

МЕТОДЫ ПРОЕКТИРОВАНИЯ НЕЧЕТКОЙ БАЗЫ ЗНАНИЙ

М. А. Сергиенко

Воронежский государственный университет

Поступила в редакцию 20.05.2008 г.

Аннотация. В данной статье будет рассмотрен вопрос обеспечения полноты и непротиворечивости нечеткой базы правил, представлен способ организации ее структуры с помощью иерархической системы лингвистических правил. В такой системе лингвистические переменные, ассоциированные с нечеткими правилами, располагаются по секторам с различными степенями гранулярности. Иерархия в этой структуре задается с помощью этих степеней.

Ключевые слова: нечеткая система управления, полнота и непротиворечивость базы нечетких правил, иерархическая система лингвистических правил.

Abstract. In the article, question of supporting fullness and consistency of hierarchical knowledge base will be considered. A single method will be introduced belong to the group of approaches inducing cooperation from the whole knowledge base (i.e. both from the data base and the rule base), the hierarchical system of linguistic rules learning methodology. In the system, the linguistic variables involved in the fuzzy rules are defined in linguistic partitions with different granularity levels, thus making the rules belong to different granularity.

Key words: fuzzy control system, fullness and consistency of fuzzy rule base, hierarchical system of linguistic rules.

ВВЕДЕНИЕ

Нечеткая система управления (НСУ) — это интеллектуальная система, использующая нечеткое описание управляемого процесса и системы его управления в виде базы нечетких правил для генерации последовательности управляющих решений, обеспечивающих достижение целей управления [3]. Основной для построения НСУ служит схема управления с участием эксперта, который на основе опыта и знаний об управлении объектом формирует качественное описание процесса управления. Это описание преобразуется в базу правил и в дальнейшем используется в системе управления уже без участия эксперта. Идея нечеткого управления заключается именно в подражании действиям опытного человека-оператора.

Под продукцией будем понимать кортеж следующего вида

$$\langle i; Q; P; A \rightarrow B; N \rangle,$$

где i — имя продукции; Q характеризует сферу применения продукции; P — условие применимости ядра продукции; $A \rightarrow B$ — ядро продукции (обычное прочтение ядра выглядит как: если A , то B); N — постусловие продукции описывает действия и процедуры, которые

необходимо выполнить после реализации B . Основной частью продукции является ядро, остальные элементы носят вспомогательный характер, поэтому в наиболее простом виде продукция может состоять лишь из имени и ядра.

Нечеткие правила — это нечеткие продукционные правила, которые при фиксированной цели управления (например, сохранение значений управляемого параметра в некоторой области допустимых значений) описывают его стратегии на качественном уровне.

Актуальная проблема разработки НСУ — формирование базы знаний. Различные подходы к решению этой проблемы можно найти в [3]. Специальным классом баз знаний являются иерархические базы знаний. В данной статье представлен подход, позволяющий учитывать различную степень неопределенности (нечеткости) иерархическими продукционными правилами и регулировать размерность базы знаний.

1. ОБЕСПЕЧЕНИЕ ПОЛНОТЫ И НЕПРОТИВОРЕЧИВОСТИ БАЗЫ НЕЧЕТКИХ ПРАВИЛ

При создании нечеткой продукционной модели системы могут быть использованы как априорные данные о моделируемой системе,

поступающие от экспертов, так и данные, полученные в результате измерений.

В первом случае, если отсутствует необходимость согласования мнений экспертов, предполагается, что задача обеспечения полноты и непротиворечивости базы нечетких правил решена заранее. В случае же, если известны только экспериментальные данные, данную задачу можно отнести к задачам идентификации системы. На практике может иметь место также смешанный случай, когда начальная база нечетких правил строится исходя из эвристических предположений, а ее уточнение проводится с использованием экспериментальных данных. Каждому из этих подходов соответствуют свои алгоритмы.

Рассмотрим основные этапы типичного алгоритма формирования базы нечетких правил заданной структуры в случае, если применяются только экспериментальные данные (обучающая выборка).

Допустим, необходимо создать базу нечетких правил с MISO-структурой, т.е. с двумя входными (x_1, x_2) и одной выходной (y) переменными. В этом случае правила будут иметь вид

$$P_i : \text{ЕСЛИ } x_1 \text{ есть } A_{i1} \text{ И...И } x_j \text{ есть } A_{ij} \text{ И...И } x_m \text{ есть } A_{im},$$

$$\text{ТО } y \text{ есть } B_i (i = \overline{1, n}).$$

Пусть задана обучающая выборка, состоящая из множества примеров следующего вида

$$(x_1^{(k)}, x_2^{(k)}, y^{(k)}) (k = \overline{1, K}),$$

где $x_1^{(k)}, x_2^{(k)}, y^{(k)}$ — соответственно значения входных переменных x_1, x_2 и выходной переменной y в k -ом примере; K — общее число примеров в обучающей выборке.

База правил формируется на основе следующей процедуры.

Этап 1. Разбиение пространств входных и выходных переменных. Пусть известны минимальные и максимальные значения каждой переменной

$$x_1 \in [x_1^{(\min)}, x_1^{(\max)}],$$

$$x_2 \in [x_2^{(\min)}, x_2^{(\max)}],$$

$$y \in [y^{(\min)}, y^{(\max)}].$$

Разобьем области определений этих переменных на отрезки. Причем число этих отрезков, а также их длина для каждой переменной подбираются индивидуально. На рис. 1 показан пример такого разбиения, при условии, что

число отрезков для $[x_1^{(\min)}, x_1^{(\max)}]$ равно трем, для $[x_2^{(\min)}, x_2^{(\max)}]$ — пяти, для $[y^{(\min)}, y^{(\max)}]$ — четырем. На каждом из отрезков задана функция принадлежности, например треугольной формы с вершиной в центре отрезка. Для x_1 определены нечеткие множества с лингвистическими значениями $\{L_1 - \text{низкое}, M_1 - \text{среднее}, H_1 - \text{высокое}\}$; для $x_2 - \{L_2, LM_2, M_2, HM_2, H_2\}$; для $y - \{L_y, LM_y, HM_y, H_y\}$.

Функции принадлежности подбираются таким образом, чтобы они перекрывались на уровне 0.5. Следует отметить, что как вид функций принадлежности, так и способ разбиения пространств входных и выходных переменных могут быть иными.

Этап 2. Формирование начальной базы правил. Можно предложить два непротиворечивых друг другу подхода к формированию начальной базы правил. Первый подход основан на генерации множества правил исходя из возможных сочетаний нечетких высказываний в предпосылках и заключениях правил, в соответствии с которым максимальное количество правил в базе определяется следующим соотношением: $l = l_1 \cdot l_2 \cdot \dots \cdot l_m \cdot l_y$, где $l_1, l_2, \dots, l_m, l_y$ — число фун-

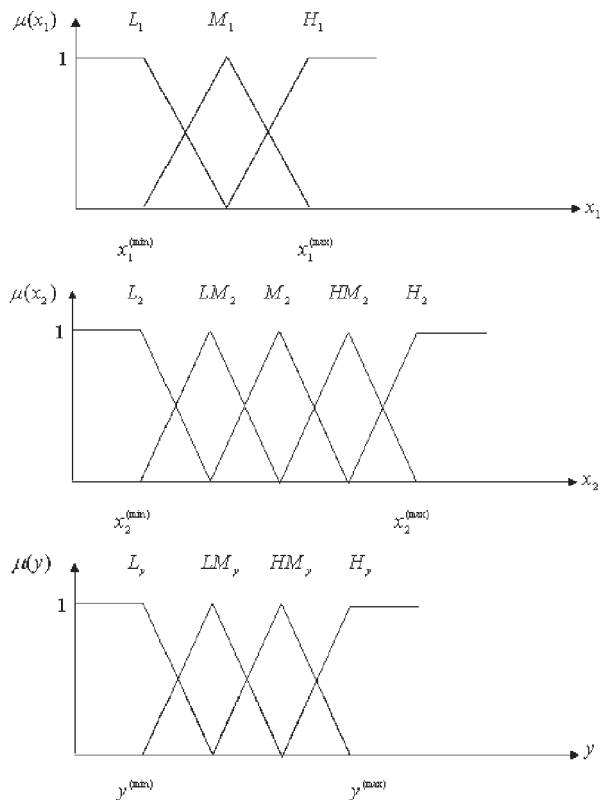


Рис. 1. Пример разбиения пространства входных/выходных переменных

кций принадлежности для задания входных/выходных переменных (x_1, \dots, x_m, y) .

Другой подход к формированию начальной базы правил основан на том, что изначально каждому примеру из выборки ставится в соответствие отдельное правило. Для этого для каждого $(x_1^{(k)}, x_2^{(k)}, y^{(k)})$ ($k = \overline{1, K}$) определяются степени принадлежности заданных значений переменных к соответствующим нечетким множествам. После чего каждому обучающему примеру ставятся в соответствие те нечеткие множества, степени принадлежности к которым у соответствующих значений переменных из этого примера являются максимальными. Сформированное таким образом множество правил и составляет начальную базу правил.

Следует отметить, что использование первого подхода при формировании начальной базы правил целесообразно при небольшом числе переменных и функций принадлежности, используемых для задания этих переменных. Второй подход стоит использовать при сравнительно небольшом количестве примеров в обучающей выборке.

Эман 3. Определение рейтингов правил. Поскольку изначально сформированная база правил наверняка является избыточной — с одинаковыми предпосылками и разными заключениями, то набор правил необходимо оптимизировать. Сделать это можно как на основе эмпирических гипотез (информации от экспертов), так и путем адаптации к имеющимся экспериментальным данным (обучающей выборке), что приводит к существенному уменьшению и к ликвидации противоречивости правил, оставаемых в базе.

Рассмотрим один из подходов к сокращению базы нечетких правил на основе экспериментальных данных.

Допустим набор экспериментальных данных (обучающая выборка) в полной мере характеризует особенности моделируемой системы.

Все примеры из обучающей выборки

$$(x_1^{(k)}, x_2^{(k)}, \dots, x_m^{(k)}, y^{(k)}) \quad (k = \overline{1, K})$$

«предъявляются» каждому правилу. В результате для каждого правила определяется его рейтинг

$$r_i = \underset{k}{\text{Agg}}(r_i^{(k)}) \quad (i = \overline{1, n}),$$

$$r_i^{(k)} = T(\mu_{A_1}(x_1^{(k)}), \dots, \mu_{A_m}(x_m^{(k)}), \mu_{B_i}(y^{(k)})),$$

$$(k = \overline{1, K}),$$

где Agg и T — оператор агрегирования и Т-норма соответственно [2].

Эман 4. Сокращение числа правил. После подсчета рейтингов правил из базы правил исключаются правила с наименьшими рейтингами. При этом в первую очередь сокращения проводятся по группам правил, имеющим одинаковые посылки и разные заключения, т.е. разные функции принадлежности переменной вывода. Такие правила являются противоречащими друг другу, и из группы подобных правил необходимо оставлять только одно (с наибольшим рейтингом). Таким образом, решается проблема противоречивых правил, а также существенно уменьшается их общее число. Оставшиеся правила формируют итоговую базу правил.

Эман 5. Адаптация параметров оставленных в базе правил. База правил может считаться окончательно сформированной, если осуществлен этап адаптации оставленных в ней правил. Этот этап сводится к нахождению, в соответствии с имеющимися экспериментальными данными и принятым критерием, оптимальных значений параметров для оставленных в базе правил. Этот этап, по сути, является этапом параметрической оптимизации конечного набора правил. Он заключается в таком изменении параметров функции принадлежности оставшихся правил, при котором обеспечиваются максимальные «степени адекватности» этих правил по всем примерам обучающей выборки.

Для рассматриваемого на рис. 1 случая, когда все функции принадлежности нечетких множеств являются треугольными, настройке подвергается один их параметр — значения мод.

Очевидно, что в зависимости от конкретной постановки задачи формирования базы нечетких продукционных правил могут быть изменены как перечень, так и содержание решаемых в ходе создания базы нечетких правил этапов.

2. ИЕРАРХИЧЕСКАЯ СИСТЕМА ЛИНГВИСТИЧЕСКИХ ПРАВИЛ

В иерархической системе лингвистических правил лингвистические переменные, ассоциированные с нечеткими правилами, располагаются по лингвистическим секторам с различными степенями гранулярности. Иерархия в этой структуре задается с помощью этих степеней.

Структура иерархической базы знаний расширяется за счет ввода понятия *уровня*. Каждый уровень определяется следующим образом:

$$HKB = HDB + HRB, \quad (1)$$

$$layer(t, n) = DB(t, n) + RB(t, n), \quad (2)$$

$$HDB = \bigcup_t DB(t, n), \quad HRB = \bigcup_t RB(t, n), \quad (3)$$

где $DB(t, n)$ — база данных уровня t , содержащая лингвистические сектора со степенью гранулярности n ; $RB(t, n)$ — база правил, сформированная для лингвистических переменных, принимающих значения в соответствующих секторах.

Пусть $DB(t, n)$ — n -лингвистические сектора, $RB(t, n)$ — n -лингвистические правила. Для простоты будем рассматривать лингвистические сектора с одинаковым количеством лингвистических термов для входных/выходных переменных. Для построения *иерархической базы данных* необходимо придерживаться следующей стратегии.

- Сохранять все возможные структуры нечеткого множества при переходе от уровня t к $t + 1$.

- Сглаживать переход между соседними уровнями, т.е. уменьшать MSE (значение квадратичной ошибки) [4].

Для построения лингвистического сектора DB уровня $t + 1$ из $DB(t, n)$ с минимальными изменениями между их степенями гранулярности, необходимо добавить по новому терму между соседними термами $DB(t, n)$ (рис. 2):

$$DB(t, n) \rightarrow DB(t + 1, 2n - 1). \quad (4)$$

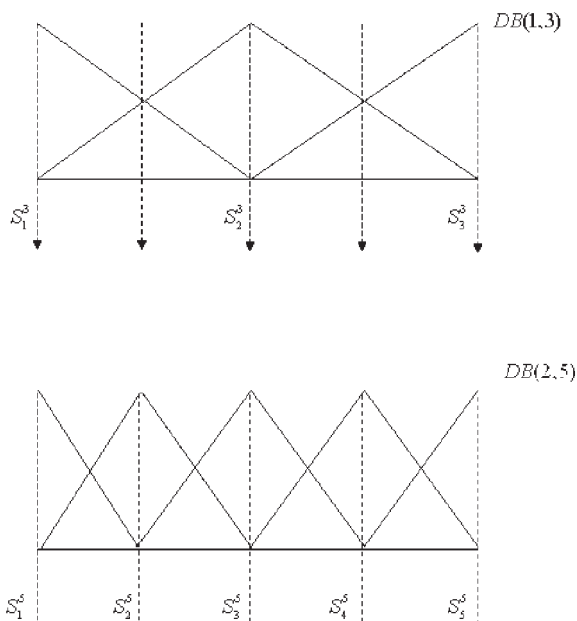


Рис. 2. Переход от уровня t к $t + 1$

Главной задачей при разработке *иерархической базы правил* является построение более точной модели функционирования рассматриваемого объекта предметной области. Правила из $RB(t, n)$, имеющие «плохое представление» (приводящие к значительной MSE), расширяются в $RB(t + 1, 2n - 1)$:

$$RB(t, n) \rightarrow RB(t + 1, 2n - 1). \quad (5)$$

Оставшиеся правила сохраняют свое $RB(t, n)$ -размещение в $RB(t + 1, 2n - 1)$. Необходимо отметить и тот факт, что множество правил с большой степенью гранулярности не всегда позволяет лучше построить модель функционирования рассматриваемого объекта.

Таким образом, (4)–(5) определяется следующим образом:

- выбор правил из $RB(t, n)$ с «плохим представлением», которые будут расширяться в $RB(t + 1, 2n - 1)$;
- подбор соответствующих термов для расширяемых правил и построение $DB(t + 1, 2n - 1)$.

Процесс построения *иерархической базы знаний* состоит из следующих шагов.

1) Генерация $RB(t, n)$:

- формируется $DB(t, n)$;
- применяется метод генерации лингвистических правил LRG [6] к термам, находящимся в заданных секторах:

$$LRG(DB(t, n), E_p),$$

где E_p — множество входных/выходных переменных.

2) Генерация $RB(t + 1, 2n - 1)$:

- вычисляется $MSE(E_p, RB(t, n))$ $RB(t, n)$;
- вычисляется $MSE(E_p, R_i^n)$ для каждого правила;
- выбираются правила из $RB(t, n)$ с «плохим представлением», которые будут расширяться в $RB(t + 1, 2n - 1)$:

$$\text{если } MSE(E_p, R_i^n) \geq \alpha \cdot MSE(E_p, RB(t, n)),$$

$$\text{то } R_i^n \in RB_{bad}, \text{ иначе } R_i^n \in RB_{good},$$

где RB_{bad}, RB_{good} — базы, содержащие правила с «плохим/хорошим представлением», α — признак расширяемости;

- получение $DB(t + 1, 2n - 1)$: построение $DB_{x_j}(t + 1, 2n - 1)$ для всех входных лингвистических переменных x_j ($j = 1, m$) и $DB_y(t + 1, 2n - 1)$ для выходной лингвистической переменной y ;

• комбинирование термов, которые использовались в правилах с «плохим представлением» и генерация новых правил.

3) Построение иерархической базы правил

$$HRB = RB_{good}(t, n) \cup RB(t + 1, 2n - 1).$$

Может получиться так, что вновь построенный набор $(2n - 1)$ лингвистических правил не даст нужного результата, хотя подмножество из $(2n - 1)$ -лингвистических правил позволит получить требуемый результат. В таком случае необходимо учитывать избыточность базы правил.

ЗАКЛЮЧЕНИЕ

Иерархическую структуру удобно использовать для представления динамических аспектов проблемной области, которые описываются с помощью продукционной модели представления знаний. Подобная конструкция позволяет разбить продукционные правила на блоки в соответствии с принадлежностью к элементам структуры и, тем самым, «сжать» базу знаний, сделав ее более компактной.

Рассмотренный алгоритм позволяет пользователю, задавая стартовую степень грануляр-

ности и α — признак расширяемости правил, добиваться нужного результата.

В дальнейшем предполагается исследовать зависимость предложенного алгоритма от размерности лингвистической шкалы, функционального представления нечетких операций, способов определения рейтингов правил, способы оптимизации структуры базы правил.

СПИСОК ЛИТЕРАТУРЫ

1. Новак В., Перфильева И., Мочкорж И. Математические принципы нечеткой логики / Пер. с англ.; Под ред. Аверкина А.Н. — М.: ФИЗМАТЛИТ, 2006. — 252 с.

2. Леденева Т.М. Обработка нечеткой информации: учебное пособие / Т.М. Леденева. — Воронеж: Воронежский государственный университет, 2006. — 233 с.

3. Kandel A., Langholz G. Fuzzy Control Systems // CRC Press LLC, 1993. — P. 187.

4. Cordon O., Herrera F. Linguistic Modeling by Hierarchical Systems of Linguistic Rules // Technical Report # DECSAI — 990114, Dept. of Computer Science and A. I., University of Granada, July, 1999.

5. Борисов В.В., Круглов В.В., Федулов А.С. Нечеткие модели и сети. — М.: Горячая линия — Телеком, 2007. — 284 с.

Сергиенко М. А. — старший специалист, представительство в г. Воронеже ЦР ТЭК ЗАО “Открытые Технологии”, E-mail: msergienko@ot.ru, fers2003@list.ru

Sergienko M. A. — senior speshialist , Voronezh, ” ЦР ТЭК ЗАО “Открытые Технологии”, E-mail: msergienko@ot.ru, fers2003@list.ru.