УДК 681.3:517.11

И.Н. ПАРАСЮК, С.В. ЕРШОВ

ТЕОРЕТИКО-КАТЕГОРНАЯ ФОРМАЛИЗАЦИЯ ПРИ ПРОЕКТИРОВАНИИ НЕЧЕТКИХ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ

Ключевые слова: *теория категорий, нечеткие графы, трансформационный подход, трансформации графов, архитектура, управляемая моделями*.

ВВЕДЕНИЕ

Модели на основе нечеткой логики распространены в интеллектуальных системах принятия решений, в экономике и финансовой сфере, в задачах прогнозирования и стратегического анализа. Системы нечеткого вывода Мамдани, Цукамото, Сугено и Ларсена способны решать классы задач в условиях неопределенности и неполноты информации [1].

В настоящее время два основных подхода к моделированию знаний основаны на объектной парадигме: модель MDA (Model Driven Architecture) [2] консорциума OMG и модель ODP (Model of Open Distributed Processing), зафиксированная в стандарте [3]. Хотя эти модели использовались в проектировании программных систем, еще недостаточно исследованы методы их формализации и представления для разработки нечетких интеллектуальных систем.

Подход MDA основан на наборе элементарных операций над моделями, называемых трансформациями моделей. Часто такие правила описаны набором правил или продукций [2]. В частности, если в основу положены графовые модели, то правила заданы в рамках СТГ — систем трансформаций графов (называемых также графовыми грамматиками [4, 5]). Однако, несмотря на применение СТГ для представления моделей, не разработаны способы их трансформации в нечетком пространстве, образованном при создании интеллектуальных систем.

Целью работы является разработка теоретико-категорной формализации нечетких диаграмм UML, нечетких графов и систем трансформации нечетких графов (СТНГ). Исследуются основные ограничения относительно нечеткости, которым должны удовлетворять диаграммы UML, представляющие основу спецификации таких моделей. Анализируются свойства спецификации вывода на основе распределенной системы трансформаций нечетких графов и предлагается новый подход к разработке инструментальных средств композиционного конструирования нечеткой системы на основе объектов, морфизмов и категорий нечетких графов.

СРЕДСТВА ПРЕДСТАВЛЕНИЯ НЕЧЕТКОСТИ ОБЪЕКТНО-ОРИЕНТИРОВАННЫХ МОДЕЛЕЙ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ

Структурная модель нечеткой интеллектуальной системы состоит из двух частей: программной среды, позволяющей создавать нечеткие системы для выбранной предметной области, и собственно интеллектуальной системы как конечного продукта. Сюда включены база нечетких знаний, модуль идентификации нечетких правил (обучения), модули фуззификации и дефуззификации, средства нечеткого логического вывода.

Основные средства представления нечеткости интеллектуальных систем в модели MDA включают четыре концептуальных уровня: 4L-M0, 4L-M1, 4L-M2 и 4L-M3. На уровне 4L-M0 представлены данные о нечетких объектах системы, на уровне 4L-M1 расположены модели, которым удовлетворяют описания объектов на предыдущем уровне. Уровень 4L-M2 занимают метамодели, определяющие способы спецификации нечетких моделей. Наконец, на уровне 4L-M3 находится единственная метамодель, не зависящая от контекста и определяющая способы построения метамоделей уровня 4L-M2. Возможность употребления ODP ограничена тремя уровнями и содержит ряд представлений модели 1-го уровня, что вызывает необходимость установления формального соответствия меры нечеткости для разных представлений одной системы.

Модели 1-го уровня 4L-M1 могут быть представлены ориентированным мультиграфом, вершины которого соответствуют объектам, а ребра — элементам определенности, характеризующим объекты. Любая вершина характеризуется множеством нечетких атрибутов, а каждое ребро помечено значимым и информативным весами (μ_V, μ_I), которые эквивалентны значениям функций принадлежности нечетких свойств и отношений. Поэтому сформированную структуру можно интерпретировать как нечеткий взвешенный ориентированный мультиграф.

Подход MDA не зависит от инструментов и языка моделирования. Однако предусматривается, что для описания графических моделей нечетких систем используется язык UML [6]. Он предоставляет набор моделей, охватывающих разные аспекты системы программного обеспечения на основе объектно-ориентированной парадигмы. С точки зрения моделирования архитектуры систем, основную модель составляют классы и отношения между ними: агрегат, композиция, обобщение, ассоциация и зависимость [6].

Рассмотрим нечеткий суперкласс A и его нечеткие подклассы $B_1, B_2, \ldots, B_n$ со степенью принадлежности $\mu_A, \mu_{B_1}, \mu_{B_2}, \ldots, \mu_{B_n}$ экземпляров, которые могут иметь степень принадлежности $\deg(A)$, $\deg(B_1)$, $\deg(B_2), \ldots, \deg(B_n)$ классов соответственно. Тогда верно отношение

$$(\forall\, e)\ (\max(\mu_{B_1}(e), \mu_{B_2}(e), \ldots, \mu_{B_n}(e)) \le$$
$$\le \mu_A(e)) \wedge (\max(\deg(B_1), \deg(B_2), \ldots, \deg(B_n)) \le \deg(A)).$$

Каждый экземпляр агрегата можно спроецировать на множество составляющих его экземпляров. Допустим, A — нечеткий агрегат нечетких наборов классов $B_1, B_2, \ldots, B_n$ со степенью принадлежности экземпляров $\mu_A, \mu_{B_1}, \mu_{B_2}, \ldots, \mu_{B_n}$ соответственно, β — заданный порог. Для $e \in A$ проекцию e на B_i обозначим $e\!\downarrow\! B_i$. Тогда $(e\!\downarrow\! B_1) \in B_1, (e\!\downarrow\! B_2) \in B_2, \ldots, (e\!\downarrow\! B_n) \in B_n$. Здесь e является объектом-экземпляром A. Тогда

$$(\forall\, e)(e \in A) \wedge (\beta \le \mu_A(e) \le \min(\mu_{B_1}(e\!\downarrow\! B_1), \mu_{B_2}(e\!\downarrow\! B_2), \ldots, \mu_{B_n}(e\!\downarrow\! B_n))).$$

Степень принадлежности агрегата A над набором классов $B_1, B_2, \ldots, B_n$ должна быть $\min_{\mu_{B_i}(e\downarrow B_i) \ge \beta} (\mu_{B_i}(e\!\downarrow\! B_i))$, $1 \le i \le n$.

Для отношений ассоциации UML можно идентифицировать два уровня нечеткости. Первый уровень означает, что отношение ассоциации нечетко существует для двух связанных классов, а именно, оно задается со степенью возможности. В общей ситуации неизвестно, состоят ли два экземпляра класса, которые соответственно принадлежат ассоциируемым классам, в заданном отношении ассоциации, хотя оно должно выполняться для двух классов. Это — второй уровень нечеткости в отношении ассоциации, и он обусловлен тем, что экземпляр принадлежит заданному классу только с определенной степенью принадлежности. Два уровня нечеткости можно определить в отношении ассоциации одновременно.

Допустим, A и B — два класса с первым уровнем нечеткости, обозначенных $\deg(A)$ и $\deg(B)$ соответственно. Пусть $\mathrm{ass}(A, B)$ — отношение ассоциации первого уровня нечеткости между A и B, которое обозначено $\deg(\mathrm{ass})$. Пусть также экземпляр e класса A — со степенью принадлежности $\mu_A(e)$, экземпляр f класса B — со степенью принадлежности $\mu_B(f)$. Тогда имеем

$$\mu(\mathrm{ass}(e, f)) = \min(\mu_A(e),\ \mu_B(f),\ \deg(A),\ \deg(B),\ \deg(\mathrm{ass})).$$

ТЕОРЕТИКО-КАТЕГОРНАЯ ФОРМАЛИЗАЦИЯ СИСТЕМ ТРАНСФОРМАЦИИ НЕЧЕТКИХ ГРАФОВ

Рассмотрим множество $M=(M_V, M_E)$, состоящее из множества M_V функций принадлежности вершин и множества M_E функций принадлежности ребер. Нечеткий граф $FG=(V,E,s,t,\mu_V,\mu_E)$ состоит из множества вершин V, ребер E, исходной s и целевой t функций ребер $s,t\colon E\to V$ соответственно и функций принадлежности, задающих для каждой вершины и ребра нечеткое множество:

$$\mu_V\colon V\to M_V, \quad \mu_E\colon E\to M_E.$$

Морфизм нечетких графов $f\colon F_1\to F_2$ — это морфизм между составляющими их основу четкими графами $f\colon F_1^{\text{ет}}\to F_2^{\text{ет}}$, который совместим с морфизмами, отображающими функции принадлежности вершин и ребер, т.е. $\mu_{2,V}\circ f_V = f_{MV}\circ\mu_{1,V}$, $\mu_{2,E}\circ f_E = f_{ME}\circ\mu_{1,E}$ и следующая диаграмма коммутирует:

$$\begin{tikzcd}
M_E \arrow[d, "f_{ME}"'] & E_1 \arrow[l, "\mu_{1,E}"'] \arrow[r, shift left, "s_1"] \arrow[r, shift right, "t_1"'] \arrow[d, "f_E"] & V_1 \arrow[r, "\mu_{1,V}"] \arrow[d, "f_V"] & M_V \arrow[d, "f_{MV}"] \\
M_E & E_2 \arrow[l, "\mu_{2,E}"'] \arrow[r, shift left, "s_2"] \arrow[r, shift right, "t_2"'] & V_2 \arrow[r, "\mu_{2,V}"] & M_V
\end{tikzcd}$$

Нечеткие графы с множеством их морфизмов и определенной покомпонентно композицией морфизмов $g\circ f=\langle g_E\circ f_E, g_V\circ f_V, g_{ME}\circ f_{ME}, g_{MV}\circ f_{MV}\rangle$, где $g_E, f_E\colon E_1\to E_2$, $g_V, f_V\colon V_1\to V_2$, образуют категорию **FGraph**.

Представление знаний на основе нечетких графов требует модели, в которой вершины являются набором значений нечетких атрибутов. С этой целью категория нечетких графов расширена до категории **FGraph**(L,U), в объектах которой элементы множества истинности, соответствующего меткам вершин и ребер, являются элементами произвольно выбранной решетки, а не только интервала [0,1]. Пусть L — полная решетка значений истинности, U (называемое пространством атрибутов) — произвольно выбранное множество. (L,U)-нечеткий граф G — направленный граф, в котором каждое ребро e помечено значением из множества L, а каждая вершина n помечена элементами L-значимого множества (U_n, ζ_n), где $U_n\subseteq U$ — множество элементарных утверждений, заданных в вершине n, а $\zeta_n\colon U_n\to L$ — функция, которая назначает значение из множества L каждому элементу U_n. Удалив метки вершины и ребер G, получим граф, представляющий собой «четкую» модель программной архитектуры (граф носителя нечеткого графа G).

Теоретико-категорный подход можно использовать для формализации СТНГ, которые являются обобщением последовательных СТГ и учитывают основные виды нечеткости, возникающие как при построении базовых категорий нечетких объектов, так и при описании трансформаций нечетких графов, порождаемых нечеткими множествами [7–9]. Для этого используем конструкцию «выдвижения», являющуюся категорным обобщением объединения двух объектов с учетом общего подобъекта. Выдвижение — специальный вид более общей конструкции, известной как копредел.

Выдвижением пары морфизмов $f\colon C\to A$, $g\colon C\to B$ в категории $\wp$ является $\wp$-объект P с парой морфизмов $j\colon A\to P$, $k\colon B\to P$ такой, что $j\circ f=k\circ g$; для любого $\wp$-объекта P' и пары морфизмов $j'\colon A\to P'$, $k'\colon B\to P'$, удовлетворяющих $j'\circ f=k'\circ g$, существует такой уникальный морфизм $h\colon P\to P'$, что коммутирует следующая диаграмма:

$$\begin{tikzcd}
 & & P' \\
A \arrow[r, "j"'] \arrow[urr, "j'"] & P \arrow[ur, dashed, "h"] & \\
C \arrow[u, "f"] \arrow[r, "g"'] & B \arrow[u, "k"] \arrow[uur, "k'"'] &
\end{tikzcd}$$

Введем понятие продукции над нечеткими графами $p=(I \xrightarrow{l} L; I \xrightarrow{r} R)$, состоящей из нечетких графов L, I, R, называемых левым, промежуточным и правым графом соответственно, а также двух инъективных морфизмов l, r. Для продукции p обратная продукция определяется как $p^{-1}=(I \xrightarrow{r} R; I \xrightarrow{l} L)$. Для такой продукции, нечеткого графа F_1, морфизма нечетких графов $m: L \to F_1$ (называемого совпадением нечетких графов) непосредственная трансформация $\Rightarrow: F_1 \xrightarrow{p,m} F_2$ задается диаграммой, в которой (1) и (2) — конструкции «выдвижения» в категории **FGraph**:

$$\begin{tikzcd}
L \arrow[d, "m"'] \arrow[dr, phantom, "(1)"] & I \arrow[l, "l"'] \arrow[r, "r"] \arrow[d, "k"] \arrow[dr, phantom, "(2)"] & R \arrow[d, "n"] \\
F_1 & D \arrow[l, "f"'] \arrow[r, "g"] & F_2
\end{tikzcd}$$

Последовательность $F_0 \Rightarrow F_1 \Rightarrow \ldots \Rightarrow F_n$ непосредственных трансформаций нечетких графов обозначим $F_0 \overset{*}{\Rightarrow} F_n$. Кроме того, для $n=0$ получаем изоморфизм $F_0 \cong F_n$, поскольку конструкция выдвижения и соответственно непосредственная трансформация уникальны с точностью до изоморфизма.

Система трансформаций нечетких графов (P, M, S) состоит из множества продукций над нечеткими графами P, набора нечетких множеств M и начального (нечеткого) графа S. Нечеткий язык FL СТНГ определен как

$$FL=\{F \mid \exists \text{ трансформация } S \overset{*}{\Rightarrow} F\}.$$

Для продукции $p=(I \xrightarrow{l} L; I \xrightarrow{r} R)$ и совпадения $m: L \to F_1$ результат непосредственной трансформации может быть получен в три этапа:

1) удалить вершины и ребра в F_1, которые находятся в L, но отсутствуют в интерфейсном графе I, т.е. $D=(F_1 \setminus m(L)) \cup m(l(I))$; таким образом, будут построены промежуточный граф D и выдвижение (1) такое, что $F_1 = L +_I D$;

2) добавить вершины и ребра, указанные только в R, т.е. $F_2 = D \cup (R \setminus r(I))$, точнее, построить выдвижение D и R по I такое, что $F_2 = R +_I D$;

3) переопределить функции принадлежности нечетких множеств для вершин и ребер результирующего нечеткого графа на основе трех типов вывода: монотонного восходящего, немонотонного и нисходящего; адекватность применения определенного типа вывода зависит от задачи и может отличаться для разных продукций, соответствующих определенным этапам решения.

При монотонном выводе существующие значения истинности вершин и ребер графа не могут быть уменьшены при наличии дополнительных свидетельств. Если NC — множество вершин, EC — множество ребер графа-консеквента, то формула монотонного вывода для значений истинности вершин и ребер, которые добавляются или остаются в соответствии с продукцией, имеет вид $\forall\, o \in NC \cup EC$: $\mu'(V'(o), x) = S(P, \mu(V(o), x))$, где функция S — так называемая S-норма, например $S(P, \mu(V(o), x)) = \max(P, \mu(V(o), x))$.

При немонотонном выводе допустимо, что новые результаты, обеспечиваемые запуском продукции, надежнее любого существующего свидетельства: $\forall\, o \in NC \cup EC$: $\mu'(V'(o), x) = P$. Нисходящий монотонный вывод эффективен, когда значение истинности $\mu'(V', x)$ есть верхний предел возможного значения: $\forall\, o \in NC \cup EC$: $\mu'(V'(o), x) = T(P, \mu(V(o), x))$, где функция T — любая T-норма, например $T(P, \mu(V(o), x)) = \max(P, \mu(V(o), x))$, что соответствует оператору вывода Мамдани.

Два альтернативных непосредственных порождения являются параллельно независимыми, если каждое из них может еще применяться после того, как другое было выполнено. Это означает, что ни $F \overset{p_1, m_1}{\Rightarrow} H_1$, ни $F \overset{p_2, m_2}{\Rightarrow} H_2$ не может удалить элементы G, которые также необходимы другому непосредственному порождению.

Другими словами, перекрытие левых сторон $p_1 = (I_1 \xrightarrow{l_1} L_1;\ I_1 \xrightarrow{r_1} R_1)$ и $p_2 = (I_2 \xrightarrow{l_2} L_2; I_2 \xrightarrow{r_2} R_2)$ в G должно войти в пересечение соответствующих графов интерфейса I_1 и I_2: $m_1(L_1) \cap m_2(L_2) \subseteq m_1(l_1(I_1)) \cap m_2(l_2(I_2))$.

Два непосредственных порождения $F \overset{p_1,\, m_1}{\Rightarrow} H_1$ и $F \overset{p_2,\, m_2}{\Rightarrow} H_2$ — параллельно независимы, если и только если существуют морфизмы $i: L_1 \to D_2$ и $j: L_2 \to D_1$ такие, что $f_2 \circ i = m_1$ и $f_1 \circ j = m_2$:

$$\begin{tikzcd}
R_1 \arrow[d, "m_1"'] & I_1 \arrow[l, "r_1"'] \arrow[r, "l_1"] \arrow[d, "d_1"] & L_1 \arrow[dr, "m_1"] \arrow[drrr, bend left, "i"] & & L_2 \arrow[dl, "m_2"'] \arrow[dlll, bend right, "j"'] & I_2 \arrow[l, "l_2"'] \arrow[r, "r_2"] \arrow[d, "d_2"] & R_2 \arrow[d, "m_2^*"] \\
H_1 & D_1 \arrow[l, "g_1"'] \arrow[rr, "f_1"] & & F & & D_2 \arrow[ll, "f_2"'] \arrow[r, "g_2"] & H_2
\end{tikzcd}$$

Доказано, что для параллельно независимых продукций $p_1 = (I_1 \xrightarrow{l_1} L_1;$ $I_1 \xrightarrow{r_1} R_1)$ и $p_2 = (I_2 \xrightarrow{l_2} L_2; I_2 \xrightarrow{r_2} R_2)$ следующие утверждения эквивалентны [10]:

1) существует непосредственный параллельный вывод $G \overset{p_1 + p_2,\, m}{\Rightarrow} X$;

2) существует последовательно независимый вывод $G \overset{p_1,\, m}{\Rightarrow} H_1 \overset{p_2,\, m_2'}{\Rightarrow} X$.

Итак, две параллельно независимые продукции над нечеткими графами можно применять в произвольном порядке. В работе [10] представлен метод декомпозиции каждой непосредственной параллельной продукции на последовательное применение продукций-компонентов; продемонстрировано, что каждый последовательно независимый вывод можно преобразовать в непосредственный параллельный вывод СТНГ.

Выбор соответствующего решения среди нескольких альтернативных продукций осуществляется в нечетком пространстве моделирования с учетом многочисленных параметров оценки (атрибутов вершин). Веса и оценка каждого параметра задаются нечеткими числами или словами, поэтому реализован метод нечеткого многоатрибутного принятия решений [8].

На основе категорного подхода формализованы преобразования *FD*-графов [9] как структурированные компонентные преобразования нечетких графов. Формально определены необходимые и достаточные условия для осуществления преобразований *FD*-графов, при которых не нарушается целостность их структуры и возможно конструктивное (покомпонентное) построение выдвижения в категории **FDGraph**.

Распределенные СТНГ, в свою очередь, являются обобщением СТНГ, что позволяет описывать допустимые преобразования *FD*-графов [9]. Изменение формы графа в течение шага преобразования дает возможность моделировать динамические сетевые структуры компонентов, задающие архитектуру нечеткой системы. При этом в одном правиле *FD*-грамматики возможно задавать изменение структуры графа одновременно в нескольких разных распределенных компонентах: например, параллельно выполняются нечеткие правила СТНГ как компонента-клиента, так и компонента, обеспечивающего сервис.

Формальные модели СТНГ используются в проектировании нечетких систем с двумя целями: как обобщение на нечеткие графы алгоритмов нечеткого логического вывода, в частности таких, как системы Мамдани и Ларсена; как спецификация преобразований между моделями, задающими разные уровни абстракции в подходе MDA («вертикальные» трансформации, преобразования «уточнения» и др.).

Обе модели — платформно-независимая и уточненная — представляют разные уровни абстракции платформы, поэтому корректное соотношение между ними определено как морфизм конкретизации модели. Для каждого шага трансформации $p = (F_1 \Rightarrow F_2)$ в платформно-независимой модели, который задается нечеткой графовой грамматикой $\wp^{\text{нез}}$, последовательность применения правил $p^{\text{ут}} = (F_1^{\text{ут}} \overset{*}{\Rightarrow} F_2^{\text{ут}})$ нечеткой графовой грамматики $\wp^{\text{ут}}$ сервис-ориентированной модели представляет собой корректную конкретизацию шага p, если $F_1^{\text{ут}}$, $F_2^{\text{ут}}$ — структурное уточнение нечетких графов F_1 и F_2 соответственно. Данный принцип используется для проверки корректности спецификации модели нечеткой системы.

Инструментарий теоретико-категорной формализации СТНГ имеет библиотеку прикладных классов и набор примеров преобразований, разработанных в среде JDK. Библиотека включает основные классы для создания нечетких множеств, нечетких отношений, нечетких графов и продукций; основные методы (операции) для композиционного создания объектов выделенных классов и выполнения правил трансформации с последующим выводом приобретенных нечетких значений и их дефуззификацией. Выделен ряд уровней (стратифицированной модели) абстракции для упрощения построения моделей высшего уровня, которые реализуются в виде объектов соответствующих категорий и морфизмов низшего уровня.

Для строгой формализации свойств СТНГ использованы объектно-ориентированные определения абстрактных классов категорий (в том числе основной категории множеств), их объектов и морфизмов. Нечеткие множества содержат классы для вычислений на основе дискретных множеств и соответствующий набор морфизмов. Каждый объект строится на основе пары объектов четких множеств — носителя и истинности. Нечеткие графы и их морфизмы используются для построения нечетких продукций.

Согласно предложенному подходу класс Production правил трансформации нечетких графов (продукций) позволяет создавать каждую такую продукцию на основе трех нечетких графов — L, I, R: Production rel = new Production («rel», L, I, R). Здесь L — граф-антецедент в левой части продукции, R — граф-результат в правой части, I — промежуточный граф, позволяющий определить место присоединения вершин $(R-I)$, которые добавляются к удаляемым вершинам $(L-I)$. Граф-результат определяется при выполнении продукции с использованием связанного с ней объекта-исполнителя класса FuzzyRuleExecutor.

ЗАКЛЮЧЕНИЕ

Обобщение моделей, представленных нечеткими мультиграфами, заключается в определении методологии формализации знаний, которые проектируются в нечетких интеллектуальных системах. Формализация нечетких объектно-ориентированных структур позволяет ввести и анализировать разные способы представления нечеткости в UML-диаграммах классов как основы спецификации знаний в такой системе. Разработанные средства теоретико-категорной трансформации нечетких систем упростили реализацию единого инструментария для анализа их свойств и проверки правильности в рамках моделе-ориентированного подхода.

СПИСОК ЛИТЕРАТУРЫ

1. Рутковская Д., Пилиньский М., Рутковский Л. Нейронные сети, генетические алгоритмы и нечеткие системы. — М.: Горячая линия –Телеком, 2004. — 452 с.
2. Kleppe A., Warmer J., Bast W. MDA Explained. The model driven architecture: practice and promise. — New York: Addison-Wesley, 2003. — 192 p.
3. ISO/IEC 10746-1:1998. Information Technology. — Open Distributed Processing – Reference Model: Overview.
4. Dictionary of information science and technology. — London; Melbourne; Singapore: IDEA Group Reference, 2007. — 770 p.
5. Zhang D.-Q., Zhang K., Cao J. A context-sensitive graph grammar formalism for the specification of visual languages // Computer J. — 2001. — **44**, N 3. — P. 186–200.
6. Буч Г., Якобсон А., Рамбо Дж. UML. — 2-е изд. — СПб.: Питер, 2006. — 736 с.
7. Парасюк И.Н., Ершов С.В. Категорный подход к построению нечетких графовых грамматик // Кибернетика и системный анализ. — 2006. — № 4. — С. 80–96.
8. Парасюк И.Н., Ершов С.В. Трансформационный подход к разработке программных архитектур на основе нечетких графовых моделей // Там же. — 2008. — № 5. — С. 139–150.
9. Парасюк И.Н., Ершов С.В. О трансформации нечетких графов, задаваемых *FD*-грамматиками // Там же. — 2007. — № 2. — С. 129–147.
10. Ершов С.В. Формализация параллельного вывода в нечетких графовых грамматиках // Компьютерная математика. — К.: Ин-т кибернетики им. В.М. Глушкова НАН Украины, 2007. — № 1. — С. 43–54.

Поступила 07.07.2009