}\,\alpha$, $64\mathrm{ctg}\,\alpha$.

Проекционные искажения. Алгоритм трассировки луча использует одновременно полярные и декартовы системы координат. Как следствие, это привело к появлению так называемых сферических искажений.

Действительно, все трассируемые лучи выходят из одной точки, и эквидистантные поверхности представляют собой концентрические сферы. Поэтому, чтобы построить стену, расположенную перпендикулярно к вектору направления взгляда, надо на карте, как на виде сверху, изобразить стену не прямой линией, а дугой окружности. Только в этом случае построенная стена будет выглядеть так, как на рис. 2, *а*.

При подготовке карты стены условно изображались в виде отрезков прямых линий, и расстояние до середины стены, расположенной перпендикулярно к вектору направления взгляда, всегда меньше расстояний до ее краев. Следовательно, высота колонки текстуры данной стены максимальна в ее середине и постепенно уменьшается при приближении к краям. Поэтому нескомпенсированное изображение на экране дисплея имеет вид, показанный на рис. 2, *б*.



а *б*

Рис. 2

Сферические искажения являются синусоидальными, и для их компенсации необходимо умножить высоту колонки стены на косинус текущего угла просмотра, т.е. высоту колонки стены надо вычислять по формуле

$$Ch = (\cos \alpha)20\,000 / D, \qquad (9)$$

где Ch — высота колонки стены; D — расстояние до стены.

Реализация открывающихся дверей. Открывающиеся двери реализованы следующим образом: если в процессе трассировки луч достигает двери, то трассировка луча продолжается дальше до пересечения со стеной или до выхода за пределы карты виртуального пространства. При поступлении от пользователя команды "открыть дверь" в специальном буфере выполняется прорисовка области, заслоненной закрытой дверью. Одновременно осуществляется сдвиг изображения двери влево (дверь убирается в косяк), а на освободившееся место выводится содержимое буфера. Это создает эффект открывания двери. Аналогичным образом возможна реализация любых полностью или частично прозрачных плоских областей (более подробная информация о реализации таких областей находится в [4]).

Использование формата данных с фиксированной запятой. Чтобы достичь значительного повышения производительности, надо заменить операции над числами с плавающей запятой на операции над числами с фиксированной запятой. Дело в том, что числа с плавающей запятой хранятся в специальном формате, в котором мантисса и порядок представлены в зашифрованном виде, поэтому перед использованием таких чисел их надо расшифровать и нормализовать. В силу этих обстоятельств на компьютере, оснащенном математическим сопроцессором, операции над числами с плавающей запятой требуют в 2 раза больше времени, чем над числами с фиксированной запятой. Поэтому настоятельно рекомендуется использовать последние числа.

Для программной реализации алгебры чисел с фиксированной запятой необходимо знать, как выполняются основные операции над такими числами. Перейдем к рассмотрению этих операций.

Желательно использовать 32-разрядный формат данных и для дробной части выделить только 8 младших разрядов. Так как для программ построения виртуальных пространств не нужна очень высокая точность (5 и более цифр после запятой), то указанных 8 разрядов вполне достаточно. Действительно, наименьшее число, которое можно записать в таком формате, равно 0,004 и в большинстве случаев погрешности по абсолютному значению не выходят за пределы диапазона 0,01 — 0,5, что вполне допустимо, поскольку 90 % всех расчетов в программе направлено на определение местоположения пикселей на экране и, следовательно, результаты округляются до ближайшего целого. Операция присваивания выполняется различно для целой и дробной частей числа с фиксированной запятой. Для присваивания целой части надо предварительно умножить присваиваемое значение на 256 или сдвинуть его на 8 разрядов влево, чтобы оставить место для дробной части. Покажем, как это выглядит на языке программирования Borland C++:

```
int i = 500 ;    long fixd = 0 ;
fixd = ( ( long ) i << 8 ) ;
```

Для присваивания дробных чисел необходимо произвести умножение чисел с плавающей точкой и записать результат в выделенные 32 разряда (в Borland C++ соответствующий формат называется long):

```
long  fixd = ( long )( 23.45 * 256 );
```

Здесь использовалась операция умножения на 256 вместо операции сдвига, поскольку последняя неприменима к числам с плавающей запятой.

Операции сложения и вычитания над числами с фиксированной запятой выполняются как и над целыми числами:

fixd3 = fixd1 + fixd2; fixd3 = fixd1 - fixd2;

Вычитание и использование отрицательных чисел в Borland C++ возможны благодаря тому, что внутреннее представление чисел в формате long учитывает знак. Таким образом, и в данном 32- разрядном формате необходимо 1 бит выделить под знак. Под целую часть было выделено 32 – 8 = 24 разряда, и модуль максимального числа, которое может участвовать в операциях сложения и вычитания, равен $2^{24-1} - 1 = 8\ 388\ 607$.

Операция умножения над числами с фиксированной запятой выполняется как и над целыми числами, за исключением того, что результат должен быть сдвинут на 8 разрядов вправо:

fixd1 = (fixd2 fixd3) >> 8;

Это нужно делать по следующей причине: когда инициализируется число с фиксированной точкой (выполняется операция присваивания), то оно умножается на 256 (см. выше); при умножении целая часть первого числа, умноженная на 256, умножается на целую часть второго числа, также умноженную на 256, поэтому результат оказывается дважды умноженным на 256. По этой же причине максимальное число с фиксированной точкой, которое может участвовать в операции умножения, равно 181, так как оно, будучи умноженным на 256 и возведенным в квадрат, может быть записано в выделенном для него 31 разряде.

Операцию деления целесообразно производить как операцию умножения делимого на величину, обратную делителю:

fixd1 = (long) (256 * 1/34) ;

fixd2 = (fixd3 fixd1) >> 8 ;

Пример изображения виртуального пространства, формируемого программой на экране дисплея, представлен на рис. 3. Данное трехмерное изображение получено при расположении пользователя, показанном на рис. 4.

Рис. 3

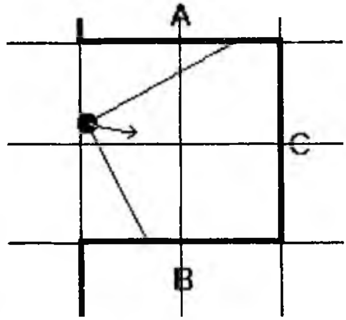


Рис. 4

Здесь сегментам стен А и В поставлена в соответствие текстура, приведенная на рис. 5, а сегменту стен С — текстура на рис. 6.

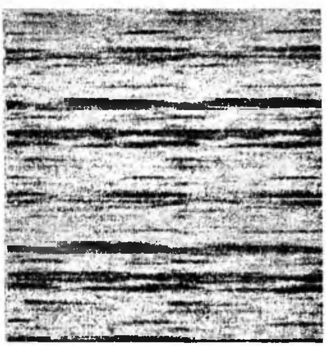


Рис. 5

Рис. 6

В заключение отметим, что в программной реализации описанного алгоритма, выполненной на языке программирования Borland C++ 4.2, используется несколько форматов чисел с фиксированной запятой, но все операции над ними производятся по изложенным выше правилам. Полученная программа при предъявлении минимальных на сегодняшний день требований к ресурсам вычислительной системы (450 Кбайт оперативной памяти, 1 Мбайт пространства на жестком диске, i286-микропроцессор) имеет следующие особенности:

— использование стандартного VGA режима 320x200 с 256 цветами, что обеспечивает практически 100 %-ю переносимость программы;

— частоту смены кадров 30 кадров в секунду;

— возможность разбиения виртуального пространства, превышающего объем свободной оперативной памяти, на несколько частей, загружаемых в заданной последовательности;

— возможность наполнения виртуального пространства неподвижными объектами произвольной формы;

— реализацию открывающихся дверей и возможность работы с анимированными текстурами;

— поддержку 256 цветных спрайтов в форматах PCX и LBM.

Список литературы: 1. *Трехмерность* в World Wide Web // Компьютер Пресс. 1997. № 3. С. 84–93. 2. *Татарников О.* VRML2.0: Виртуальная нереальность // Там же. С. 67–59. 3. *Белоус Н.В., Выродов А.П., Шубин И.Ю.* О некоторых алгоритмах построения виртуальных пространств // Проблемы бионики. 1998. № 48. С. 52–59. 4. *Секреты* программирования игр: Пер. с англ. / *А. Ла Мот, Дж. Ратклифф, М. Семинаторе, Д. Тайлер.* СПб.: Питер, 1995. 616 с.

Поступила в редколлегию 09.04.98

АВТОРЫ ВЫПУСКА

Андриани Ольга Вячеславовна, асп., Харьковский государственный технический университет радиоэлектроники (ХТУРЭ)
Белоус Наталья Валентиновна, канд. техн. наук, доц., ХТУРЭ
Бодянский Евгений Владимирович, канд. техн. наук, проф., ХТУРЭ
Вороной Максим Филиппович, асп., ХТУРЭ
Выродов Александр Петрович, студент, ХТУРЭ
Гацкий Михаил Юрьевич, системный администратор фирмы "СПИН", г. Харьков
Гвоздинская Наталья Анатольевна, асп., ХТУРЭ
Гевчук Дмитрий Александрович, студент, ХТУРЭ
Герасин Сергей Николаевич, канд. техн. наук, доц., ХТУРЭ
Голобродский Олег Юрьевич, асп., Харьковский авиационный университет
Дидык Алексей Александрович, асп., Херсонский государственный технический университет
Дударь Зоя Владимировна, канд. техн. наук, доц., ХТУРЭ
Ерохин Андрей Леонидович, канд. техн. наук, доц., Харьковский университет внутренних дел
Журавок Елена Владимировна, мл. науч. сотр., ХТУРЭ
Коротин Константин Ефимович, инж., Институт проблем машиностроения НАН Украины (ИПМАШ), г. Харьков
Кравец Наталья Сергеевна, асп., ХТУРЭ
Лебедев Олег Григорьевич, канд. техн. наук, доц., Харьковский институт летчиков ВВС Украины
Мартынов Анатолий Никифорович, доц., Херсонский государственный технический университет
Маторин Сергей Игоревич, канд. техн. наук, ст. науч. сотр., ХТУРЭ
Митюшкин Юрий Игоревич, магистрант, Винницкий государственный технический университет
Новожилова Марина Владимировна, канд. физ.-мат. наук, ст. науч. сотр., ИПМАШ, г. Харьков
Пославский Сергей Александрович, канд. физ.-мат. наук, доц., Харьковский государственный университет
Походенко Виталина Алексеевна, асп., ХТУРЭ
Рассадникова Анна Владимировна, асп., ХТУРЭ
Ронин Евгений Маркович, студент, ХТУРЭ
Ротштейн Александр Петрович, д-р техн. наук, проф., Винницкий государственный технический университет
Самофалов Павел Леонидович, асп., ХТУРЭ
Сироджа Игорь Борисович, д-р техн. наук, проф., Харьковский авиационный университет
Степанский Константин Григорьевич, асп., Херсонский государственный технический университет
Тищенко Владимир Владимирович, канд. техн. наук, доц., ХТУРЭ

Тоница Олег Владимирович, мл. науч. сотр., ИПМАШ, г. Харьков
Чикина Валентина Алексеевна, канд. техн. наук, ведущий науч. сотр., ХТУРЭ
Числин Николай Иванович, инж., ХТУРЭ
Ходаков Виктор Егорович, канд. техн. наук, проф., Херсонский государственный технический университет
Шабанов-Кушнаренко Сергей Юрьевич, д-р техн. наук, ведущий науч. сотр., ХТУРЭ
Шабанов-Кушнаренко Юрий Петрович, засл. деят. науки и техники Украины, д-р техн. наук, проф., ХТУРЭ
Шевченко Александр Николаевич, д-р физ.-мат. наук, проф., ИПМАШ, г. Харьков
Шерстюк Владимир Григорьевич, канд. техн. наук, Херсонский государственный технический университет
Шубин Игорь Юрьевич, канд. техн. наук, доц., ХТУРЭ

СОДЕРЖАНИЕ

CONTENTS

УДК 519.7

О фундаментальной алгебре предикатных операций / З.В. Дударь, Н.С. Кравец, Ю.П. Шабанов-Кушнаренко // Проблемы бионики. 1998. Вып. 49. С. 3—13.

С целью формального описания объектов и процессов, наблюдаемых в субъективном мире человека и машины, разрабатываются алгебры предикатов и предикатных операций. Доказаны теоремы о неполноте алгебры предикатных операций с константами и переменными, о полноте алгебры булевых функций, о полноте дизъюнктивно-конъюктивной алгебры предикатных операций, о полноте фундаментальной алгебры. о полноте системы основных тождеств фундаментальной алгебры.

Табл. 1. Библиогр.: 4 назв.

УДК 519.7

Про фундаментальну алгебру предикатних операцій / З.В. Дударь, Н.С. Кравець, Ю.П. Шабанов-Кушнаренко // Проблеми біоніки. 1998. Вип. 49. С. 3—13.

З метою формального опису об'єктів і процесів у суб'єктивному світі людини і комп'ютера розробляються алгебри предикатів і предикатних операцій. Довсдено теореми про неповноту алгебри предикатних операцій з константами і змінними, про повноту алгебри булевих функцій, про повноту диз'юнктивно-кон'юнктивної алгебри предикатних операцій, про повноту фундаментальної алгебри, про повноту системи основних тотожностей фундаментальної алгебри.

Табл. 1. Бібліогр.: 4 назви.

UDC 519.7

On fundamental algebra of predicate operations / Z.V. Dudar, N.S. Kravec, Yu.P. Shabanov-Kushnarenko // Problemy Bioniki. 1998. N 49. P. 3—13.

With the purpose of the formal description of objects and processes noticed in the subjective world of man and computer the algebras of predicates and of predicate operations are developed. The theorems are proved of incompletness of algebra of predicate operations with constants and variables, about completeness of Boolean functions algebra, about completeness of disjunction-conjunction algebra of predicate operations, about completeness of fundamental algebra, about completeness of system of the basic identities of fundamental algebra.

1 tab. Ref.: 4 titles.

УДК 519.7

О прикладной алгебре предикатных операций / З.В. Дударь, Н.С. Кравец, Ю.П. Шабанов-Кушнаренко // Проблемы бионики. 1998. Вып. 49. С. 14—22.

Рассматривается задача разработки алгебр предикатных операций, более удобных практически для приложений, чем фундаментальная алгебра. На базе аппарата фундаментальной алгебры изучены операции над предикатами для построения более практичной алгебры предикатных операций — алгебры подстановочных операций, или прикладной алгебры.

Библиогр.: 3 назв.

УДК 519.7

Про прикладну алгебру предикатних операцій / З.В. Дударь, Н.С. Кравець, Ю.П. Шабанов-Кушнаренко // Проблеми біоніки. 1998. Вип. 49. С. 14—22.

Розглядається задача розробки алгебр предикатних операцій, більш зручних для практичного застосування, ніж фундаментальна алгебра. На базі апарату фундаментальної алгебри вивчено операції над предикатами для побудови більш практичної алгебри предикатних операцій — алгебри підстановчих операцій, або прикладної алгебри.

Бібліогр.: 3 назв.

UDC 519.7

On applied algebra of predicate operations / Z.V. Dudar, N.S. Kravec, Yu.P. Shabanov-Kushnarenko // Problemy Bioniki. 1998. N 49. P. 14—22.

The task of development of predicate operations algebras, which are more convenient than the fundamental algebra for practical applications, is considered. Operations over predicates for constructing predicate operations algebras, substitution operation algebras or applied algebras are investigated using the apparatus of the fundamental algebra.

Ref.: 3 titles.

УДК 681.513.7

Обнаружение разладок в нелинейных стохастических последовательностях с помощью рекуррентных искусственных нейронных сетей / Е.В. Бодянский // Проблемы бионики. 1998. Вып. 49. С. 23—33.

Предложен подход к решению задачи обнаружения изменения свойств стохастических последовательностей, описываемых нелинейными уравнениями авторегрессии – скользящего среднего. Предполагается, что изменение свойств последовательности может иметь вид параметрической или структурной нестационарности (изменение порядка). Описаны архитектура многослойной рекуррентной нейронной сети и алгоритмы настройки нейронов, обеспечивающие максимальное быстродействие процесса обучения и высокое качество прогнозирования. Достоинствами подхода являются возможность диагностирования и предсказания стохастических последовательностей произвольной структуры, высокое быстродействие и вычислительная простота.

Ил. 2. Библиогр.: 23 назв.

УДК 681.513.7

Виявлення розладнань у нелінійних стохастичних послідовностях за допомогою рекурентних штучних нейронних мереж / Є.В. Бодянський // Проблеми біоніки. 1998. Вип. 49. С. 23—33.

Запропоновано підхід до вирішення задачі виявлення зміни властивостей стохастичних послідовностей, що описуються нелінійними рівняннями авторегресії – ковзного середнього. Припускається, що зміна властивостей послідовності може мати вигляд параметричної або структурної нестаціонарності (зміна порядку). Описано архітектуру багатошарової рекурентної нейронної мережі та алгоритми настроювання нейронів, що забезпечують максимальну швидкодію процесу навчання і високу якість прогнозування. Достоїнствами підходу є можливість діагностування і передбачення стохастичних послідовностей довільної структури, висока швидкодія та обчислювальна простота.

Іл. 2. Бібліогр.: 23 назви.

UDC 681.513.7

Fault detection in nonlinear stochastic sequences based on recurrent artificial neural networks / Ye.V. Bodyanskiy // Problemy Bioniki. 1998. N 49. P. 23—33.

The approach is proposed to solving the problem of properties change detection in stochastic sequences, that are described by nonlinear autoregression — sliding average equations. It is assumed that changes of properties may have a form of both parametrical and structural nonstationarity (change of order). Architecture of multilayer recurrent neural network and neurons parameters tuning algorithms that provide the maximal rate of learning processes and high quality of forecasting are proposed. Advantage of this approach lies in the possibility of diagnosting and forecasting arbitrary structure stochastic sequences, high rate and computational simplicity.

2 fig. Ref.: 23 titles.

УДК 681.513.7

Автоматическое обнаружение разладок с помощью искусственной нейронной метасети / Е.В. Бодянский // Проблемы бионики. 1998. Вып. 49. С. 34—38.

Предложен более эффективный подход к решению задачи обнаружения изменения свойств стохастических последовательностей, описываемых нелинейными уравнениями авторегрессии – скользящего среднего. Представлены архитектура искусственной нейронной метасети и алгоритм настройки выходного нейрона. Достоинствами подхода являются высокое быстродействие и вычислительная простота.

Ил. 1. Библиогр.: 4 назв.

УДК 681.513.7

Автоматичне виявлення розладнань за допомогою штучної нейронної метамережі / Є.В. Бодянський // Проблеми біоніки. 1998. Вип. 49. С. 34—38.

Запропоновано більш ефективний підхід до вирішення задачі виявлення зміни властивостей стохастичних послідовностей, що описуються нелінійними рівняннями авторегресії – ковзного середнього. Подано архітектуру штучної нейронної метамережі та алгоритм настроювання вихідного нейрона. Достоїнствами підходу є висока швидкодія та обчислювальна простота.

Іл. 1. Бібліогр.: 4 назви.

UDC 681.513.7

Automatic fault detection based on artificial neural metanetwork / Ye.V. Bodyanskiy // Problemy Bioniki. 1998. N 49. P. 34—38.

The approach is proposed to solving the problem of properties change detection in stochastic sequences, that are described by nonlinear autoregression — sliding average equations. The architecture of artificial neural metanetwork and output neuron tuning algorithm is proposed. Advantage of this approach is high rate and computational simplicity.

1 fig. Ref.: 4 titles

УДК 519.7

Условия существования линейного предиката и его содержательная интерпретация / С. А. Пославский, В.А. Походенко, С.Ю. Шабанов-Кушнаренко // Проблемы бионики. 1998. Вып. 49. С. 39—45.

Рассмотрена полная система признаков линейной конечномерной операции, идентифицируемой компараторным методом. Если предикат, реализуемый системой компараторной идентификации объекта, обладает указанными свойствами, то объект идентификации можно описать в виде линейного оператора. Рассмотрен способ практической проверки характеристических свойств линейного объекта.

Библиогр.: 9 назв.

УДК 519.7

Умови існування лінійного предиката та його змістовна інтерпретація / С. А. Пославський, В.А. Походенко, С.Ю. Шабанов-Кушнаренко // Проблеми біоніки. 1998. Вип. 49. С. 39—45.

Розглянуто повну систему ознак лінійної скінченномірної операції, що ідентифікується компараторним методом. Якщо предикат, що реалізується системою компараторної ідентифікації об'єкта, має вказані властивості, то об'єкт ідентифікації можна описати у вигляді лінійного оператора. Викладено спосіб практичної перевірки характеристичних властивостей лінійного об'єкта.

Бібліогр.: 9 назв.

UDC 519.7

Conditions of existence of a linear predicate and its meaningful interpretation / S.A. Poslavsky, V.A. Pohodenko, S.Yu. Shabanov-Kushnarenko // Problemy Bioniki. 1998. N 49. P. 39—45.

The complete system of attributes of a linear finite-dimension operation identified by a Comparator method is considered. If the predicate realized by a comparator system of identification of an object has the specified properties, the object of identification can be described as a linear operator. The way of practical checking of characteristic properties of linear objects is considered.

Ref.: 9 titles.

УДК 519.7

Идентификация объектов, описываемых векторами / О.Г. Лебедев, В.А. Походенко, С.Ю. Шабанов-Кушнаренко // Проблемы бионики. 1998. Вып. 49. С. 46—53.

Многие идентифицируемые объекты удобно описывать в виде векторов, т.е. наборов отдельных компонентов (признаков). В статье ставится задача отыскания таких полных систем свойств, с помощью которых можно было бы осуществлять компараторную идентификацию объектов, подпадающих под понятие линейного пространства и под понятие линейной операции над векторами линейных пространств.

Ил. 1. Библиогр.: 8 назв.

УДК 519.7

Ідентифікація об'єктів, що описуються векторами / О.Г. Лебедев, В.А. Походенко, С.Ю. Шабанов-Кушнаренко // Проблеми біоніки. 1998. Вип. 49. С. 46—53.

Багато об'єктів, що ідентифікуються, зручно описувати у вигляді векторів, тобто наборів окремих компонентів (ознак). У статті ставиться задача відшукування таких повних систем властивостей, за допомогою яких можна було б здійснити компараторну ідентифікацію об'єктів, що підпадають під поняття лінійного простору та під поняття лінійної операції над векторами лінійних просторів.

Іл. 1. Бібліогр.: 8 назв.

UDC 519.7

Identification of objects described by vectors / O.G. Lebedev, V.A. Pohodenko, S.Yu. Shabanov-Kushnarenko // Problemy Bioniki. 1998. N 49. P. 46—53.

Many identified objects are convenient for describing as vectors, i.e. sets of separate components (attributes). The task of searching such complete systems of properties is set, with whose help it would be possible to carry out comparators identification of objects falling under the concept of linear space and under the concept of linear operation over vectors of linear spaces.

1 fig. Ref.: 8 titles.

УДК 519.711.3

Компьютерний грамматический словарь / О.В. Андриани, Е.М. Ронин, В.А. Чикина // Проблемы бионики. 1998. Вып. 49. С. 54—58.

Описан программный комплекс "Компьютерный грамматический словарь", применяемый для осуществления словоизменительного анализа слов русского языка. Комплекс состоит из словаря и программы управления базой данных, которая осуществляет автоматизированное пополнение словаря, хранение, анализ и обработку информации.

Табл. 4. Ил. 2. Библиогр.: 3 назв.

УДК 519.711.3

Комп'ютерний граматичний словник / О.В. Андріані, Є.М. Ронін, В.О. Чікіна // Проблеми біоніки. 1998. Вип. 49. С. 54—58.

Описано програмний комплекс "Комп'ютерний граматичний словник", який здатен виконувати словозміновий аналіз слів російської мови. Комплекс складається із словника та програми, що керує базою даних і здійснює автоматизоване поповнення словника, зберігання, аналіз та обробку інформації.

Табл. 4. Іл. 2. Бібліогр.: 3 назви.

UDC 519.711.3

Computer grammatical dictionary / O.V. Andriani, E.M. Ronin, V.A. Chikina // Problemy Bioniki. 1998. N 49. P. 54—58.

The present article describes the program complex called Computer Grammatical Dictionary. This program is used for execution word-changing analysis of Russian words. The Computer Grammatical Dictionary consist of the glossary and database-management program, which provides the automatic updating of the database, storage, changing and processing of information within the dictionary.

4 tab. 2 fig. Ref.: 3 titles.

УДК 519.7

Схемная реализация некоторых морфологических отношений / В.А. Чикина // Проблемы бионики. 1998. Вып. 49. С. 59—66.

Изложен аппаратурный метод решения логических уравнений. В качестве математического аппарата использована алгебра конечных предикатов. Математически описаны и схемно реализованы связи между переменными — буквами окончаний полных непритяжательных имен прилагательных.

Табл. 2. Ил. 6. Библиогр.: 2 назв.

УДК 519.7

Схемна реалізація деяких морфологічних відношень / В.О. Чікіна // Проблеми біоніки. 1998. Вип. 49. С. 59—66.

Викладено апаратурний метод розв'язання логічних рівнянь. Як математичний апарат використано алгебру скінченних предикатів. Математично описано та схемно реалізовано зв'язки між змінними — літерами закінчень повних неприсвійних прикметників.

Табл. 2. Іл. 6. Бібліогр.: 2 назви.

UDC 519.7

Circuit realization of some morphological relations / V.A. Chikina // Problemy Bioniki. 1998. N 49. P. 59—66.

Hardware method of solution of logical equations has been described. Algebra of finite predicates has been used as mathematical apparatus. Connections between variable quantities — letters endings of full nonpossessive adjectives have been mathematically described and schematically realized.

2 tab. 6 fig. Ref.: 2 titles.

УДК 681.3:621.311

Распознавание аварийных ситуаций в энергосистемах / А.Л. Ерохин // Проблемы бионики. 1998. Вып. 49. С. 67—71.

Рассмотрены задачи распознавания аварий и таксономии применительно к энергосистемам. Намечены пути создания систем поддержки принятия решений при аварийных ситуациях в энергосистемах.

Библиогр.: 4 назв.

УДК 681.3:621.311

Розпізнавання аварійних ситуацій в енергосистемах / А.Л. Єрохін // Проблеми біоніки. 1998. Вип. 49. С. 67—71.

Розглянуто задачі розпізнавання аварій і таксономії в енергосистемах. Накреслено шляхи створення систем підтримки прийняття рішень під час аварійних ситуацій в енергосистемах.

Бібліогр.: 4 назви.

UDC 681.3:621.311

The recognition of emergencies in power systems / A.L. Yerokhin // Problemy Bioniki. 1998. N 49. P. 67—71.

The tasks of taxonomy and recognition of failure in power systems are considered. The known methods of decision are systematized. The ways of creating a system of support of decisions making in an emergency are considered.

Ref.: 4 titles.

УДК 612.821:007

Детерминантный анализ системы переработки информации человека / С.И. Маторин // Проблемы бионики. 1998. Вып. 49. С. 72—80.

Предложена партитивная (целочастная) классификация уровней сенсомоторной активности, разработанная с помощью средств детерминантного анализа. Дано определение текущей внутренней детерминанты человека. Рассмотрена проблема допустимости создания искусственного интеллекта, тождественного естественному по функциональным свойствам. Обоснована необходимость предварительного исследования внешней детерминанты естественного интеллекта средствами системологии.

Ил. 1. Библиогр.: 16 назв.

УДК 612.821:007

Детермінантний аналіз системи переробки інформації людини / С.І. Маторін // Проблеми біоніки. 1998. Вип. 49. С. 72—80.

Запропоновано партитивну (цілочастинну) класифікацію рівнів сенсомоторної активності, яку розроблено за допомогою засобів детермінантного аналізу. Подано визначення поточної внутрішньої детермінанти людини. Розглянуто проблему допустимості створення штучного інтелекту, тотожного природному за функціональними властивостями. Обгрунтовано необхідність попереднього дослідження зовнішньої детермінанти природного інтелекту засобами системології.

Іл. 1. Бібліогр.: 16 назв.

UDK 612.821:007

Determinant analysis of information processing system by man / S.I. Matorin // Problemy Bioniki. 1998. N 49. P. 72—80.

A whole-part classification of sensomotor activity levels with the help of the determinant analysis tools is developed. The definition of man's current internal determinants is suggested. The problem of admissibility of creation of an artificial intelligence identical in the functional properties to natural is considered. The necessity of a preliminary natural intelligence external determinants research with the help of systemology tools is justified.

1 fig. Ref.: 16 titles.

УДК 612.821:007

С **Сравнительный анализ систем переработки информации компьютера и человека** / Д.А. Гевчук, С.И. Маторин // Проблемы бионики. 1998. Вып. 49. С. 81—89.

Рассмотрены результаты сравнительного анализа функций человеческого интеллекта, моделируемых средствами искусственного интеллекта (ИИ). и компьютерных информационных процессов, посредством которых осуществляется это моделирование. Обоснована необходимость моделирования инстинкта самосохранения для обеспечения адекватности систем ИИ естественному интеллекту. Использованная методология партитивной классификации предложена в качестве универсального средства анализа сложных систем, обладающих активностью и отражением.

Табл. 1. Ил. 1. Библиогр.: 6 назв.

УДК 612.821:007

Порівняльний аналіз систем переробки інформації комп'ютера і людини / Д.О. Гевчук, С.І. Маторін // Проблеми біоніки. 1998. Вип. 49. С. 81—89.

Розглянуто результати порівняльного аналізу функцій інтелекту людини, що моделюються засобами штучного інтелекту (ШІ), і комп'ютерних інформаційних процесів, за допомогою яких здійснюється це моделювання. Обгрунтовано необхідність моделювання інстинкту самозбереження для забезпечення адекватності систем ШІ природному інтелекту. Використану методологію партитивної класифікації запропоновано як універсальний засіб аналізу складних систем, що мають активність та відбиття.

Табл. 1. Іл. 1. Бібліогр.: 6 назв.

UDC 612.821:007

Comparative analysis of information processing by man and computer / D A. Gevchuk, S.I. Matorin // Problemy Bioniki. 1998. N 49. P. 81—89.

The results are considered of the comparative analysis of human intelligence functions, simulated by tools of artificial intelligence, and computer information processes, by means of which this simulation is carried out. The necessity of the self-saving instinct simulation for a support of adequacy of artificial intelligence system to natural intelligence system is justified. The used methodology of whole-part classification is offered as a universal remedy for the analysis of complicated systems possessing activity and reflection.

1 tab. 1 fig Ref.: 6 titles.

УДК 519.7

О логических операторах / Н.А. Гвоздинская // Проблемы бионики. 1998. Вып. 49 С. 90—94.

Введены понятия логического оператора, произведения логических операторов. линейного логического оператора. Описаны единичный и обратимый логические операторы Для совершенного логического пространства доказана теорема о существовании единственного линейного логического оператора, переводящего его базис в некоторый базис другого логического пространства.

Библиогр.: 4 назв.

УДК 519.7

Про логічні оператори / Н.А. Гвоздінська // Проблеми біоніки. 1998. Вип. 49. С. 90—94.

Уведено поняття логічного оператора, добутку логічних операторів, лінійного логічного оператора. Описано одиничний та оборотний логічні оператори. Для досконалого логічного простору доведено теорему про існування єдиного лінійного логічного оператора, що переводить його базис у деякий базис іншого логічного простору.

Бібліогр.: 4 назви.

UDC 519.7

On the logical operators / N.A. Gvozdinska // Problemy Bioniki. 1998. N 49. P. 90—94.

The notion of a logical operator is introduced.The operation of the multiplication of the logical operators is considered. The unit operator and the invertable operator are described. The notion of the linear operator is introduced. The theorem of the existence of the one and only one linear operator for the perfect logical spaces mapping its basis on some basis of another logical space are proved.

Ref.: 4 titles.

УДК 519.85

Решение задачи размещения неориентированного объекта в области с переменными метрическими характеристиками / М.В. Новожилова // Проблемы бионики. 1998. Вып. 49. С. 95—98.

Рассмотрена оптимизационная задача размещения невыпуклого неориентированного многоугольника в невыпуклой области с переменными метрическими характеристиками. Описан метод поиска глобального минимума.

Библиогр.: 2 назв.

УДК 519.85

Розв'язання задачі розміщення неорієнтованого об'єкта в області зі змінними метричними характеристиками / М.В. Новожилова // Проблеми біоніки. 1998. Вип. 49. С. 95—98.

Розглянуто оптимізаційну задачу розміщення неопуклого неорієнтованого багатокутника в неопуклій області зі змінними метричними характеристиками. Описано метод пошуку глобального мінімуму.

Бібліогр.: 2 назви.

UDC 519.85

Solving the problem of placement of a non-oriented object into a region with variable metric characteristics / M.V. Novozhilova // Problemy Bioniki. 1998. N 49. P. 95—98.

The optimization problem of placement of a non-oriented non-convex object into non-convex region with variable metric characteristics has been considered. The method of searching the global minimum of the problem objective function has been described.

Ref.: 2 titles.

УДК 007.681.518.2

Задание топологии на многоаспектных классификационных структурах при моделировании концептуальных знаний / М.Ф. Вороной // Проблемы бионики. 1998. Вып. 49. С. 99—108.

Рассмотрена формализация представления концептуальных слабоформализованных знаний произвольной природы в виде многоаспектных классификационных структур на основе аппарата топологии. Предложенный способ создания понятийных моделей может быть использован для реализации средств поддержки и управления базами знаний.

Ил. 1. Библиогр.: 8 назв.

УДК 007.681.518.2

Завдання топології на багатоаспектних класифікаційних структурах під час моделювання концептуальних знань / М.П. Вороний // Проблеми біоніки. 1998. Вип. 49. С. 99—108.

Розглянуто формалізацію подання концептуальних слабкоформалізованих знань довільної природи у вигляді багатоаспектних класифікаційних структур на основі апарату топології. Запропонований спосіб створення понятійних моделей знань може бути використаний для реалізації засобів підтримки та керування базами знань.

Іл. 1. Бібліогр.: 8 назв.

UDC 007.681.518.2

Topology assignment to multiaspect classification structures in conceptual knowledge modelling / M.F. Voronoy // Problemy Bioniki. 1998. N 49. P. 99—108.

The questions of topology assignment to multiaspect classification structures in conceptual knowledge modelling are reviewed. The suggested method can be used as a formalism for knowledge based system construction and control.

1 fig. Ref.: 8 titles.

УДК 007.681.518.2

Формализация противоречивых знаний на концептуальных классификационных структурах в терминах открытых множеств аппарата топологии / М.Ф. Вороной // Проблемы бионики. 1998. Вып. 49. С. 109—112.

Рассмотрен перспективный путь формализации представления противоречивой информации для многоаспектных концептуальных классификационных структур, при котором используется понятие открытого множества из аппарата топологии. Введение формальной трактовки противоречивых понятийных знаний необходимо для создания автоматизированных систем приобретения и использования знаний.

Библиогр.: 3 назв.

УДК 007.681.518.2

Формалізація суперечливих знань на концептуальних класифікаційних структурах у термінах відкритих множин апарату топології / М.П. Вороний // Проблеми біоніки. 1998. Вип. 49. С. 109—112.

Розглянуто перспективний шлях формалізації подання суперечливої інформації для багатоаспектних концептуальних класифікаційних структур, коли використовується поняття відкритої множини з апарату топології. Уведення формального трактування суперечливих знань необхідне для створення автоматизованих систем надбання та використання знань.

Бібліогр.: 3 назви.

UDC 007.681.518.2

Conception of contradictory knowledge assigment for conceptual classification structures in terms of topology open sets / M.F. Voronoy // Problemy Bioniki. 1998. N 49. P. 109—112.

A promising way of a formal presentation of contradictory knowledge for conceptual classification structures is investigated in terms of topology open sets by topology. This investigation is useful for automated systems that acquire and utilize knowledge.

Ref.: 3 titles.

УДК 519.23/25

Области локализации характеристических корней квазистохастических матриц / С.Н. Герасин // Проблемы бионики. 1998. Вып. 49. С. 113—117.

Рассмотрены вопросы локализации характеристических корней квазистохастических матриц на комплексной плоскости. Определены границы области плоскости, где находятся характеристические корни всего многообразия квазистохастических матриц. Установлена связь между корнями с отрицательной действительной частью и стационарным режимом соответствующей системы уравнений Колмогорова, с помощью которой можно описывать различные модели биологических, экологических и экономических процессов.

Библиогр.: 3 назв.

УДК 519.23/25

Області локалізації характеристичних коренів квазістохастичних матриць / С.М. Герасін // Проблеми біоники. 1998. Вип. 49. С. 113—117.

Розглянуто питання локалізації характеристичних коренів квазістохастичних матриць на комплексній площині. Визначено межі області площини, де знаходяться характеристичні корені всього многовиду квазістохастичних матриць. Виявлено зв'язок між коренями з від'ємною дійсною частиною та стаціонарним режимом відповідної системи рівнянь Колмогорова, за допомогою якої можна описувати різні моделі біологічних, екологічних та економічних процесів.

Бібліогр.: 3 назви.

UDC 519.23/25

Charachteristic root localization regions for quasi-stochastic matrices / S.N. Gerasin // Problemy Bioniki. 1998. N 49. P. 113—117.

The complex root localization problem for a quasi-stochastic matrix is considered. Limits of root location region on the complex plane were found for the whole set of quasi-stochastic matrices. The accordance was found between roots with negative real part and stationary modes of respective Kolmogorov's system that describe various process models in biology, ecology and economics.

Ref.: 3 titles.

УДК 519.85

Особенности решения систем линейных неравенств при использовании модифицированного метода размещения многоугольников в полосе / К.Е. Коротин // Проблемы бионики. 1998. Вып. 49. С. 118—123.

Рассмотрены особенности поиска глобального минимума в задаче размещения невыпуклых многоугольников в полубесконечной полосе. Описано правило отсечения, позволяющее существенно сократить время решения задачи.

Ил. 2. Библиогр.: 3 назв.

УДК 519.85

Особливості розв'язання систем лінійних нерівностей під час використання модифікованого методу розміщення багатокутників у смузі / К.Ю. Коротін // Проблеми біоніки. 1998. Вип. 49. С. 118—123.

Розглянуто особливості пошуку глобального мінімуму в задачі розміщення неопуклих багатокутників у напівнескінченній смузі. Описано правило відтинання, яке дозволяє суттєво скоротити час розв'язання задачі.

Іл. 2. Бібліогр.: 3 назви.

UDC 519.85

The solution features of the linear inequalities system in the problem of polygons allocation in the strip / K.Y. Korotin // Problemy Bioniki. 1998. N 49. P. 118—123.

The features of the algorithm of the global extremum search in the improved method of nonconvex polygons allocation in the semi-infinite strip are described. The cutting rule allowing to reduce significantly the algorithm run time is treated.

2 fig. Ref.: 3 titles.

УДК 801.1+ 801.7

Принципы построения двуязычного универсального электронного словаря / А.В. Рассадникова // Проблемы бионики. 1998. Вып. 49. С. 124—129.

Проанализированы практические цели, преимущества, источники формирования, лингвистические и программные принципы построения двуязычного универсального электронного словаря нового поколения. Сформулированы требования к организации словаря с помощью ЭВМ.

Табл. 1. Библиогр.: 3 назв.

УДК 801.1+801.7

Принципи побудови двомовного універсального електроного словника / Г.В. Рассаднікова // Проблеми біоніки. 1998. Вип. 49. С. 124—129.

Проаналізовано практичні цілі, переваги, джерела формування, лінгвістичні та програмні принципи створення двомовного універсального електронного словника нового покоління. Сформульовано вимоги щодо організації словника за допомогою ЕОМ.

Табл. 1. Бібліогр.: 3 назви.

UDC 801.1+801.7

Principles of making a universal electronic dictionary / A.V. Rassadnikova // Problemy Bioniki. 1998. N 49. P. 124—129.

Consideration is given to practical purposes, advantages, sources of forming, linguistic and programming construction principles of a bilingual universal electronic dictionary of a new generation. Demands to organization of the dictionary with the help of computer are formulated.

Tab. 1. Ref.: 3 titles.

УДК 518.5

Моделирование геометрических объектов в системах анализа физических полей / А.Н. Шевченко, О.В. Тоница // Проблемы бионики. 1998. Вып. 49. С. 130—134.

Рассмотрено построение аналитических описаний геометрических объектов с использованием теории R-функций и нечеткой логики. Предложены методы и алгоритмы моделирования геометрических объектов в системах анализа физических полей, учитывающие технические и технологические допуски. Описаны два класса алгоритмов вычисления функции принадлежности текущей геометрической области эталонной: классы характеристик по дискретному множеству характерных точек и интегральных характеристик.

Библиогр.: 6 назв.

УДК 518.5

Моделювання геометричних об'єктів у системах аналізу фізичних полів / О.М. Шевченко, О.В. Тоніца // Проблеми біоніки. 1998. Вип. 49. С. 130—134.

Розглянуто побудову аналітичних описів геометричних об'єктів із використанням теорії R-функцій та нечіткої логіки. Запропоновано методи та алгоритми моделювання геометричних об'єктів у системах аналізу фізичних полів, що враховують технічні та технологічні допуски. Описано два класи алгоритмів обчислення функції належності поточної геометричної області еталонній: класи характеристик по дискретній множині характерних точок та інтегральних характеристик.

Бібліогр.: 6 назв.

UDC 518.5

Simulation of geometric objects in systems for analysis of physical fields / A.N. Shevchenko, O.V. Tonitsa // Problemy Bioniki. 1998. N 49. P. 130—134.

The paper deals with a problems of creating an analytical description for geometric objects with the help of the R-functions theory and fuzzy logics. Some methods and algorithms for simulation of geometrical objects in systems for analysis of physical fields are proposed taking into account some technical and technological assumptions. The algorithms for computing fuzzy membership functions for geometric domains are elaborated.

Ref.: 6 titles.

УДК 518.5

Развитие интеллектуальных систем анализа физических полей серии «Поле» // А.Н. Шевченко, О.В. Тоница // Проблемы бионики. 1998. Вып. 49. С. 135—140.

Рассмотрены вопросы развития систем анализа физико-механических полей серии «Поле» для учета стохастического характера технологических допусков на физическую и геометрическую информацию. Изложена методика, позволяющая моделировать поля и получать экспертные заключения о приемлемости найденных решений. Методика основана на построении описаний нечетких точечных множеств методами теории R-функций и нечеткой логики. Она включает в себя формирование допусков на поле в задаче анализа или допусков на геометрию в задаче синтеза и решение реальной задачи моделирования.

Библиогр.: 3 назв.

УДК 518.5

Розвиток інтелектуальних систем аналізу фізичних полів серії «Поле» // О.М. Шевченко, О.В. Тоніца // Проблеми біоніки. 1998. Вип. 49. С. 135—140.

Розглянуто питання розвитку систем аналізу фізико-механічних полів серії «Поле» для врахування стохастичного характеру технологічних допусків на фізичну та геометричну інформацію. Викладено методику, яка дозволяє моделювати поля і одержувати експертні висновки щодо прийнятності винайдених рішень. Методика грунтується на побудуванні описів нечітких точечних множин методами теорії R-функцій та нечіткої логіки. Вона містить формування допусків на поле в задачі аналізу або допусків на геометрію в задачі синтезу та розв'язання реальної задачі моделювання.

Бібліогр.: 3 назви.

UDC 518.5

Development of intellectual systems of POLYE family for analysis of physical fields // A.N. Shevchenko, O.V. Tonitsa // Problemy Bioniki. 1998. N 49. P. 135—140.

The paper deals with problems of development of systems of POLYE family for analysis of physical-mechanical fields with taking into account stochastic characteristics of technological assumptions on physical and geometrical information. The methodology is suggested which allows us to simulate fields and obtain expert conclusions about acceptability of the found solutions. The technique elaborated suggests description of fuzzy sets by means of the R-functions theory and fuzzy logics. The technique suggested includes forming assumptions on a field in an analysis problem or assumptions on a geometry in a synthesis problem, and solving real problem and expert conclusion about acceptability of the found solutions.

Ref.: 3 titles.

УДК 519.23–25

Эргодические свойства цепей Маркова с переменным числом состояний и модель устойчивого развития психики / Н.И. Числин // Проблемы бионики. 1998. Вып. 49. С. 141—146.

Рассмотрена модель функционирования психики на этапе интенсивного развития. Для описания процесса смены психических состояний применен аппарат цепей Маркова с переменным числом состояний. Путем распространения на такие цепи понятий слабой и сильной эргодичностей получены условия относительной асимптотической устойчивости психических реакций. Предложенная модель может быть использована для определения правил поведения систем искусственного интеллекта при изменяющемся количестве альтернатив.

Библиогр.: 2 назв.

УДК 519.23–25

Ергодичні властивості ланцюгів Маркова зі змінною кількістю станів і модель стійкого розвитку психіки / М.І. Числін // Проблеми біоніки. 1998. Вип. 49. С. 141—146.

Розглянуто модель функціонування психіки на етапі інтенсивного розвитку. Для опису процесу зміни психічних станів застосовано апарат ланцюгів Маркова зі змінною кількістю станів. Шляхом розповсюдження на такі ланцюги понять слабкої та сильної ергодичностей отримано умови відносної асимптотичної стійкості психічних реакцій. Запропонована модель може бути використана для визначення правил поводження систем штучного інтелекту при змінюваній кількості альтернатив.

Бібліогр.: 2 назви.

UDC 519.23–25

Ergodic properties of Markov's chains with the variable number of states and the model of steady development of psychics / N.I. Chyslin // Problemy Bioniki. 1998. N 49. P. 141—146.

The model of a psychic process at the stage of active growth has been considered. In order to describe the consequent psychic states changing the means of Markov's chains with the variable number of states have been used. By way of extending the definition of weak and strong ergodicity onto such chains the conditions of relative asymptotic stability of psychic reactions have been yielded. The proposed model can be used for the determination of the rules of behaviour for the systems of artificial intelligence under a variable number of possible alternatives.

Ref.: 2 titles.

УДК 519.7

Моделирование процесса морфологической обработки словоформ методом решения кванторных предикатных уравнений / Е.В. Журавок // Проблемы бионики. 1998. Вып. 49. С. 147—150.

Предложен алгоритм, который обеспечивает решение системы уравнений, задающей морфологическое отношение. В основу алгоритма положена теория линейных логических операторов. Показано, что из решений соответствующих операторных уравнений в линейном логическом пространстве получаются решения необходимых кванторных уравнений.

Библиогр.: 2 назв.

УДК 519.7

Моделювання процесу морфологічної обробки словоформ методом розв'язання кванторних предикатних рівнянь / О.В. Журавок // Проблеми біоніки. 1998. Вип. 49. С. 147—150.

Запропоновано алгоритм, який забезпечує розв'язання системи рівнень, що задає морфологічне відношення. В основу алгоритму покладено теорію лінійних логічних операторів. Доведено, що із розв'язків відповідних операторних рівнянь у лінійному логічному просторі виходять розв'язки необхідних кванторних рівнянь.

Бібліогр.: 2 назви.

UDC 519.7

Modelling the morphological processing of word-forms by a method of solving quantifier predicate equations / E.V. Zhuravok // Problemy Bioniki. 1998. N 49. P. 147—150.

An algorithm ensuring the solving of a system of equations, specifying some morphological relations is developed. The algorithm is based on the theory of linear logic operators. From the solutions of appropriate operator equations in linear logic space the solutions of necessary quantifier equations are obtained.

Ref.: 2 titles.

УДК 681.324

Формализация нечетких знаний на основе моделей нечетких алгоритмических квантов знаний (НАКЗ-моделей) для компьютерного принятия решений в условиях неопределенности / М.Ю. Гацкий // Проблемы бионики. 1998. Вып. 49. С. 151—161.

Рассмотрен механизм образования многоуровневых нечетких $\tilde{k}$-знаний из исходных нечетких квантов, являющихся стандартной формой представления нечетких знаний при использовании метода нечетких алгоритмических квантов знаний (НАКЗ-метод). Определена функция достоверности фактов и закономерностей, описанных в виде нечетких квантов $\tilde{k}$-знаний.

Табл. 1. Библиогр.: 4 назв.

УДК 681.324

Формалізація нечітких знань на основі моделей нечітких алгоритмічних квантів знань (НАКЗ-моделей) для комп'ютерного прийняття рішень в умовах невизначеності / М.Ю. Гацький // Проблеми біоніки. 1998. Вип. 49. С. 151—161.

Розглянуто механізм утворення багаторівневих нечітких $\tilde{k}$-знань з вихідних нечітких квантів, які є стандартною формою подання нечітких знань під час використання методу нечітких алгоритмічних квантів знань (НАКЗ-метод). Визначено функцію вірогідності фактів і закономірностей, що описані у вигляді нечітких квантів $\tilde{k}$-знань.

Табл. 1. Бібліогр.: 4 назви.

UDC 681.324

Formalization of fuzzy knowledge on the basis of fuzzy algorithm quanta knowledge (FAQK-models) for computer making decisions in the conditions of uncertainty / M.Yu. Gatsky // Problemy Bioniki. 1998. N 49. P. 151—161.

The mechanism of the formation of multi-level fuzzy $\tilde{k}$ -knowledge from initial fuzzy quanta which is a standard form of presenting fuzzy knowledge in the metod of fuzzy algorithm quanta knowledge (FAQK-method) was considered. The authenticity function of examined facts and mechanisms described in the form of fuzzy quanta $\tilde{k}$ -knowledge was defined.

1 tab. Ref.: 4 titles.

УДК 681.324

Операторы индуктивного и дедуктивного вывода $\tilde{k}$ -знаний в классе моделей нечетких алгоритмических квантов знаний (НАКЗ-моделей) / М.Ю. Гацкий // Проблемы бионики. 1998. Вып. 49. С. 162—167.

Определен алгоритм индуктивного вывода нечетких знаний и построения базы нечетких знаний исходя из имеющихся фактов, описан алгоритм дедуктивного вывода нечетких знаний в классе моделей нечетких алгоритмических квантов знаний (НАКЗ-моделей).

Библиогр.: 3 назв.

УДК 681.324

Оператори індуктивного й дедуктивного виведення $\tilde{k}$ -знань у класі моделей нечітких алгоритмічних квантів знань (НАКЗ-моделей) / М.Ю. Гацький //Проблеми біоніки. 1998. Вип. 49. С. 162—167.

Визначено алгоритм індуктивного виведення нечітких знань і побудови бази нечітких знань виходячи із наявних фактів, описано алгоритм дедуктивного виведення нечітких знань у класі моделей нечітких алгоритмічних квантів знань (НАКЗ-моделей).

Бібліогр.: 3 назви.

UDC 681.324

Operators of inductive and deductive output of $\tilde{k}$ -knowledge in a class of fuzzy algorithm quanta knowledge (FAQK-models) / M.Yu. Gatsky // Problemy Bioniki. 1998. N 49. P. 162—167.

An algorithm of inductive output of fuzzy knowledge and creating a base of fuzzy knowledge out of available facts and an algorithm of deductive output of fuzzy knowledge in a class of models of fuzzy algorithm quanta knowledge (FAQK-models) are described.

Ref.: 3 titles.

УДК 658.012:007

Идентификация нелинейных зависимостей нейронными сетями / А.П. Ротштейн, Ю.И. Митюшкин // Проблемы бионики. 1998. Вып. 49. С. 168—174.

Предложен метод идентификации нелинейных зависимостей путем обучения многослойной нейронной сети. Обучение осуществляется на основе заранее определенной обучающей выборки по методу обратного распространения ошибки. На первом этапе каждой итерации процедуры обучения вычисляются значения выходов нейронной сети, на втором этапе находиться погрешность и на ее основании производится модификация межнейронных связей по правилу градиента. Приведены результаты компьютерного эксперимента, иллюстрирующего обучение нейронной сети как идентификатора одномерной нелинейной зависимости.

Ил. 4. Библиогр.: 7 назв.

УДК 658.012:007

Ідентифікація нелінійних залежностей нейронними мережами / О.П. Ротштейн. Ю.І. Мітюшкін // Проблеми біоніки. 1998. Вип. 49. С. 168—174.

Запропоновано метод ідентифікаці нелінійних залежностей шляхом навчання багатошарової нейронної мережі. Навчання здійснюється на основі наперед визначеної навчальної вибірки за методом зворотнього розповсюдження помилки. На першому етапі кожної ітерації процедури навчання обчислюються значення виходів нейронної мережі, на другому етапі обчислюється похибка і на її основі проводиться модифікація міжнейронних зв`язків за правилом градієнта. Наведено результати комп`ютерного експерименту, що ілюструє навчання нейронної мережі як ідентифікатора одномірної нелінійної залежності.

Іл. 4 Бібліогр.: 7 назв.

UDC 658.012:007

Identification of nonlinear functions by neural networks / A.P. Rotshteyn, Y.I. Mitushkin // Problemy Bioniki. 1998. N 49. P. 168—174.

Proposes method of nonlinear functions identification by teaching of multylayer neural network. The teaching is realised by training data by the back-propagation algorithm. Outputs of neural network are computed in the first phase of every teaching iteration. Error of neural network is computed in the second phase, and then the weights of interneuron connections are changed by the gradient rule. Computer simulation illustrates the teaching of neural network by the example of nonlinear function with one input variable.

4 fig. Ref.: 7 titles.

УДК 519.712

Использование таблиц решений в описании интеллектуальных задач / П. Л. Самофалов // Проблемы бионики. 1998. Вып. 49. С.175—180.

Рассмотрена методика представления баз знаний таблицами решений. Особенностью подхода является замена логического вывода выводом в аналоге семантической сети из продукций, представленных таблицами решений. Это позволяет повысить эффективность построения и функционирования программных систем с элементами искусственного интеллекта.

Табл. 3. Ил. 1. Библиогр.: 4 назв.

УДК 519.712

Використання таблиць рішень в описі інтелектуальних задач / П. Л. Самофалов // Проблеми біоніки. 1998. Вип. 49. С. 175—180.

Розглянуто методику зображення баз знань за допомогою таблиць рішень. Особливістю підходу є заміна логічного виведення виведенням в аналозі семантичної сітки з продукцій, що являють собою таблиці рішень. Це дозволяє підвищити ефективність побудови та функціонування програмних систем з елементами штучного інтелекту.

Табл. 3. Іл. 1. Бібліогр.: 4 назви.

UDC 519.712

Intellectual tasks description with the use of decision tables / P.L. Samofalov // Problemy Bioniki. 1998. N 49. P. 175—180.

The method for knowledge base presentation with the use of decision tables is considered. The specific feature of the suggested approach is the replacement of logical deduction by deduction using a semantic network which consists of productions represented by decision tables. At that, decision tables can be better converted into programs. This approach allows to increase the efficiency of constructing and functioning software systems with artificial intelligence elements.

3 tab. 1 fig. Ref.: 4 items.

УДК 371.315.7: 681.513.6

Об одной методологии разработки программных моделей / В.В. Тищенко // Проблемы бионики. 1998. Вып. 49. С. 181—185.

Рассмотрена методология конструирования программных моделей, имеющая широкий спектр применения в проектировании систем. Описаны стадии доведения разработки до постановки задачи на проектирование, включая стадию дескриптизации и исследования.

Библиогр.: 4 назв.

УДК 371.315.7: 681.513.6

Про одну методологію розробки програмних моделей / В.В. Тищенко // Проблемы біоніки. 1998. Вип. 49. С. 181—185.

Розглянуто методологію конструювання програмних моделей, яка має широкий спектр використання в проектуванні систем. Описано стадії доведення розробки до постановки задачі на проектування, включаючи стадію дескриптизації та дослідження.

Бібліогр.: 4 назви.

UDC 371.315.7: 681.513.6

On one methodology of program models development / V.V. Tischenko // Problemy Bioniki. 1998. N 49. P. 181—185.

The methodology of program models construction having a wide range of usage in designing systems is considered. The methodology brings the development up to the defining the task of designing, including the stage of description and research.

Ref.:4 titles.

УДК 681.142.1.01

Методы оценки релевантности информационных структур в базах знаний / В.Е. Ходаков, В.Г. Шерстюк, К.Г. Степанский, А.А. Дидык, А.Н. Мартынов // Проблемы бионики. 1998. Вып. 49. С. 186—195.

Рассмотрены методы определения релевантности информационных структур в базах знаний в целях прогнозирования изменений, контроля за полнотой знаний, оценки альтернативных путей решения задач, переоценки принятых решений, уменьшения вреда от неудачных решений.

Ил. 2. Библиогр.: 10 назв.

УДК 681.142.1.01

Методи оцінювання релевантності інформаційних структур у базах знань / В.Є. Ходаков, В.Г. Шерстюк, К.Г. Степанський, О.О. Дідик, А.М. Мартинов // Проблеми біоніки. 1998. Вип. 49. С. 186—195.

Розглянуто методи визначення релевантності інформаційних структур у базах знань з метою прогнозування наслідків змін, контролю за повнотою знань, оцінювання альтернативних шляхів вирішення задач, переоцінки прийнятих рішень, зменшення шкоди від невдалих рішень.

Іл. 2. Бібліогр.: 10 назв.

UDC 681.142.1.01

Relevance evaluation methods for information structures in knowledge bases / V.E. Khodakov, V.G. Sherstyuk, K.G. Stepanskiy, A.A. Didyk, A.N. Martynov // Problemy Bioniki. 1998. N 49. P. 186—195.

Relevance evaluation methods for information structures in knowledge bases are considered with the purpose of prediction the changes consequences, control a knowledge completeness, evaluation of alternate paths in problem solving, revaluation of the adopted solutions, reduction of consequences, that appeared after unsuccessful solutions.

2 fig. Ref.: 10 titles.

УДК 519.713

Модели принятия решений на основе иерархических вероятностных квантов знаний / О.Ю. Голобродский, И.Б. Сироджа // Проблемы бионики. 1998. Вып. 49. С. 196—202.

Описаны знаниеориентированные модели и алгоритмические средства для автоматизации процесса принятия решений на основе иерархических вероятностных квантов знаний. Сформулированы индуктивный принцип построения Базы Знаний и дедуктивный принцип принятия решений.

Библиогр.: 2 назв.

УДК 519.713

Моделі прийняття рішень на основі ієрархічних імовірнісних квантів знань / О.Ю. Голобродський, І.Б. Сироджа // Проблеми біоніки. 1998. Вип. 49. С. 196—202.

Описано знанняорієнтовані моделі та алгоритмічні засоби для автоматизації процесу прийняття рішень на основі ієрархічних імовірнісних квантів знань. Сформульовано індуктивний принцип побудови Бази Знань та дедуктивний принцип прийняття рішень.

Бібліогр.: 2 назви.

UDC 519.713

Hierarhical probabilistic knowledge quanta based models for decision making / O.Yu. Golobrodsky, I.B. Sirodza // Problemy Bioniki. 1998. N 49. P. 196—202.

The goal of the paper is to describe the knowledge-oriented models and algorithmic means for decision making automatization with the help of hierarhical probabilistic knowledge quanta. Inductive principle for knowledge-base building and deductive decision making method are presented.

Ref.: 2 titles.

УДК 371.385:681.3

Математические модели построения виртуальных пространств / Н.В. Белоус, А.П. Выродов, И.Ю. Шубин // Проблемы бионики. 1998. Вып. 49. С. 203—211.

Приведена математическая модель метода трассировки луча, оптимизирующая процесс превращения двухмерного представления пространства в трехмерное. Рассмотрены возникающие проекционные искажения и указан способ их устранения. Описаны основные моменты программной реализации данного метода.

Ил. 6. Библиогр.: 4 назв.

УДК 371.385:681.3

Математичні моделі побудови віртуальних просторів / Н.В. Білоус, О.П. Виродов, І.Ю. Шубін // Проблеми біоніки. 1998. Вип. 49. С. 203—211.

Наведено математичну модель методу трасування променя, яка оптимізує процес перетворення двовимірного подання простору в тривимірне. Розглянуто проекційні викривлення, що виникають, і вказано спосіб їх усунення. Описано основні моменти програмної реалізації даного методу.

Іл. 6. Бібліогр. 4 назви.

UDC 371.385:681.3

Mathematical models for constructing of virtual spaces / N.V. Belous, A.P. Vyrodow, I.Y. Schubin // Problemy Bioniki. 1998. N 49. P. 203—211.

This article contains mathematics model of the ray casting method, an explanation of projection distortions and a way of their removal, the principal moments of program realization of the given method. The ray casting method describes the process where a 2D representation of a virtual space is rendered into 3D.

6 fig. Ref.: 4 titles.