

УДК 004.891.3

**НЕКОТОРЫЕ АСПЕКТЫ ПОВЫШЕНИЯ ЭФФЕКТИВНОСТИ ПРИНЯТИЯ РЕШЕНИЯ ИНТЕЛЛЕКТУАЛЬНЫМИ СИСТЕМАМИ, ИСПОЛЬЗУЮЩИМИ НЕЧЕТКИЙ ВЫВОД**

**Долинина О.Н., Шварц А.Ю.**

*ФГБОУ ВПО «Саратовский государственный технический университет им. Гагарина Ю.А.», Саратов, e-mail: olga@sstu.ru*

Рассмотрены продукционные базы знаний экспертных систем, использующих нечеткий вывод. Приведен сравнительный анализ различных подходов и алгоритмов принятия решений на основе нечеткой логики. Описанные методы нечеткого вывода могут быть выполнены параллельно на нескольких вычислительных узлах, что может значительно снизить временные затраты на принятие решений в интеллектуальных системах. Для повышения эффективности параллельных вычислений при принятии решения предлагается построение графа зависимостей правил в базе знаний и зависимостей лингвистических переменных в правилах. Обработка правил и переменных предполагает предварительное разбиение на группы. На основе сформированных групп выполнения правил и дефаззификации лингвистических переменных планируется повысить эффективность процесса принятия решений за счет применения параллельных вычислений в пределах каждой группы, что актуально для баз знаний большой размерности.

**Ключевые слова:** экспертные системы, нечеткий вывод, повышение эффективности принятия решения, алгоритмы построения зависимостей правил

**SEVERAL ASPECTS OF INCREASING OF THE EFFECTIVENESS OF THE MAKING DECISIONS BY INTELLIGENT SYSTEMS USING FUZZY INFERENCE**

**Dolinina O.N., Shvarts A.Y.**

*Yury Gagarin State Technical University of Saratov, Saratov, e-mail: olga@sstu.ru*

There are considered rule-based expert systems using fuzzy inference. Comparative analysis of different approaches and algorithms of making decisions on the base of fuzzy logic is given. Described methods can be processed on several computational nodes in parallel what can reduce time of making decision by intellectual systems. Building of graph of the dependence of the rules and the graph of dependence of the linguistic variables are suggested. Rules and variables should be divided into groups previously. On the base of the developed groups of rules and defuzzification of the linguistic variables it is planned to increase the effectiveness of the decision making with using of parallel calculations for each group what is considered to be an actual problem for the large dimension knowledge bases.

**Keywords:** expert systems, fuzzy inference, increasing of the effectiveness of the making decisions, algorithms of the building of the rules' dependence

Интеллектуальные системы являются одной из наиболее динамично развивающихся отраслей науки и техники. Это обусловлено не только растущими темпами накопления человечеством информации, но и необходимостью эффективно использовать аккумулируемые данные и знания для автоматизированного принятия решений, управления сложными техническими и иными системами, реализации новых видов взаимодействия между человеком и вычислительными устройствами.

Среди интеллектуальных систем наибольшую популярность получили экспертные системы (ЭС), позволяющие хранить, накапливать и использовать экспертные знания с целью автоматизации принятия решений в различных областях. Ядром ЭС является база знаний, включающая механизм принятия решений. Необходимо отметить, что с момента появления первых экспертных

систем в семидесятих годах двадцатого века до настоящего времени развитие экспертных систем прошло достаточно большой путь в смысле развития моделей представления знаний: от простейших продукционных моделей с детерминированными правилами вида до комбинации различных способов представления знаний, позволяющих учитывать особенности предметной области.

Представим базу нечетких правил в виде

$$(V, I, G, P, R), \tag{1}$$

где  $V$  – универсальное множество лингвистических переменных системы;  $I$  – множество входных лингвистических переменных,  $I \subset V$ ;  $G$  – множество выходных лингвистических переменных,  $G \subset V$ ;  $P$  – множество промежуточных лингвистических переменных,  $P \subset V$ ,  $I \cup G \cup P = V$ ,  $I \cap G = \emptyset$ ,  $I \cap P = \emptyset$ ,  $P \cap G = \emptyset$ ;  $R$  – множество правил вида

$$\langle r_j, u_j \rangle \text{ ЕСЛИ } v_{j,1} \text{ есть } t_{j,1} \text{ И...И } v_{j,n_j-1} \text{ есть } t_{j,n_j-1} \text{ ТО } v_{j,n_j} \text{ есть } t_{j,n_j} \tag{2}$$

где  $r_j$  – уникальное имя правила;  $u_j$  – лингвистическая степень уверенности в истинности правила  $r_j$ ;  $n_j$  – количество лингвистических переменных, задействованных в правиле;  $v_{j,1}, \dots, v_{j,n_j-1}$  – входные лингвистические переменные для правила  $r_j$ ; их множество обозначим  $G_j$ ;  $t_{j,1}, \dots, t_{j,n_j-1}$  – значения входных лингвистических переменных (термы) для правила  $r_j$ ;  $v_{j,n_j}$  – выходная лингвистическая переменная для правила  $r_j$ ;  $t_{j,n_j}$  – значение выходной лингвистической переменной для правила  $r_j$ .

Чаще всего эксперт не может однозначно трактовать те ли иные факты и их влияние на принятие решения. Среди методов, позволяющих обрабатывать неточные знания, часто используется аппарат нечеткой логики, введенный в классических работах

Заде [11,12]. Нечеткое множество  $A$  может быть представлено в виде

$$A = \{(x, \mu_A(x)) \mid x \in X\}, \quad (3)$$

где  $X$  – универсальное множество;  $x$  – элемент множества  $X$ ;  $\mu_A(x)$  – функция принадлежности.

Функция принадлежности  $\mu_A(x)$  характеризует степень принадлежности каждого  $x \in X$  нечеткому множеству  $A$ , причем  $\mu: X \rightarrow [0, 1]$ . Таким образом,  $\mu_A(x)$  принимает значение 1 в случае, если элемент  $x$  полностью включен во множество  $A$ , и значение 0 в случае, если элемент  $x$  не включен во множество  $A$ . Значения функции принадлежности, лежащие на интервале  $(0, 1)$ , характеризуют нечетко включенные элементы.

Общая схема механизма принятия решений в ЭС, основанных на нечетком выводе, представлена на рисунке.

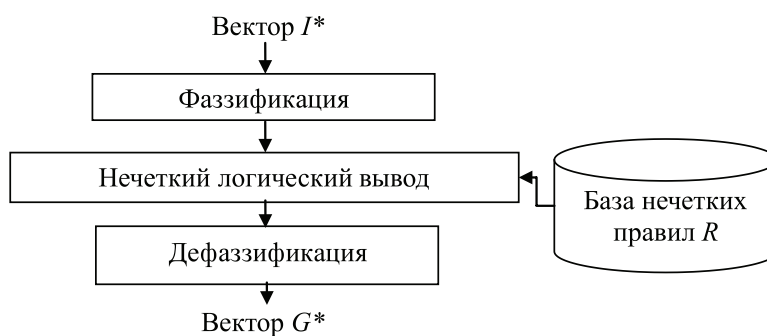


Схема системы нечеткого логического вывода

В литературе описано несколько различных подходов и алгоритмов принятия решений с использованием нечеткого вывода. Стоит отметить, что каждый из описанных в литературе алгоритмов [3–10] отличается не только характером и последовательностью операций, выполняемых в ходе фаззификации, нечеткого логического вывода и дефаззификации, но и видом используемых нечетких правил.

Методы Мамдани, Ларсена, Цукамото и product-sum-gravity оперируют нечеткими правилами, вид которых наиболее близок образу представления знаний человеком. Следовательно, формирование базы правил экспертом в случае использования этих методов будет наименее трудозатратным.

Методы Сугено, синглтона, упрощенного нечеткого вывода и SIRMs требуют от экспертов определения четких числовых значений и/или коэффициентов, что во многих предметных областях является нетривиальной задачей. Однако эти методы используют лишь простейшие арифметические операции, что дает выигрыш в вычисли-

тельной сложности по сравнению с первой группой методов.

Метод Цукамото накладывает требования к монотонности функций принадлежности выходных переменных, что во многих системах недопустимо.

Метод SIRMs требует при формировании базы правил группировать их в модули, кроме того, каждое правило должно содержать только одну входную переменную.

Отметим, что современные интеллектуальные системы, использующие механизмы нечеткого принятия решения, имеют базу знаний большой размерности, что требует использования специальных механизмов управления выводом для повышения эффективности принятия решения, в том числе для сокращения времени вывода. Описанные в литературе методы нечеткого вывода могут быть выполнены параллельно на нескольких вычислительных узлах, что может значительно снизить временные затраты на принятие решений в интеллектуальных системах. В данной статье предложены некоторые алгоритмы, позволяющие повысить

эффективность принятия решения для продукционных баз знаний, использующих нечеткий вывод.

В нечетких базах знаний большой размерности распространены ситуации, когда обработать некоторое правило невозможно, предварительно не обработав ряд других правил. Например, такое ограничение появляется, когда среди посылок правила встречаются переменные, не подаваемые на вход системы, а получаемые в ходе выполнения других правил.

Для решения описанной проблемы предлагается метод определения очередности выполнения правил в базах правил с описанными выше особенностями.

На первом этапе сгенерируем матрицу зависимостей правил. Рассмотрим следующий алгоритм.

Алгоритм генерации матрицы зависимостей правил

**Шаг 1.** Инициализировать нулями квадратную матрицу  $DR$  размерностью  $|R| \times |R|$  (где  $|R|$  – мощность множества  $R$ ).

**Шаг 2.** Для каждого  $r_j \in R$  выполнить Шаг 3.

**Шаг 3.** Для каждого  $r_q \in R, r_q \neq r_j$  выполнить Шаг 4.

**Шаг 4.** Если  $v_{j,n_j} \in G_q$ , то  $DR_{j,q} = 1$  и  $DR_{q,j} = -1$ .

Полученная матрица  $DR$  является матрицей смежности для ориентированного графа зависимостей правил. Данный граф может быть использован для определения того, от обработки каких правил зависит выбор каждого конкретного правила в базе знаний.

Каждому правилу  $r_j \in R$  поставим в соответствие значение функции  $\rho(r_j)$ , определяющее порядковый номер группы выполнения правил. Порядковый номер группы имеет следующий смысл: правило  $r_{j_1}$  не может быть выполнено раньше правила  $r_{j_2}$ , если  $\rho(r_{j_2}) < \rho(r_{j_1})$ .

**Алгоритм расчета функции  $\rho(r)$ , определяющей порядковый номер группы выполнения правил**

**Шаг 1.** Инициализировать нулями множество  $RO, |RO| = |R|$ .

**Шаг 2.**  $j = 1$ .

**Шаг 3.** Если  $j \leq |R|$ , то перейти к Шагу 4, иначе перейти к Шагу 10.

**Шаг 4.**  $i = 1$ .

**Шаг 5.** Если  $i \leq |R|$ , то перейти к Шагу 6, иначе перейти к Шагу 8.

**Шаг 6.** Если  $DR_{j,i} = -1$ , то перейти к Шагу 9.

**Шаг 7.**  $i = i + 1$ .

**Шаг 8.**  $ro_j = 1$ .

**Шаг 9.**  $j = j + 1$ . Перейти к шагу 3.

**Шаг 10.**  $j = 1, c = 0$ .

**Шаг 11.** Если  $j \leq |R|$ , то перейти к Шагу 12, иначе перейти к Шагу 20.

**Шаг 12.**  $i = 1, m = 1$ .

**Шаг 13.** Если  $i \leq |R|$ , то перейти к Шагу 14, иначе перейти к Шагу 17.

**Шаг 14.** Если  $DR_{j,i} = -1$ , то перейти к Шагу 15, иначе перейти к Шагу 16.

**Шаг 15.** Если  $ro_i > m$ , то  $m = ro_i$ .

**Шаг 16.**  $i = i + 1$ . Перейти к Шагу 13.

**Шаг 17.** Если  $ro_j < m + 1$ , то  $c = 1$ .

**Шаг 18.**  $ro_j = m + 1$ .

**Шаг 19.**  $j = j + 1$ . Перейти к Шагу 11.

**Шаг 20.** Если  $c = 0$ , то перейти к Шагу 21, иначе перейти к Шагу 10.

**Шаг 21.** Для каждого  $r_j: \rho(r_j) = ro_j$ .

На основе полученных значений  $\rho(r_j)$  формируются множества

$$RG_m: r_j \in RG_m \Leftrightarrow \rho(r_j) = m, m = \overline{1, \dots, L_R},$$

где  $L_R$  – количество групп выполнения правил.

В ходе нечеткого логического вывода вычисляются не только результирующие функции принадлежности каждого отдельного правила  $r_j$ , но и происходит дефаззификация лингвистических переменных. Очередность дефаззификации далеко не всегда связана с зависимостью между правилами базы знаний, т.к. каждая переменная может содержаться в посылках нескольких правил с различающимися порядковыми номерами групп выполнения  $\rho(r_j)$ .

Таким образом, возникает необходимость также определить наличие зависимостей между лингвистическими переменными. Рассмотрим следующий алгоритм генерации матрицы зависимостей лингвистических переменных.

**Шаг 1.** Инициализировать нулями квадратную матрицу  $DV$  размерностью  $|V| \times |V|$ .

**Шаг 2.** Для каждого  $v_k \in V$  выполнить Шаг 3.

**Шаг 3.** Для каждого  $v_h \in V, v_h \neq v_k$  выполнить Шаг 4.

**Шаг 4.** Для каждого  $r_j \in R$  выполнить Шаг 5.

**Шаг 5.** Если  $v_k \in G_j$  и  $v_h = v_{j,n_j}$  то  $DV_{k,h} = 1$  и  $DV_{h,k} = -1$ .

Полученная матрица  $DV$  может быть интерпретирована как матрица смежности ориентированного графа зависимости лингвистических переменных.

Каждой лингвистической переменной  $v_k \in V$  поставим в соответствие значение функции  $\rho(v_k)$ , определяющее порядковый номер группы дефазификации переменной. Порядковый номер группы имеет следующий смысл: переменная  $v_{k_1}$  не может быть дефазифицирована раньше переменной  $v_{k_2}$ , если  $\rho(v_{k_2}) < \rho(v_{k_1})$ .

#### Алгоритм расчета функции $\rho(v_k)$ , определяющей порядковый номер группы дефазификации переменной

**Шаг 1.** Инициализировать нулями множество  $VO$ ,  $|VO| = |V|$ .

**Шаг 2.**  $k = 1$ .

**Шаг 3.** Если  $k \leq |V|$ , то перейти к Шагу 4, иначе перейти к Шагу 10.

**Шаг 4.**  $h = 1$ .

**Шаг 5.** Если  $h \leq |V|$ , то перейти к Шагу 6, иначе перейти к Шагу 8.

**Шаг 6.** Если  $DV_{k,h} = -1$ , то перейти к Шагу 9.

**Шаг 7.**  $h = h + 1$ .

**Шаг 8.**  $vo_k = 1$ .

**Шаг 9.**  $k = k + 1$ . Перейти к шагу 3.

**Шаг 10.**  $k = 1$ ,  $c = 0$ .

**Шаг 11.** Если  $k \leq |V|$ , то перейти к Шагу 12, иначе перейти к Шагу 20.

**Шаг 12.**  $h = 1$ ,  $m = 1$ .

**Шаг 13.** Если  $h \leq |V|$ , то перейти к Шагу 14, иначе перейти к Шагу 17.

**Шаг 14.** Если  $DV_{k,h} = -1$ , то перейти к Шагу 15, иначе перейти к Шагу 16.

**Шаг 15.** Если  $vo_h > m$ , то  $m = vo_h$ .

**Шаг 16.**  $h = h + 1$ . Перейти к Шагу 13.

**Шаг 17.** Если  $vo_k < m + 1$ , то  $c = 1$ .

**Шаг 18.**  $vo_k = m + 1$ .

**Шаг 19.**  $k = k + 1$ . Перейти к Шагу 11.

**Шаг 20.** Если  $c = 0$ , то перейти к Шагу 21, иначе перейти к Шагу 10.

**Шаг 21.** Для каждого  $v_k$ :  $\rho(v_k) = vo_k$ .

На основе полученных значений  $\rho(v_k)$  формируются множества

$$VG_m: v_k \in VG_m \Leftrightarrow \rho(v_k) = m, m = \overline{1, \dots, L_V},$$

где  $L_V$  – количество групп выполнения переменных.

Подразумевается, что каждая переменная, для которой  $\rho(v_k) = 1$ , является входной переменной ( $v_k \in I$ ) и не нуждается в дефазификации.

На основе сформированных групп выполнения правил и дефазификации лингвистических переменных планируется повысить эффективность процесса принятия решений за счет применения параллельных вычислений в пределах каждой группы. Эффективная параллельная обработка правил и переменных невозможна без предварительного разбиения на группы, т.к. зависимости (в т.ч. транзитивные) между ними зачастую позволяют обработать два правила только последовательно. Указанные алгоритмы были успешно применены для реализации механизма нечеткого вывода в экспертной системе обнаружения неисправностей газоперекачивающих агрегатов GAZDETECT [1].

#### Список литературы

1. Долинина О.Н., Антропов П.Г., Кузьмин А.К., Шварц А.Ю. Использование интеллектуальных систем для диагностики неисправностей газоперекачивающих агрегатов // Современные проблемы науки и образования. – 2013. – № 6; URL: <http://www.science-education.ru/113-11252> (дата обращения: 23.12.2013).
2. Jang J.-S. R., ANFIS: Adaptive-Network-Based Fuzzy Inference System // IEEE Transactions on Systems & Cybernetics. – 1993. – Vol. 23. – P. 665–685.
3. Larsen P.M., Industrial Applications Of Fuzzy Logic Control // International Journal Of Man-Machine Studies. – 1980. – Vol. 12. – P. 3–10.
4. Mamdani E.H., Assilian S., An Experiment In Linguistic Synthesis With Fuzzy Logic Controller // International Journal Of Man-Machine Studies. – 1975. – Vol. 7, № 1. – P. 1–13.
5. Mizumoto M., Fuzzy controls under various fuzzy reasoning methods // Information Science. – 1988. – Vol. 45. – P. 129–151.
6. Seki H., Mizumoto M., On the Equivalence Conditions of Fuzzy Inference Methods – Part 1: Basic Concept and Definition // IEEE Transactions on Fuzzy Systems. – 2011. – Vol. 9, no. 6. – P. 1097–1106.
7. Sugeno M. On stability of fuzzy systems expressed by fuzzy rules with singleton consequents // IEEE Transactions on Fuzzy Systems. – 1999. – Vol. 7, № 2. – P. 201–224.
8. Tsukamoto Y., An Approach To Fuzzy Reasoning Method // Gupta M.M. Et Al (Eds.), Advances In Fuzzy Set Theory And Applications. – 1979. – P. 137–149.
9. Takagi T., Sugeno M., Fuzzy Identification Of Systems And Its Applications To Modeling And Control // IEEE Transactions On Systems, Man, And Cybernetics. – 1985. – Vol. 15, № 1. – P. 116–132.
10. Yubazaki N., Yi J., Hirota K., SIRMs (Single Input Rule Modules) connected fuzzy inference model // Journal of Advanced Computational Intelligence and Intelligent Informatics. – 1997. – Vol. 1, № 1. – P. 23–30.
11. Zadeh L.A., Fuzzy Sets // Information and Control. – 1965. – Vol. 8. – P. 338–353.
12. Zadeh L.A., The Concept Of A Linguistic Variable And Its Application To Approximate Reasoning // Information Sciences. – 1975. – Vol. 8. – P. 199–249, 301–357; Vol. 9, P. 43–80.

**References**

1. Dolinina O.N., Antropov P.G., Kuz'min A.K., Shvarc A.Ju. Ispol'zovanie intellektual'nyh sistem dlja diagnostiki neispravnostej gazoperekachivajushhh agregatov // *Sovremennye problemy nauki i obrazovanija*. 2013. no. 6; URL: <http://www.science-education.ru/113-11252> (data obrashhenija: 23.12.2013).
2. Jang J.-S. R., ANFIS: Adaptive-Network-Based Fuzzy Inference System // *IEEE Transactions on Systems & Cybernetics*, vol. 23, pp. 665–685, 1993.
3. Larsen P.M., Industrial Applications Of Fuzzy Logic Control // *International Journal Of Man-Machine Studies*, Vol. 12, pp. 3–10, 1980.
4. Mamdani E.H., Assilian S., An Experiment In Linguistic Synthesis With Fuzzy Logic Controller // *International Journal Of Man-Machine Studies*, Vol. 7, no. 1, pp. 1–13, 1975.
5. Mizumoto M., Fuzzy controls under various fuzzy reasoning methods // *Information Science*, Vol. 45, pp. 129–151, 1988.
6. Seki H., Mizumoto M., On the Equivalence Conditions of Fuzzy Inference Methods – Part 1: Basic Concept and Definition // *IEEE Transactions on Fuzzy Systems*, Vol. 9, no. 6, pp. 1097–1106, 2011.
7. Sugeno M. On stability of fuzzy systems expressed by fuzzy rules with singleton consequents // *IEEE Transactions on Fuzzy Systems*, Vol. 7, no. 2, pp. 201–224, 1999.
8. Tsukamoto Y., An Approach To Fuzzy Reasoning Method // Gupta M.M. Et Al (Eds.), *Advances In Fuzzy Set Theory And Applications*, pp. 137–149, 1979.
9. Takagi T., Sugeno M., Fuzzy Identification Of Systems And Its Applications To Modeling And Control // *IEEE Transactions On Systems, Man, And Cybernetics*, Vol. 15, no. 1, pp. 116–132, 1985.
10. Yubazaki N., Yi J., Hirota K., SIRMs (Single Input Rule Modules) connected fuzzy inference model // *Journal of Advanced Computational Intelligence and Intelligent Informatics*, Vol. 1, no. 1, pp. 23–30, 1997.
11. Zadeh L.A., Fuzzy Sets // *Information and Control*, Vol. 8, pp. 338–353, 1965.
12. Zadeh L.A., The Concept Of A Linguistic Variable And Its Application To Approximate Reasoning // *Information Sciences*, Vol. 8, pp. 199–249, 301–357; Vol. 9, pp. 43–80, 1975.

**Рецензенты:**

Кушников В.А., д.т.н., профессор, ведущий научный сотрудник, зам. директора по научно-организационной работе, Институт проблем точной механики и управления РАН, г. Саратов;

Шульга Т.Э., д.т.н., зав. кафедрой «Прикладная информатика и программная инженерия», международный факультет прикладных информационных технологий, Саратовский государственный технический университет им. Гагарина Ю.А., г. Саратов.

Работа поступила в редакцию 16.12.2014.