

Идентификация нелинейных зависимостей с помощью нечеткого логического вывода в системе MATLAB

1. Введение

Идентификация, т. е. построение математической модели по результатам наблюдений, является важной задачей, возникающей в технике, экономике, политике, медицине, биологии и в других областях. В современной теории идентификации все более важную роль начинают играть методы, привлекающие лингвистическую информацию при построении моделей нелинейных зависимостей. Одним из наиболее разработанных в инженерном отношении инструментов учета лингвистической информации является теория нечетких множеств и нечеткая логика. Теория нечетких множеств ведет свое начало с 1965 года, когда профессор Лотфи Заде из Калифорнийского университета Беркли опубликовал основополагающую статью «Fuzzy Sets» в журнале «Information and Control». Прилагательное «fuzzy», которое можно перевести на русский язык как нечеткий, размытый, ворсистый, пушистый, введено в название новой теории с целью дистанцирования от традиционной четкой математики и аристотелевой логики, оперирующих с четкими понятиями: «принадлежит—не принадлежит», «истина—ложь». Концепция нечеткого множества зародилась у Л. Заде «как неудовлетворенность математическими методами классической теории систем, которая вынуждала добиваться искусственной точности, неуместной во многих системах реального мира, особенно в так называемых гуманитарных системах, включающих людей» [1].

Работы по идентификации нелинейных зависимостей на основе нечетких множеств и нечеткой логики интенсивно проводятся за рубежом с 90-х годов. Среди русскоязычных публикаций выделим монографии и серию статей профессора А. П. Ротштейна [2–7], в которых разработан метод двухэтапной идентификации нелинейных зависимостей с помощью нечетких баз знаний. Первый этап — структурная идентификация — представляет собой формирование нечеткой базы знаний, которая грубо отражает взаимосвязь между входами и выходом с помощью лингвистических правил «ЕСЛИ–ТО». Лингвистические правила генерируются экспертом либо получаются в результате экстракции нечетких знаний из экспериментальных данных. На втором этапе проводится параметрическая идентификация исследуемой зависимости путем нахождения таких параметров нечеткой базы знаний, которые минимизируют отклонение модельных и экспериментальных результатов.

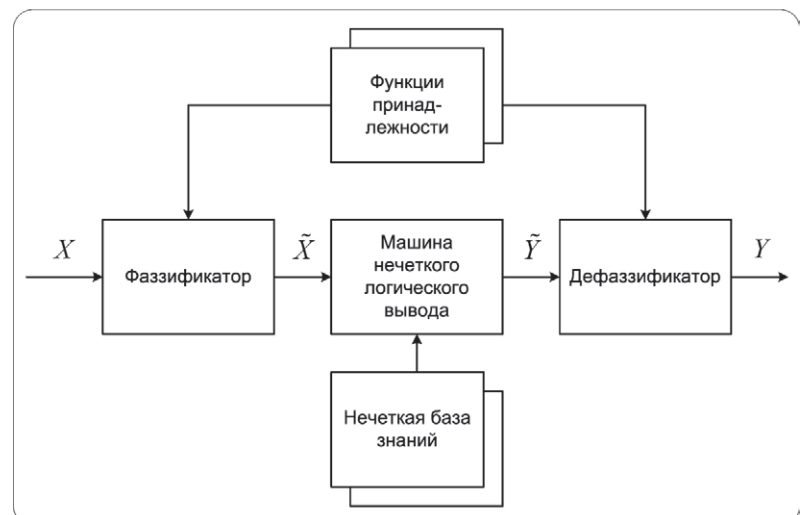
Настоящая работа раскрывает вопросы применения пакетов Fuzzy Logic Toolbox и Optimization Toolbox системы MATLAB для идентификации нелинейных зависимостей с помощью нечеткого логического вывода. Она состоит из двух частей: теоретической, в которой приводятся

В технике, экономике, политике, социологии, медицине, биологии и в других областях часто возникает задача построения математической модели по результатам наблюдений, или задача идентификации (процессов, систем). В тех случаях, когда синтезируемые модели базируются на экспертных лингвистических высказываниях типа «Если инфляционные ожидания — *высокие*, и политическая ситуация — *нестабильная*, и состояние экономики — *кризисное*, то курс национальной валюты — *сильно упадет*», основным инструментом построения моделей являются методы нечеткой логики. Использование лингвистических правил «ЕСЛИ–ТО» позволяет значительно снизить объем экспериментальных данных, необходимых для качественной идентификации. В статье показаны возможности системы MATLAB по автоматизации идентификации нелинейных зависимостей на базе нечетких моделей.

основные сведения по идентификации с помощью нечетких баз знаний, и практической, показывающей возможности системы MATLAB по идентификации нелинейных зависимостей нечеткими моделями различных типов. При написании первой части статьи использовались работы [1–9], второй — руководства пользователя пакетов Fuzzy Logic Toolbox [10] и Optimization Toolbox [11].

2. Модели нечеткого логического вывода

Нечеткий логический вывод — это аппроксимация зависимости «входы–выход» на основе лингвистических высказываний типа «ЕСЛИ–ТО» и операций над нечеткими множествами. Основные положения теории нечетких множеств, используемые в настоящей статье, приведены в Приложении 1. Типовая структура модели на основе нечеткого логического вывода (которую в дальнейшем будем называть нечеткой моделью) показана на рис. 1. Нечеткая модель содержит следующие блоки:



Задача

Идентификация нелинейных систем нечеткими моделями.

Программные средства

Fuzzy Logic Toolbox
Optimization Toolbox

Результаты

Приведены теоретические сведения по идентификации с помощью нечетких баз знаний. Показаны практические возможности MATLAB по идентификации нелинейных зависимостей нечеткими моделями.

▼ **Рис. 1.** Типовая структура модели нечеткого логического вывода.

- *фаззификатор*, преобразующий фиксированный вектор влияющих факторов X в вектор нечетких множеств \tilde{X} , необходимых для выполнения нечеткого логического вывода;
- *нечеткая база знаний*, содержащая информацию о зависимости $Y = f(X)$ в виде лингвистических правил типа «ЕСЛИ–ТО»;
- *машина нечеткого логического вывода*, которая на основе правил базы знаний определяет значение выходной переменной в виде нечеткого множества \tilde{Y} , соответствующего нечетким значениям входных переменных \tilde{X} ;
- *дефаззификатор*, преобразующий выходное нечеткое множество \tilde{Y} в четкое число Y .

В пакете Fuzzy Logic Toolbox реализованы два типа нечетких моделей — типа Мамдани и типа Сугэно. Эти модели отличаются форматом базы знаний и процедурой дефаззификации.

2.1. Нечеткая модель типа Мамдани

В модели типа Мамдани взаимосвязь между входами $X = (x_1, x_2, \dots, x_n)$ и выходом y определяется нечеткой базой знаний следующего формата:

- ЕСЛИ $(x_1 = a_{1,j1})$ И $(x_2 = a_{2,j1})$ И ... И $(x_n = a_{n,j1})$
 ИЛИ $(x_1 = a_{1,j2})$ И $(x_2 = a_{2,j2})$ И ... И $(x_n = a_{n,j2})$
 ...
 ИЛИ $(x_1 = a_{1,jk_j})$ И $(x_2 = a_{2,jk_j})$ И ... И $(x_n = a_{n,jk_j})$
 ТО $y = d_j, \quad i = \overline{1, m},$

где $a_{i,jp}$ — лингвистический терм, которым оценивается переменная x_i в строке с номером jp ($p = \overline{1, k_j}$); k_j — количество строк-конъюнкций, в которых выход y оценивается лингвистическим термом d_j ; m — количество термов, используемых для лингвистической оценки выходной переменной y .

С помощью операций \cup (ИЛИ) и \cap (И) нечеткую базу знаний перепишем в более компактном виде:

$$\bigcup_{p=1}^{k_j} \bigcap_{i=1}^n (x_i = a_{i,jp}) \rightarrow y = d_j, \quad j = \overline{1, m}. \quad (1)$$

Все лингвистические термы в базе знаний (1) представляются как нечеткие множества, заданные соответствующими функциями принадлежности.

Нечеткая база знаний (1) может трактоваться как некоторое разбиение пространства влияющих факторов на подобласти с размытыми границами, в каждой из которых функция отклика принимает значение, заданное соответствующим нечетким множеством. Правило в базе знаний представляет собой «информационный сгусток», отражающий одну из особенностей зависимости «входы–выход». Такие «сгустки насыщенной информации» или «гранулы знаний» могут рассматриваться как аналог вербального кодирования, которое, как установили психологи, происходит в человеческом мозге при обучении. Видимо поэтому формирование нечеткой базы знаний в конкретной предметной области, как правило, не составляет трудностей для эксперта.

Введем следующие обозначения, необходимые для дальнейшего изложения материала:

$\mu_{jp}(x_i)$ — функция принадлежности входа x_i нечеткому терму $a_{i,jp}$, $i = \overline{1, n}, \quad j = \overline{1, m}, \quad p = \overline{1, k_j}$, т. е.

$$a_{i,jp} = \int_{x_i}^{\overline{x_i}} \mu_{jp}(x_i) / x_i, \quad x_i \in [\underline{x_i}, \overline{x_i}];$$

$\mu_{dj}(y)$ — функция принадлежности выхода y нечеткому терму d_j , $j = \overline{1, m}$, т. е.

$$d_j = \int_{\underline{y}}^{\overline{y}} \mu_{dj}(y) / y, \quad y \in [\underline{y}, \overline{y}].$$

Степень принадлежности входного вектора $X^* = (x_1^*, x_2^*, \dots, x_n^*)$ нечетким термам d_j из базы знаний (1) определяется следующей системой нечетких логических уравнений:

$$\mu_{dj}(X^*) = \bigvee_{p=1, k_j} \bigwedge_{i=1, n} [\mu_{jp}(x_i^*)], \quad j = \overline{1, m}, \quad (2)$$

где \vee (\wedge) — операция из s-нормы (t-нормы), т. е. из множества реализаций логических операций ИЛИ (И). Наиболее часто используются следующие реализации: для операции ИЛИ — нахождение максимума, для операции И — нахождение минимума.

Нечеткое множество \tilde{y} , соответствующее входному вектору X^* , определяется следующим образом:

$$\tilde{y} = \text{agg} \left(\int_{\underline{y}}^{\overline{y}} \text{imp}(\mu_{dj}(X^*), \mu_{dj}(y)) / y \right),$$

где *imp* — импликация, обычно реализуемая как операция нахождения минимума; *agg* — агрегирование нечетких множеств, которое наиболее часто реализуется операцией нахождения максимума.

Четкое значение выхода y , соответствующее входному вектору X^* , определяется в результате дефаззификации нечеткого множества \tilde{y} . Наиболее часто применяется дефаззификация по методу центра тяжести:

$$y = \frac{\int_{\underline{y}}^{\overline{y}} y \cdot \mu_{\tilde{y}}(y) dy}{\int_{\underline{y}}^{\overline{y}} \mu_{\tilde{y}}(y) dy}.$$

2.2. Нечеткая модель типа Сугэно

В модели типа Сугэно взаимосвязь между входами $X = (x_1, x_2, \dots, x_n)$ и выходом y задается нечеткой базой знаний вида:

$$\bigcup_{p=1}^{k_j} \bigcap_{i=1}^n (x_i = a_{i,jp}) \rightarrow y = b_{j,0} + b_{j,1} \cdot x_1 + b_{j,2} \cdot x_2 + \dots + b_{j,n} \cdot x_n, \quad j = \overline{1, m}, \quad (3)$$

где $b_{j,i}$ — некоторые числа.

База знаний (3) аналогична (1) за исключением заключений правил d_j , которые задаются не нечеткими термами, а линейной функцией от входов:

$$d_j = b_{j,0} + \sum_{i=1}^n b_{j,i} \cdot x_i.$$

Таким образом, база знаний в модели типа Сугэно является гибридной — ее правила содержат послышки в виде нечетких множеств и заключения в виде четкой линейной функции. База знаний (3) может трактоваться как некоторое разбиение пространства влияющих факторов на нечеткие подобласти, в каждой из которых значение функции отклика рассчитывается как линейная комбинация входов. Правила являются своего рода переключателями с одного линейного закона «входы–выход» на другой, тоже линейный. Границы подобластей размытые, следовательно, одновременно могут выполняться несколько линейных законов, но с различными весами. Результирующее значение выхода y определяется как суперпозиция линейных зависимостей, выполняемых в данной точке X^* n -мерного факторного пространства. Это может быть взвешенное среднее

$$y = \frac{\sum_{j=1}^m \mu_{d_j}(X^*) \cdot d_j}{\sum_{j=1}^m \mu_{d_j}(X^*)}$$

или взвешенная сумма

$$y = \sum_{j=1}^m \mu_{d_j}(X^*) \cdot d_j.$$

Значения $\mu_{d_j}(X^*)$ рассчитываются как и для модели типа Мамдани, т. е. по формуле (2). Обратим внимание, что в модели Сугэно в качестве операций \vee и \wedge обычно используются соответственно вероятностное ИЛИ и умножение. В этом случае нечеткая модель типа Сугэно может рассматриваться как особый класс многослойных нейронных сетей прямого распространения сигнала, структура которой изоморфна базе знаний. Такие сети получили название нейро-нечетких.

Модели типа Мамдани и типа Сугэно будут идентичными, когда заключения правил заданы четкими числами, т. е. в случае, если:

1) термы d_j выходной переменной в модели типа Мамдани задаются синглтонами — нечеткими аналогами четких чисел. В этом случае степени принадлежности для всех элементов универсального множества равны нулю, за исключением одного со степенью принадлежности равной единице;

2) заключения правил в базе знаний модели типа Сугэно заданы функциями, в которых все коэффициенты при входных переменных равны нулю.

3. Основные идеи идентификации на основе нечеткого вывода

Будем предполагать, что идентифицируемая нелинейная зависимость представлена выборкой данных «входы–выход»:

$$(X_r, y_r), \quad r = \overline{1, M}, \quad (4)$$

где $X_r = (x_{r,1}, x_{r,2}, \dots, x_{r,n})$ — вектор входов и y_r — выход в r -паре; M — объем выборки.

Задача идентификации состоит в нахождении нечеткой модели F , обеспечивающей мини-

мальное значение среднеквадратической невязки:

$$R = \frac{1}{M} \sum_{j=1}^M (y_r - F(X_r))^2 \rightarrow \min, \quad (5)$$

где $F(X_r)$ — значение выхода нечеткой модели при значении входов, заданных вектором X_r .

Выход нечеткой модели зависит от ее структуры — базы знаний и параметров: функций принадлежности, реализаций логических операций, метода дефаззификации, а также коэффициентов линейных функций в заключениях правил для модели типа Сугэно. Нахождение структуры и параметров нечеткой модели, обеспечивающих минимальное значение критерия (5) и является задачей идентификации.

Все еще нерешенной научной задачей остается автоматическая экстракция из данных нечетких правил, близких по своим свойствам к экспертным, т. е. полезных, достоверных, интересных, новых и понятных не только математикам. Это относится прежде всего к базе знаний модели типа Мамдани, так как прозрачность правил модели типа Сугэно, т. е. возможность их содержательной интерпретации пользователем, является довольно низкой. Будем предполагать, что кроме числовых пар «входы–выход» существует еще и лингвистическая информация об идентифицируемой зависимости в форме экспертных высказываний. Обозначим через I вектор параметров функций принадлежности термов входных переменных, через O — вектор параметров функций принадлежности термов выходной переменной модели типа Мамдани и через B — вектор коэффициентов линейных функций в заключениях правил в модели типа Сугэно. Тогда параметрическая идентификация сводится к следующей задаче математического программирования:

• для модели типа Мамдани — *найти такой вектор (I, O) , чтобы*

$$\frac{1}{M} \sum_{j=1}^M (y_r - F(I, O, X_r))^2 \rightarrow \min;$$

• для модели типа Сугэно — *найти такой вектор (I, B) , чтобы*

$$\frac{1}{M} \sum_{j=1}^M (y_r - F(I, B, X_r))^2 \rightarrow \min.$$

В приведенных выше задачах оптимизации на управляемые переменные (I, O) обычно накладываются ограничения, обеспечивающие линейную упорядоченность элементов терм-множеств. Такие ограничения не позволяют алгоритмам оптимизации сделать, например, терм «Низкий» выше терма «Высокий».

4. Нечеткая идентификация в системе MATLAB

Пакет Fuzzy Logic Toolbox, входящий в состав системы MATLAB, содержит набор GUI-модулей, обеспечивающих проведение этапа структурной идентификации в диалоговом режиме. На этом этапе пользователь выбирает количество входов и выходов модели, задает количество термов и типы функций принадлежности, формирует базу знаний. Вызов основного GUI-модуля осуществляется по команде **fuzzy**. Более детальную инфор-

мацию о построении нечетких моделей в системе MATLAB можно найти на сайте <http://www.matlab.ru> в разделе Fuzzy Logic Toolbox.

Автоматизацию этапа параметрической идентификации пакет Fuzzy Logic Toolbox обеспечивает только для моделей типа Сугэно. Настройка нечеткой модели типа Сугэно может проводиться как в командном режиме с помощью функции **anfis**, так и в диалоговом режиме с использованием GUI-модуля **anfisedit**. Настройка выполняется с использованием технологии ANFIS (*Adaptive Network-based Fuzzy Inference System*) — одного из первых методов построения нейро-нечетких систем для аппроксимации функций, предложенного Янгом в 1991 году. Настройка представляет собой итерационную процедуру нахождения параметров функций принадлежности, которые минимизируют расхождения между действительным и желаемым поведением модели. Пользователь может выбрать метод настройки — метод обратного распространения ошибки или его комбинацию с методом наименьших квадратов. Пакет Fuzzy Logic Toolbox позволяет автоматически синтезировать из данных нечеткую базу знаний для модели типа Сугэно. Для этого используются два алгоритма — решеточное разбиение (grid partition) и субтрактивная кластеризация (subtractive clustering). На выходе первого алгоритма получается база знаний, содержащая все возможные правила. В результате субтрактивной кластеризации генерируются правила, соответствующие областям наибольшей концентрации данных.

Настройка моделей типа Мамдани в пакете Fuzzy Logic Toolbox не предусмотрена. Мы предлагаем для решения задачи параметрической идентификации нечеткой базы знаний использовать функции нелинейной оптимизации, реализованные в пакете Optimization Toolbox. Применение пакетов Fuzzy Logic Toolbox и Optimization Toolbox для идентификации нелинейной зависимости с помощью нечеткого логического вывода ниже иллюстрируется на примере.

5. Пример нечеткой идентификации

В этом разделе описывается процесс построения нечетких моделей, идентифицирующих нелинейную зависимость

$$y = x_1^2 \cdot \sin(x_2 - 1) \tag{6}$$

в диапазоне $x_1 \in [-7, 3]$, $x_2 \in [-4.4, 1.7]$. Исходными данными для идентификации являются график функции $y = x_1^2 \cdot \sin(x_2 - 1)$ (рис. 2, а) и обучающая выборка в форме (4).

5.1. Идентификация на основе модели Мамдани

Входы и выход модели будем рассматривать как лингвистические переменные, значения которых определяются из следующих терм-множеств: {«Низкий», «Средний», «Высокий»} для x_1 и x_2 и {«Низкий», «Ниже Среднего», «Средний», «Выше Среднего», «Высокий»} для y . Формализацию термов осуществим с помощью симметричной гауссовской функции принадлежности (**gaussmf**):

$$\mu(x) = e^{-\frac{(x-h)^2}{2c^2}}$$

где x — элемент универсального множества; h — параметр функции принадлежности (координата максимума); c — параметр функции принадлежности (коэффициент концентрации).

Выбор такого типа функции принадлежности обусловлен ее достаточной гибкостью и простотой — она задается лишь двумя параметрами. Это позволяет сократить размерность задачи оптимизации, возникающей на этапе параметрической идентификации.

В результате визуального наблюдения графика идентифицируемой зависимости сформулируем семь правил базы знаний (табл. 1). На этом этапе структурной идентификации заканчивается.

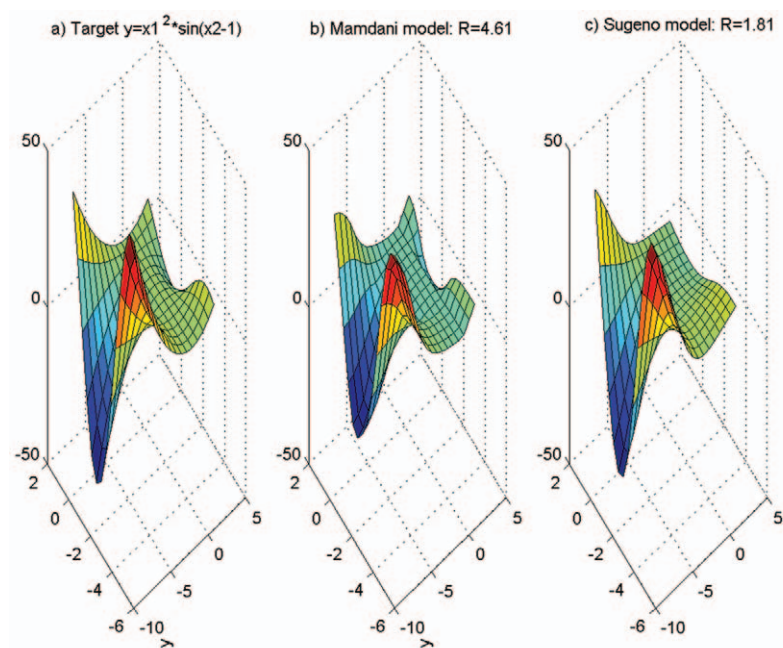
Таблица 1

Нечеткая база знаний модели типа Мамдани

x_1	x_2	y
Низкий	Низкий	Высокий
Низкий	Средний	Низкий
Низкий	Высокий	Высокий
Средний	—	Средний
Высокий	Низкий	Выше Среднего
Высокий	Средний	Ниже Среднего
Высокий	Высокий	Выше Среднего

Для проведения параметрической идентификации будем использовать программу, приведенную в Приложении 2. Она состоит из m -сценария, вызывающего функцию нелинейной оптимизации **constr**, и m -функции **goal_fun**, вычисляющей невязку при заданных значениях управляемых переменных. Количество управляемых переменных равно шестнадцати: одиннадцать коэффициентов концентраций функций принадлежности термов входных и выходной переменных; две координаты максимумов центров функций принадлежности термов «Средний» входных переменных; три координаты максимумов центров функций принадлежности некрайних

▼ Рис. 2. Поверхности «входы-выход»: а) исходной зависимости; б) нечеткой модели типа Мамдани; в) нечеткой модели типа Сугэно.



термов выходной переменной (т. е. термов «Низе Среднего», «Средний» и «Выше Среднего»). Координаты максимумов функций принадлежности крайних термов («Низкий» и «Высокий») не настраиваются, так как нет никаких логических оснований предполагать, что они будут отличаться от границ диапазонов изменения переменных.

В результате обучения найдены оптимальные функции принадлежности нечетких термов (рис. 3). Поверхность «входы–выход», соответствующая настроенной нечеткой модели типа Мамдани, показана на рис. 2, б. Нечеткая модель оптимизирована по обучающей выборке из 20 пар «входы–выход». Средняя квадратическая ошибка идентификации этой модели на контрольной выборке из 1000 точек составляет 4.61.

5.2. Идентификация на основе модели Сугэно

По рис. 2, а визуально трудно оценить коэффициенты линейных функций, связывающих входы и выход зависимости (б) для конкретной подобласти. Поэтому базу знаний сгенерируем непосредственно из обучающей выборки. Обучающая выборка представляет собой набор случайно сгенерированных точек факторного пространства, поэтому нет оснований для применения кластеризации при синтезе базы знаний. Будем использовать алгоритм решеточного разбиения с параметрами, установленными по умолчанию, — по два термина для оценки каждой входной переменной с обобщенной колоколообразной функцией принадлежности

$$\mu(x) = \frac{1}{1 + \left| \frac{x-h}{c} \right|^{2v}}$$

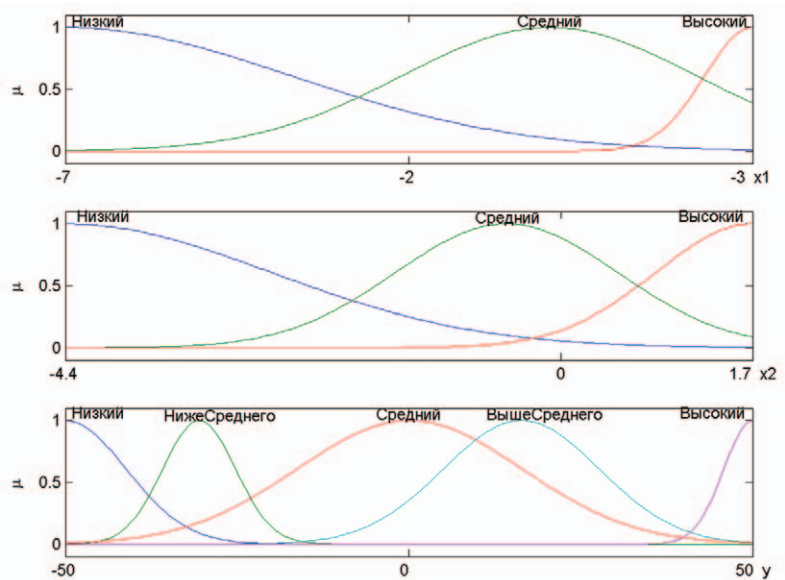
где c — коэффициент концентрации функции принадлежности; v — коэффициент крутизны функции принадлежности; h — координата максимума функции принадлежности.

Пакет Fuzzy Logic Toolbox позволяет синтезировать модель типа Сугэно и обучать ее путем вызова одной функции **anfis** в следующем формате:

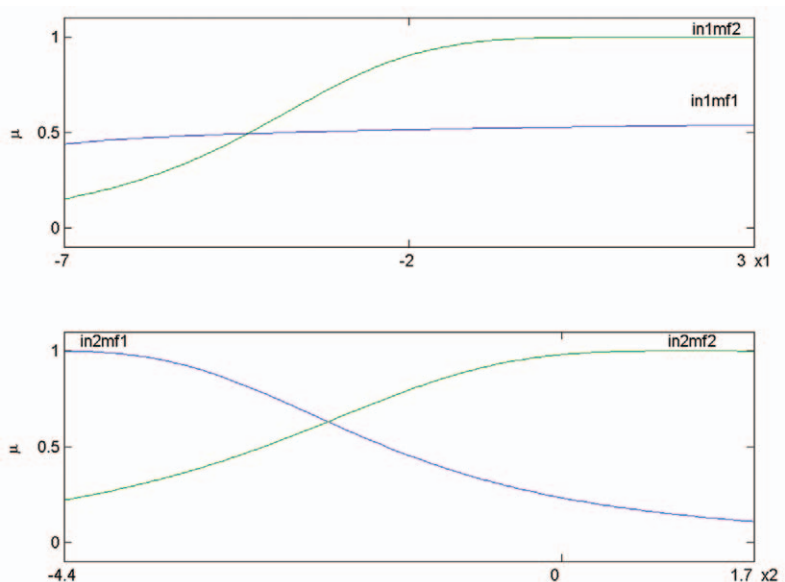
fuzzy_model = anfis (training_data, [], ... number_of_iterations).

При обучении настраивалось 24 параметра нечеткой модели — по 3 коэффициента в линейной функции для каждого из 4-х правил базы знаний и по 3 параметра для каждого из 4-х термов входных переменных. Такое количество настраиваемых параметров является минимальным для нечеткой модели типа Сугэно с двумя входами и одним выходом.

На рис. 2, б приведена поверхность «входы–выход», соответствующая настроенной нечеткой модели типа Сугэно. Нечеткая модель оптимизирована по обучающей выборке из 100 пар «входы–выход». Средняя квадратическая ошибка идентификации этой модели на контрольной выборке из 1000 точек составляет 1.81. Графики функций принадлежности входных переменных показаны на рис. 4; нечеткая база знаний для этой модели приведена в табл. 2.



▲ Рис. 3. Оптимальные функции принадлежности нечеткой модели типа Мамдани.



▲ Рис. 4. Оптимальные функции принадлежности нечеткой модели типа Сугэно.

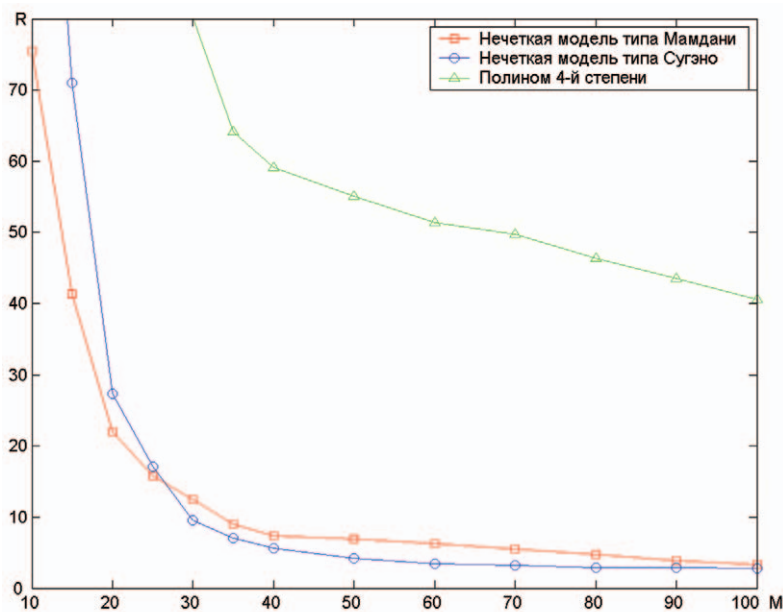
Таблица 2

Нечеткая база знаний модели типа Сугэно

x_1	x_2	y
in1mf1	in2mf1	$3.9 + 94.8x_1 + 696.2x_2$
in1mf1	in2mf2	$-1.8 + 118.6x_1 - 249x_2$
in1mf2	in2mf1	$0.7 - 46.2x_1 - 341.1x_2$
in1mf2	in2mf2	$-0.6 - 58.5x_1 + 121.7x_2$

5.3. Сравнение результатов идентификации

Для сравнения результатов были построены кривые обучения нечетких моделей в виде зависимости ошибки идентификации на контрольной выборке от размера обучающей выборки (рис. 5). Каждая точка на графике рассчитывалась как среднее значение пяти экспериментов с различными случайно сгенерированными обучающими выборками. Среднее количество итераций обучения модели типа Мамдани составляло 1000, а мо-



▲ Рис. 5. Кривые обучения.

дели типа Сугэно — 100 итераций. При небольших объемах обучающей выборки количество итераций уменьшалось с целью избежания эффекта переобучения модели.

При малых обучающих выборках качество идентификации существенно выше для модели типа Мамдани. Это объясняется тем, что исходная, основанная на экспертных высказываниях, нечеткая модель уже отражает основные особенности идентифицируемой зависимости. С увеличением объема обучающей выборки лучшее качество идентификации обеспечивает модель типа Сугэно. Заметим, что хорошие результаты идентификации

После обучения модель типа Мамдани остается прозрачной: ее параметры — функции принадлежности — легко интерпретируются лингвистическими терминами. Для моделей типа Сугэно типовое явление — сложность содержательной интерпретации ее параметров.

получаются, когда объем обучающей выборки в два и более раз превышает количество настраиваемых параметров модели. При превышении более чем в три раза объема обучающей выборки над количеством настраиваемых

параметров наступает эффект насыщения — точность идентификации практически не улучшается с увеличением числа наблюдений. При больших выборках точность идентификации модели типа Сугэно выше, чем для модели типа Мамдани. Однако после обучения модель типа Мамдани остается прозрачной: ее параметры — функции принадлежности — легко интерпретируются лингвистическими терминами (см. рис. 3). Для моделей типа Сугэно типовое явление — сложность содержательной интерпретации ее параметров. Например, трудно объяснить специалистам из прикладных областей — врачам, биологам, социологам — базу знаний, аналогичную приведенной в табл. 2, не говоря уже об интерпретации функции принадлежности термина in1mf1 (рис. 4).

На рис. 5 в качестве примера приведена также кривая обучения и для аппроксимирующего полинома 4-го порядка. Видно, что ошибка идентификации для нечетких моделей значительно ниже, чем для полиномиальных.



Автор:

Штовба Сергей Дмитриевич, доцент, кандидат технических наук, доцент кафедры компьютерных систем управления; Винницкий государственный технический университет, г. Винница, Украина

6. Заключение

Идентификация с помощью нечеткого логического вывода является эффективным методом построения моделей нелинейных зависимостей, наблюдаемых в технике, экономике, социологии, биологии, спорте и других областях. В статье показаны возможности системы MATLAB по автоматизации такой идентификации на базе нечетких моделей двух типов — Мамдани и Сугэно. В результате компьютерных экспериментов установлено, что использование лингвистической информации в виде экспертных правил «ЕСЛИ–ТО» позволяет значительно снизить необходимый объем обучающей выборки для нечеткой идентификации. При больших объемах выборки экспериментальных данных идентификация с помощью модели типа Сугэно обеспечивает, как правило, большую точность. Однако при этом возникают трудности с содержательной интерпретацией параметров нечеткой модели и с объяснением логического вывода. С моделью типа Мамдани таких трудностей не возникает, ее параметры и после обучения легко интерпретируются содержательно. Процедура нечеткого логического вывода в модели типа Мамдани интуитивно понятна и заказчикам нечетких моделей: технологах, экономистам, врачам, биологам. Поэтому для задач, где более важна точность идентификации, целесообразным будет использование нечетких моделей типа Сугэно, а для задач, где более важным является объяснение, обоснование принятого решения, будут иметь преимущество нечеткие модели типа Мамдани.

Литература

1. Заде Л. Понятие лингвистической переменной и ее применение к принятию приближенных решений. — М: Мир, 1976.— 167 с.
2. Ротштейн А. П. Интеллектуальные технологии идентификации: нечеткая логика, генетические алгоритмы, нейронные сети. — Винница: УНИВЕРСУМ—Винница, 1999.— 320 с.
3. Митюшкин Ю. И., Мокин Б. И., Ротштейн А. П. Soft-Computing: идентификация закономерностей нечеткими базами знаний. — Винница: УНИВЕРСУМ—Винница, 2002.— 145 с.
4. Rotshtein A. Desig and Tuning of Fuzzy Rule-Based System for Medical Diagnosis // In Fuzzy and Neuro-Fuzzy Systems in Medicine / Ed. Teodorescu N. H. — CRC-Press, 1998.— P. 243–289.
5. Ротштейн А. П., Кательников Д. И. Идентификация нелинейных зависимостей нечеткими базами знаний // Кибернетика и системный анализ.— 1998.— №5.— С. 53–61.
6. Ротштейн А. П., Митюшкин Ю. И. Извлечение нечетких правил из экспериментальных данных с помощью генетических алгоритмов // Кибернетика и системный анализ.— 2001.— №3.— С. 45–53.
7. Ротштейн А.П., Штовба С.Д. Влияние методов дефазификации на скорость настройки нечеткой модели // Кибернетика и системный анализ.—2002.— №5.— С.169–176.
8. Zimmermann H.-J. Fuzzy Set Theory and its Applications. 3rd ed. — Dordrecht: Kluwer Academic Publishers, 1996.— 315 p.
9. Yager R., Filev D. Essential of Fuzzy Modeling and Control.— John Willey & Sons, 1994.— 388 p.
10. Fuzzy Logic Toolbox. User's Guide, Version 2.— The MathWorks, Inc., 1999.
11. Optimization Toolbox. User's Guide, Version 2.— The MathWorks, Inc., 1999.

Приложение 1. Основные определения теории нечетких множеств

Нечетким множеством \tilde{A} на универсальном множестве U называется совокупность пар $(\mu_A(u), u)$, где $\mu_A(u)$ — степень принадлежности элемента $u \in U$ к нечеткому множеству \tilde{A} . Степень принадлежности — число из диапазона $[0, 1]$. Чем выше степень принадлежности, тем в большей мере элемент универсального множества соответствует свойствам нечеткого множества.

Функцией принадлежности называется функция, которая позволяет вычислить степень принадлежности произвольного элемента универсального множества к нечеткому множеству.

Если универсальное множество состоит из конечного количества элементов $U = \{u_1, u_2, \dots, u_k\}$, то нечеткое множество \tilde{A} записывается в виде

$$\tilde{A} = \sum_{i=1}^k \mu_A(u_i) / u_i.$$

В случае непрерывного множества U используют обозначение

$$\tilde{A} = \int_U \mu_A(u) / u.$$

Здесь знаки \sum и \int означают совокупность пар $\mu_A(u)$ и u .

Лингвистической переменной называется переменная, значениями которой могут быть слова или словосочетания некоторого естественного или искусственного языка.

Терм-множеством называется множество всех возможных значений лингвистической переменной.

Термом называется любой элемент терм-множества. В теории нечетких множеств терм формализуется нечетким множеством с помощью функции принадлежности.

Дефаззификацией называется процедура преобразования нечеткого множества в четкое число. В теории нечетких множеств процедура дефаззификации аналогична нахождению характеристик положения (математического ожидания, моды, медианы) случайных величин в теории вероятности. Простейшим способом выполнения процедуры дефаззификации является выбор четкого числа, соответствующего максимуму функции принадлежности. Однако пригодность этого способа ограничивается лишь одноэкстремальными функциями принадлежности. Для многоэкстремальных функций принадлежности наиболее часто используется дефаззификация путем нахождения центра тяжести плоской фигуры, ограниченной осями координат и функцией принадлежности.

Приложение 2. Программа обучения нечеткой модели типа Мамдани

```
global FIS TR_INP TR_OUT NUM_TERMS
FIS=readfis('mamdani0.fis');
%загрузка исходной нечеткой модели
data=load('exp_data.dat'); %загрузка обучающей выборки
TR_INP=data(1:tr_vol, 1:2);
%значения входных переменных обучающей выборки
TR_OUT=data(1:tr_vol, 3);
%значения входных переменных обучающей выборки
NUM_TERMS=[3 3 5]; %количество термов

%=====УПРАВЛЯЕМЫЕ ПЕРЕМЕННЫЕ=====
%—Коэффициенты концентраций функций принадлежности—
%Первая входная переменная x1:
vlb_in1_con(1:NUM_TERMS(1))=0.2; %ограничения снизу
vub_in1_con(1:NUM_TERMS(1))=4; %ограничения сверху
x0_in1_con(1:NUM_TERMS(1))=3; %начальные значения
%Вторая входная переменная x2:
vlb_in2_con(1:NUM_TERMS(2))=0.15; %ограничения снизу
vub_in2_con(1:NUM_TERMS(2))=2.5; %ограничения сверху
x0_in2_con(1:NUM_TERMS(2))=1.35; %начальные значения
%Выходная переменная y:
vlb_out_con(1:NUM_TERMS(3))=0.15; %ограничения снизу
vub_out_con(1:NUM_TERMS(3))=4.0; %ограничения сверху
x0_out_con(1:NUM_TERMS(3))=1.0; %начальные значения
%Координаты максимумов функций принадлежности
%некрайних термов
vlb_center=[-6 -4 -4.5 -4.5 -4.5]; %ограничения снизу
vub_center=[2 1.3 4.5 4.5 4.5]; %ограничения сверху
x0_center=[-2 -1.35 -2.5 0 2.5]; %начальные значения
%Объединение управляемых переменных в один вектор:
vlb=[vlb_in1_con vlb_in2_con vlb_out_con vlb_center];
%ограничения снизу
vub=[vub_in1_con vub_in2_con vub_out_con vub_center];
%ограничения сверху
x0=[x0_in1_con x0_in2_con x0_out_con x0_center];
%начальная точка

%=====ЦЕЛЕВАЯ ФУНКЦИЯ=====
funf='f=goal_fun(x)'; %задание файла целевой функции
fung='g=[x(14)-x(15)+.5; x(15)-x(16)+.5]';
%условия линейной упорядоченности
fun=[funf fung]; %целевая функция + ограничения

%=====ПАРАМЕТРЫ АЛГОРИТМА ОПТИМИЗАЦИИ=====
options(1)=1; %вывод промежуточных результатов
options(14)=1000; %максимальное количество итераций

%=====ОПТИМИЗАЦИЯ=====
[xopt options]=constr(fun, x0, options, vlb, vub);
```

```
%=====ЗАПИСЬ НЕЧЕТКОЙ МОДЕЛИ НА ДИСК=====
writefis(FIS)

function delta=goal_fun(x)
global TR_INP TR_OUT FIS NUM_TERMS
out=length(NUM_TERMS); %количество переменных модели
ii=1; %счетчик управляемых переменных

%==УСТАНОВКА НОВЫХ ЗНАЧЕНИЙ ПАРАМЕТРОВ ФУНКЦИЙ ПРИНАДЛЕЖНОСТИ

%—Коэффициенты концентрации—
%Для входных переменных:
for j=1:out-1
for i=1:NUM_TERMS(j)
FIS.input(j).mf(i).params(1)=x(ii);
ii=ii+1;
end
end
%Для выходной переменной:
for i=1:NUM_TERMS(out)
FIS.output(1).mf(i).params(1)=10*x(ii);
ii=ii+1;
end

%—Координаты максимумов—
%Для входных переменных:
for j=1:out-1
for i=2:NUM_TERMS(j)-1
FIS.input(j).mf(i).params(2)=x(ii);
ii=ii+1;
end
end
%Для выходной переменной:
for i=2:NUM_TERMS(out)-1
FIS.output(1).mf(i).params(2)=10*x(ii);
ii=ii+1;
end

%=====НЕЧЕТКИЙ ЛОГИЧЕСКИЙ ВЫВОД=====
fis_out=evalfis(TR_INP, FIS);

%=====РАСЧЕТ НЕВЯЗКИ=====
delta=sum((fis_out-TR_OUT).^2)/length(TR_OUT);
```