Лабораторная работа №1

Нечёткая логика и нечёткие множества. Построение нечёткой аппроксимирующей системы в пакете Fuzzy Logic Toolbox

Цель работы: изучить основные определения теории нечётких множеств, способы задания функций принадлежности и научиться их строить средствами MATLAB; изучить пакет Fuzzy Logic Toolbox, научиться строить нечёткие аппроксимирующие системы.

Продолжительность работы: 4 часа.

Теоретические сведения

Нечёткая логика

В классической логике Аристотеля возможно оперирование только чётко определёнными понятиями. Все высказывания в классических логических системах могут иметь два состояния: «истина» со значением истинности 1 или «ложь» со значением истинности 0. Для этих двух состояний существуют различные логические операции, из которых содержательно интерпретируются лишь пять: И (∧), ИЛИ (∨), исключающее ИЛИ (сложение по модулю 2), импликация (→) и эквивалентность (↔).

Любая логическая функция может быть представлена в виде дизъюнктивной или конъюнктивной нормальной формы, т.е. в виде комбинации функций: конъюнкция (И, a&b), дизъюнкция (ИЛИ, a||b) и отрицания (НЕ, ā).

В нечёткой логике количество возможных значений истинности может быть бесконечным, но все они лежат на отрезке [0, 1]. В общем виде табличное представление логических операций в нечёткой логике, в отличие от классической, невозможно.

Нечёткие множества

Рассмотрим некоторое множество A. Подмножество B множества A описываем некоторой характеристической функцией:

$$\mu_B(x) = \begin{cases} 1, x \in B; \\ 0, x \notin B. \end{cases} \quad (1)$$

Под нечётким множеством понимается подмножество B множества A, характеризующееся своей функцией принадлежности $\mu_B: A \to [0, 1]$. Значение функции принадлежности в точке x показывает степень принадлежности этой точки нечёткому множеству. Нечёткое множество описывает неопределённость, соответствующую точке x: она одновременно и входит, и не входит в нечёткое множество B.

Область значений функции принадлежности должна принадлежать диапазону [0, 1].

Рассмотрим определение теоретико-множественных операций над нечёткими множествами. Пусть C и D – два нечётких подмножества A с функциями принадлежности и соответственно. Пересечение $C \cap D$, произведение CD, объединение $C \cup D$, отрицание $\bar{C}$, сумма $C+D$ будут иметь следующие функции принадлежности:

$$\mu_{C\cap D}(x) = \min(\mu_C(x), \mu_D(x))\,;$$
$$\mu_{CD}(x) = \mu_C(x)\mu_D(x);$$
$$\mu_{\bar{C}}(x) = 1 - \mu_C(x); \tag{2}$$
$$\mu_{C\cup D}(x) = \max(\mu_C(x), \mu_D(x))\,;$$
$$\mu_{C+D}(x) = \mu_C(x) + \mu_D(x) - \mu_C(x)\mu_D(x), x \in A.$$

Нечёткая и лингвистическая переменные

Нечёткая переменная определяется как кортеж:

$$< a, X, A >, \tag{3}$$

где a – название нечёткой переменной; X – область её определения, A – нечёткое множество на X, описывающее возможные значения, которые может принимать переменная a.

Инструментарий нечёткой логики в составе пакета MATLAB содержит 11 встроенных типов функций принадлежности (ФП). К наиболее распространённым ФП можно отнести треугольную и трапециевидную.

Лингвистической переменной называется переменная, значениями которой могут быть слова или словосочетания естественного или искусственного языка. Именно из лингвистических переменных состоят нечёткие множества. Формально лингвистическая переменная определяется пятёркой:

$$< x, T, U, G, M >, \tag{4}$$

где x – название лингвистической переменной; T – базовое терм-множество (множество значений этой переменной, каждое из которых представляет собой название отдельной нечёткой переменной); U – область определения нечётких переменных, которые входят в определение лингвистической переменной x; G – некоторая синтаксическая процедура, которая описывает процесс образования из множества T новых осмысленных значений для данной лингвистической переменной (синтаксические правила), M – семантическая процедура, которая позволяет поставить в соответствие каждому новому значению данной лингвистической переменной, получаемому с помощью процедуры G, некоторое осмысленное содержание посредством формирования соответствующего нечёткого множества.

Основоположник теории нечёткой логики математик Лофти Заде предложил такие функции принадлежности термов «истинно» и «ложно»:

$$\mu_{\text{истинно}}(u) = \begin{cases} 0, 0 \le u \le a; \\ 2\left(\dfrac{u-a}{1-a}\right)^2, a < u \le \dfrac{a+1}{2}; \\ 1 - 2\left(\dfrac{u-a}{1-a}\right)^2, \dfrac{a+1}{2} < u \le 1; \end{cases} \tag{5}$$
$$\mu_{\text{ложно}}(u) = \mu_{\text{истинно}}(1-u), u \in [0, 1],$$

где $a \in [0, 1]$ - параметр, определяющий носители нечётких множеств «истинно» и «ложно».

Для нечёткого множества «истинно» носителем будет интервал (a, 1], а для «ложно» - [0, a].

На рис.1 построены графики функций принадлежности термов «истинно» и «ложно» при a=0,4. Как видно, графики представляют собой зеркальные отображения.

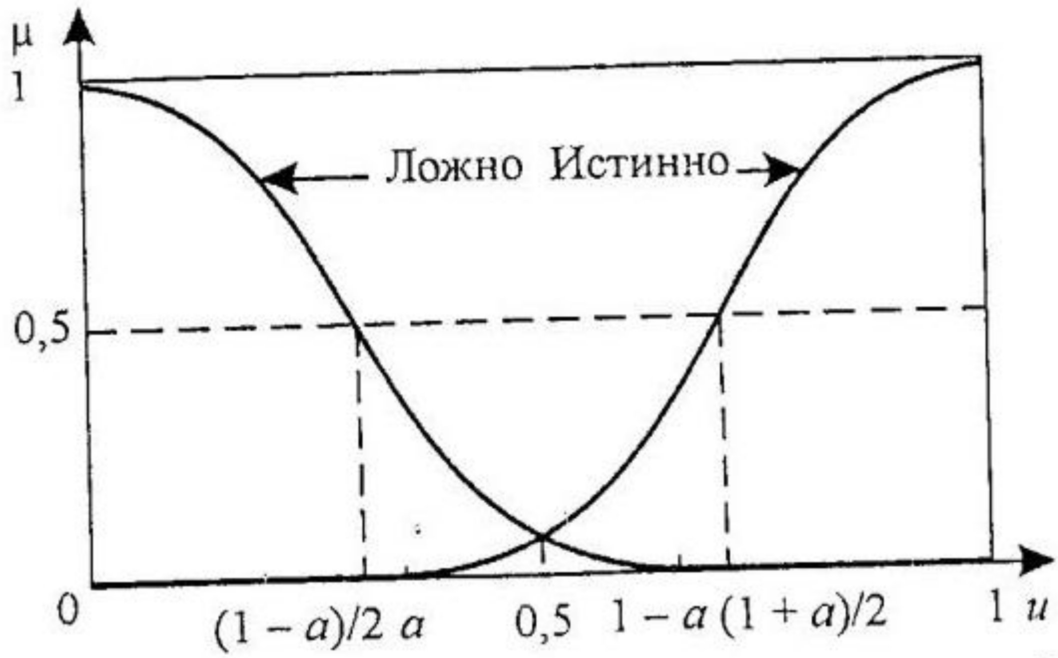


Рис. 1. Лингвистическая переменная «истинность» по Заде

Нечёткие отношения

Нечётким отношением R на множестве X называется нечёткое подмножество декартова произведения $X \times X$, которое характеризуется такой функцией принадлежности $\mu_R(x,y)$, что $X \times X \xrightarrow{\mu_R} [0,1]$. Причём $\mu_R(x,y)$ принимается как субъективная мера выполнения отношения *xRy*.

Пример. Задать отношение «схожий менталитет» для следующих национальностей: украинцы (У), чехи (Ч), австрийцы (А), немцы (Н).

Использование обычного, не нечёткого отношения позволяет выделить только одну пару наций со схожими менталитетами: немцев и австрийцев. Этим отношением не отражается тот факт, что по менталитету чехи более близки к немцам, чем украинцы. Нечёткое отношение позволяет легко представить такую информацию.

$$\tilde{R} = \begin{array}{c} \begin{array}{cccc} \text{У} & \text{Ч} & \text{А} & \text{Н} \end{array} \\ \begin{bmatrix} 1 & 0{,}5 & 0{,}2 & 0{,}1 \\ 0{,}5 & 1 & 0{,}6 & 0{,}3 \\ 0{,}2 & 0{,}6 & 1 & 0{,}8 \\ 0{,}1 & 0{,}3 & 0{,}8 & 1 \end{bmatrix} \end{array} \begin{array}{c} \text{У} \\ \text{Ч} \\ \text{А} \\ \text{Н} \end{array} \qquad (6)$$

Пусть на множестве $X \times X$ заданы да нечётких отношения A и B с функциями принадлежности $\mu_A(x,y), \mu_B(x,y)$. Тогда множества $C = A \cup B$ и $D = A \cap B$ представляют собой объединение и пересечение нечётких отношений A и B на множестве X соответственно, если их функции принадлежности определяются выражениями

$$\begin{gathered} \mu_C(x,y) = \max\{\mu_A(x,y), \mu_B(x,y)\}, \\ \mu_D(x,y) = \min\{\mu_A(x,y), \mu_B(x,y)\}. \end{gathered} \qquad (7)$$

Нечёткое отношение B включает в себя нечёткое отношение A $(A \subset B)$, если $\mu_A(x,y) \le \mu_B(x,y), \forall x,y \in X$.

Если R – нечёткое отношение с функцией принадлежности $\mu_R(x,y)$, то его дополнение $\bar{R}$ имеет функцию принадлежности $\mu_{\bar{R}}(x,y) = 1 - \mu_R(x,y), \forall x,y \in X$. Обратное к R отношение на X определяется как $xR^{-1}y \times\leftrightarrow yRx$, при этом функции принадлежности равны: $\mu_{R^{-1}}(x,y) = \mu_R(x,y)$.

Свойства нечётких отношений:

1. Рефлексивность. Нечёткое отношение R рефлексивно на X, если $\mu_R(x,x) = 1, \forall x \in X$.
2. Антирефлексиность. Нечёткое отношение R антирефлексивно на X, если $\mu_R(x,x) = 0, \forall x \in X$.
3. Симметричность. Нечёткое отношение R симметрично на X, если $\mu_R(x,y) = \mu_R(y,x), \forall x,y \in X$.
4. Антисимметричность. Нечёткое отношение R антисимметрично на X, если $\mu_R(x,y) > 0 \to \mu_R(y,x) = 0, \forall x,y \in X$.

Функции принадлежности

Инструментарий нечёткой логики в составе пакета MATLAB содержит 11 встроенных типов функций принадлежности (ФП). К наиболее распространённым ФП можно отнести треугольную и трапециевидную.

1. Построим треугольную и трапециевидную функции принадлежности.
В аналитическом виде треугольная ФП может быть задана следующим образом:

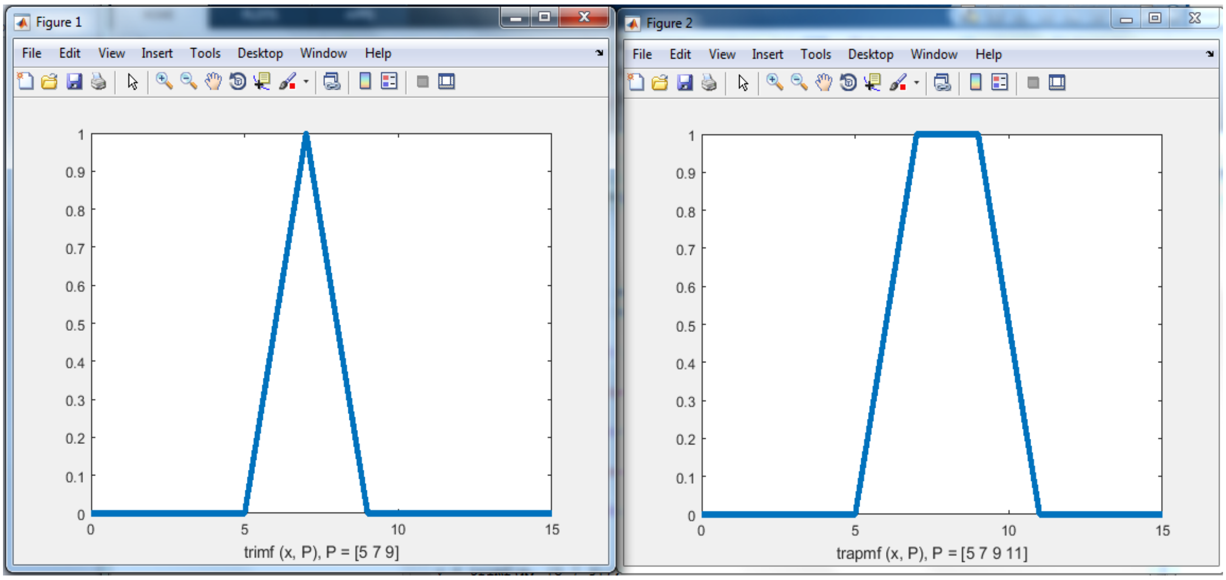
$$f(x,a,b,c) = \begin{cases} 0, x < a \text{ или } x > c; \\ \dfrac{x-a}{b-a}, a \le x < b; \\ \dfrac{c-x}{c-b}, b \le x \le c. \end{cases} \qquad (8)$$

Аналитическая запись трапециевидной функции имеет вид:

$$f(x,a,b,c,d) = \begin{cases} 0, x < a \text{ или } x > d; \\ \dfrac{x-a}{b-a}, a \le x < b; \\ 1, b < x \le c; \\ \dfrac{d-x}{d-c}, c < x \le d. \end{cases} \qquad (9)$$

Теперь построим эти функции в MATLAB.

```
x = 0:0.1:15;
y = trimf(x, [5 7 9]);
plot (x, y, 'LineWidth', 4);
xlabel ('trimf (x, P), P = [5 7 9]');
figure
y = trapmf(x, [5 7 9 11]);
plot (x, y, 'LineWidth', 4);
xlabel ('trapmf (x, P), P = [5 7 9 11]');
```



(а) (б)

Рис. 2. Треугольная (а) и трапециевидная (б) функции принадлежности

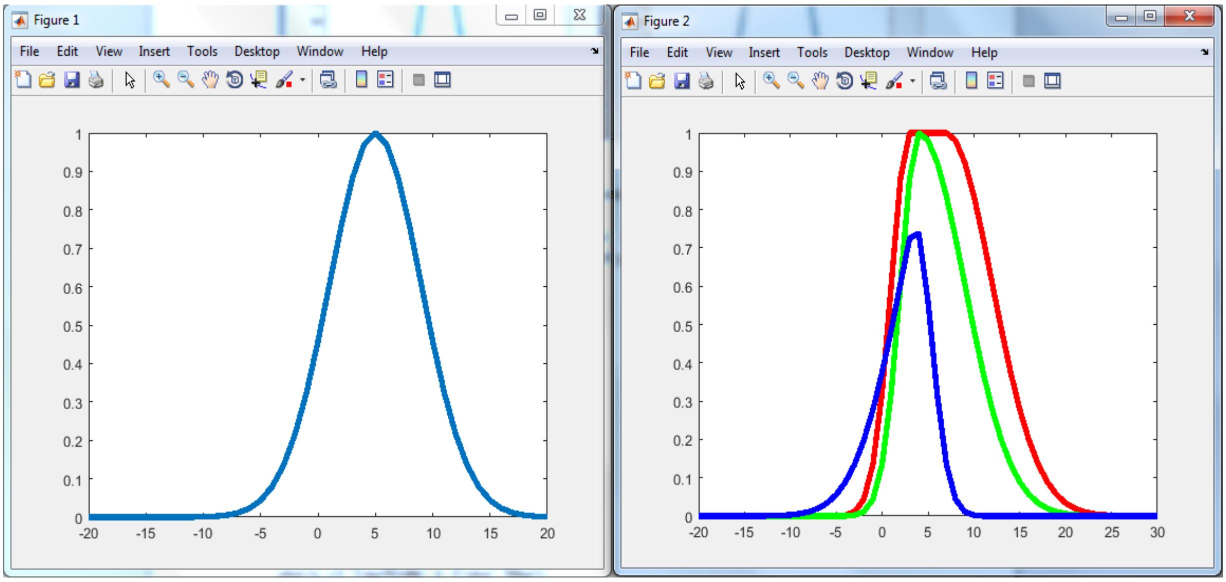
2. Построим простую и двухстороннюю функцию принадлежности Гаусса, образованную с помощью различных функций распределения.

Симметричная (простая) функция Гаусса зависит от двух параметров:

$$f(x, \delta, c) = \left\{ e^{\frac{-(x-c)^3}{2\delta^2}} \right. . \qquad (10)$$

Двухсторонняя функция распределения Гаусса представляет собой комбинацию двух простых.

```
x = -20:1:20;
y = gaussmf (x, [4 5]);
plot (x, y, 'LineWidth', 4);
clear x;
x = [-20:30]';
y1 = gauss2mf (x, [2 3 5 7]);
y2 = gauss2mf (x, [2 4 5 4]);
y3 = gauss2mf (x, [5 7 2 3]);
figure
plot (x, y1, 'LineWidth', 4, 'Color', 'Red');
hold on;
plot (x, y2, 'LineWidth', 4, 'Color', 'Green');
hold on;
plot (x, y3, 'LineWidth', 4, 'Color', 'Blue');
```



(а) (б)

Рис. 3. Простая (а) и двухсторонняя (б) функции принадлежности Гаусса

3. Построим функцию принадлежности "обобщенный колокол".
Данная ФП зависит от трёх параметров и имеет следующую аналитическую запись:

$$f(x, a, b, c) = \left\{ \frac{1}{1 + \left|\frac{x-c}{a}\right|^{2b}} \right. . \tag{11}$$

```
x = -20:1:20;
y = gbellmf (x, [2 3 4]);
plot (x, y, 'LineWidth', 4);
xlabel ('gbellmf (x, P), P = [2 3 4]');
```

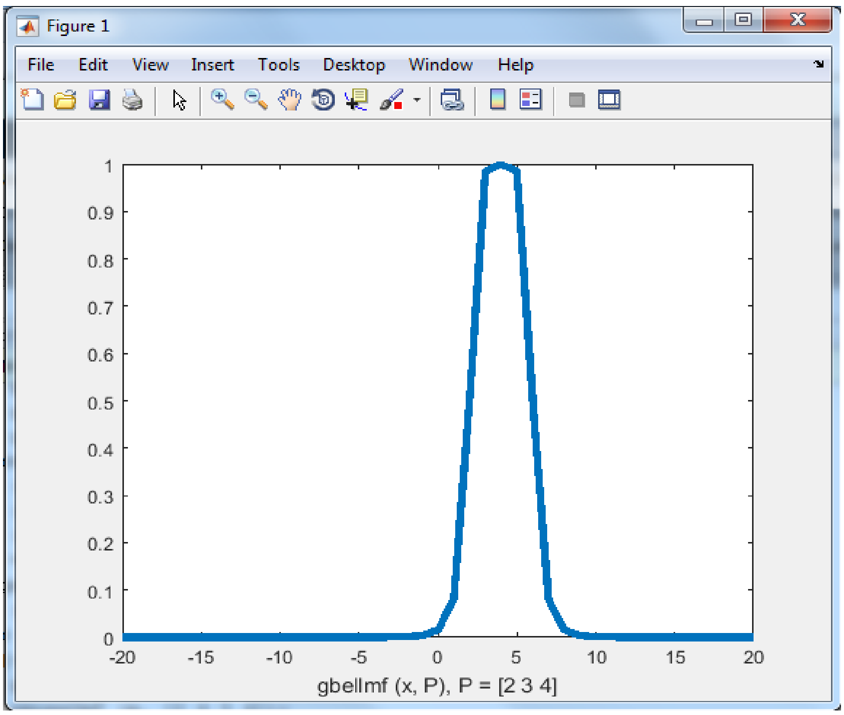


Рис. 4. Функция принадлежности "обобщенный колокол"

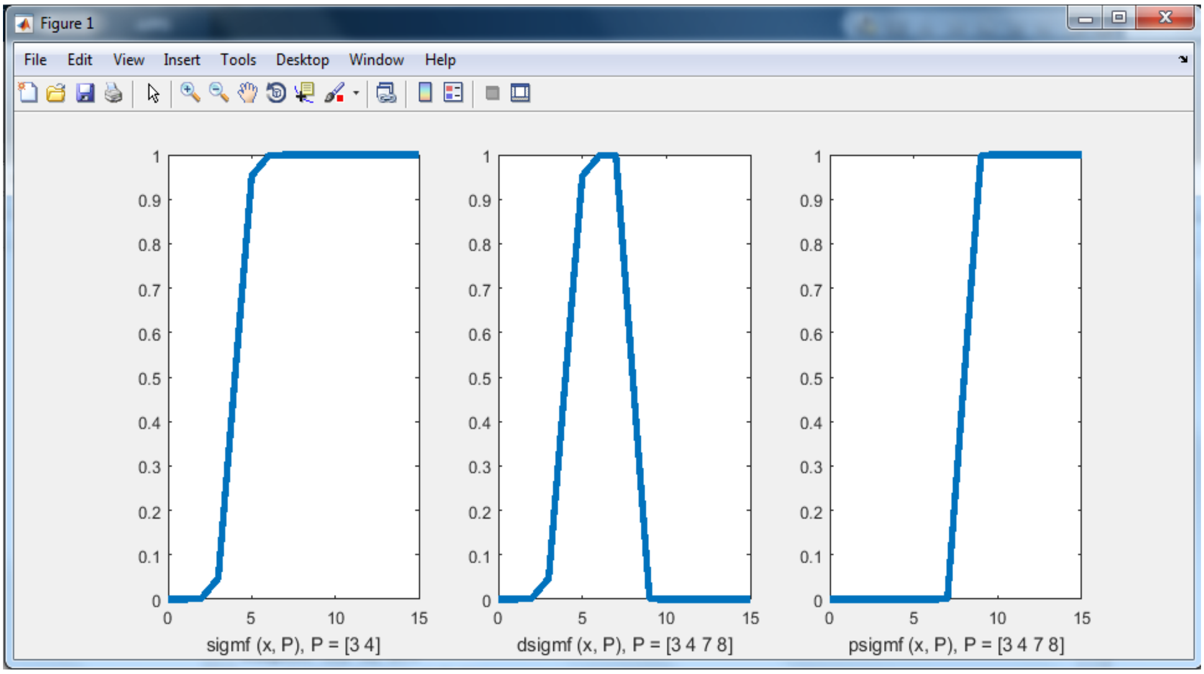
4. Построим набор сигмоидных функций:
• основную одностороннюю, которая открыта слева или справа;
• дополнительную двухстороннюю;
• дополнительную несимметричную.

В аналитической форме сигмоидная функция записывается так:

$$f(x,a,c) = \left\{\frac{1}{1+e^{-c(x-c)}}\right. . \tag{12}$$

В зависимости от знака параметра *a* ФП будет открыта или справа, или слева.
Дополнительная двухсторонняя сигмоидная функция зависит от четырёх параметров и представляет собой разность двух основных ФП, задаваемых двумя параметрами. Дополнительная несимметричная также зависит от четырёх параметров, но является произведением двух основных сигмоидных функций.

```
x = [0:15];
subplot (1, 3, 1);
y = sigmf (x,[3 4]);
plot (x, y, 'LineWidth', 4);
xlabel ('sigmf (x, P), P = [3 4]');
subplot (1, 3, 2);
y = dsigmf (x, [3 4 7 8]);
plot (x, y, 'LineWidth', 4);
xlabel ('dsigmf (x, P), P = [3 4 7 8]');
subplot (1, 3, 3);
y = psigmf (x, [3 4 7 8]);
plot (x, y, 'LineWidth', 4);
xlabel ('psigmf (x, P), P = [3 4 7 8]');
```



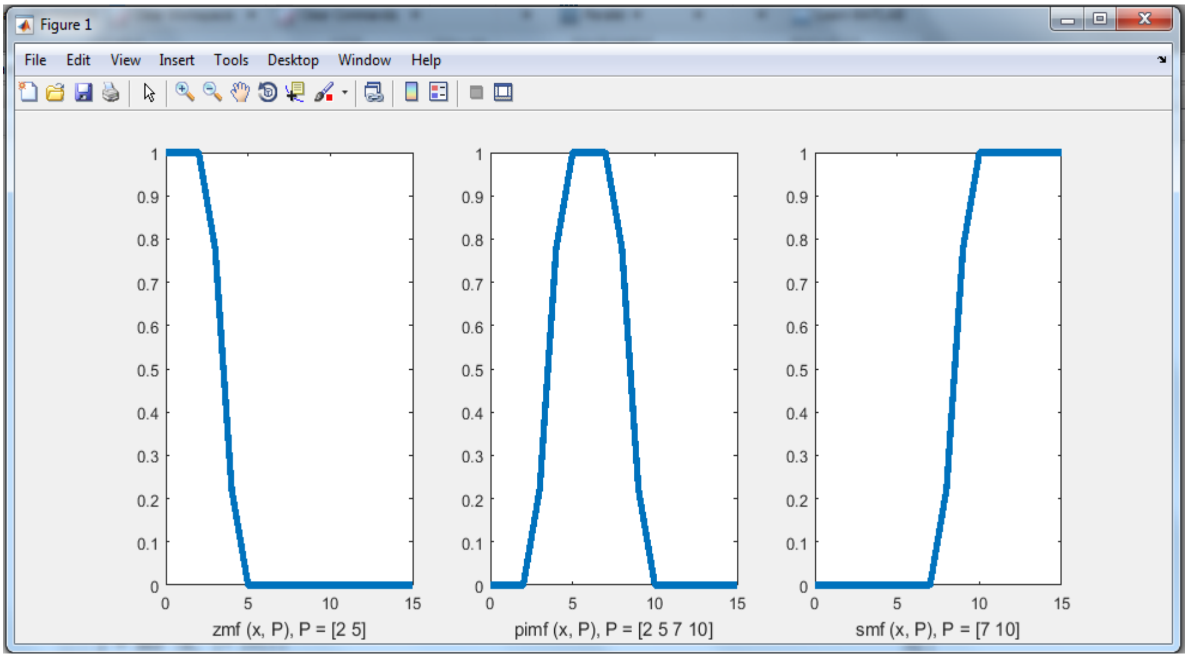
(а) (б) (в)

Рис. 5. Сигмоидные функции: основная односторонняя, открытая справа (а), дополнительная двухсторонняя (б) и дополнительная несимметричная (в)

5. Построим набор полиномиальных функций принадлежности (Z-, PI-и S-функций).

Эти функции формируются на основе полиномиальных кривых.

```
x = [0:15];
subplot (1, 3, 1);
y = zmf (x,[2 5]);
plot (x, y, 'LineWidth', 4);
xlabel ('zmf (x, P), P = [2 5]');
subplot (1, 3, 2);
y = pimf (x, [2 5 7 10]);
plot (x, y, 'LineWidth', 4);
xlabel ('pimf (x, P), P = [2 5 7 10]');
subplot (1, 3, 3);
y = smf (x, [7 10]);
plot (x, y, 'LineWidth', 4);
xlabel ('smf (x, P), P = [7 10]');
```



(а) (б) (в)

Рис. 6. Полиномиальные функции принадлежности: Z-функция (а), PI-функция (б) и S-функция (в)

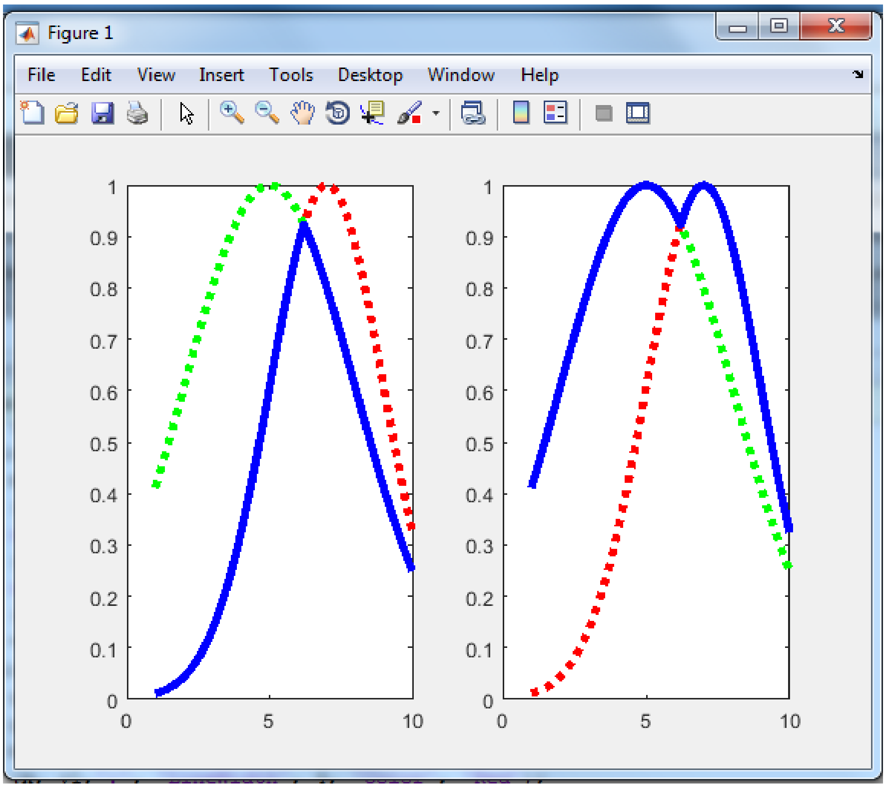
6. Построим минимаксную интерпретацию логических операторов с использованием операций поиска минимума и максимума.

Конъюнкция и дизъюнкция в данной интерпретации определены как минимум и максимум двух функций соответственно:

$$\begin{aligned} y_\cap &= \min(y_1, y_2), \\ y_\cup &= \max(y_1, y_2). \end{aligned} \tag{13}$$

```
x = 1:0.1:10;
subplot (1, 2, 1);
y1 = gaussmf (x, [2 7]);
y2 = gaussmf (x, [3 5]);
y3 = min ([y1; y2]);
plot (x, y1,':', 'LineWidth', 4, 'Color', 'Red');
hold on;
```

```
plot (x, y2,':', 'LineWidth', 4, 'Color', 'Green');
hold on;
plot (x, y3, 'LineWidth', 4, 'Color', 'Blue');
subplot (1, 2, 2);
y4 = max ([y1; y2]);
plot (x, y1,':', 'LineWidth', 4, 'Color', 'Red');
hold on;
plot (x, y2,':', 'LineWidth', 4, 'Color', 'Green');
hold on;
plot (x, y4, 'LineWidth', 4, 'Color', 'Blue');
```



(а) (б)

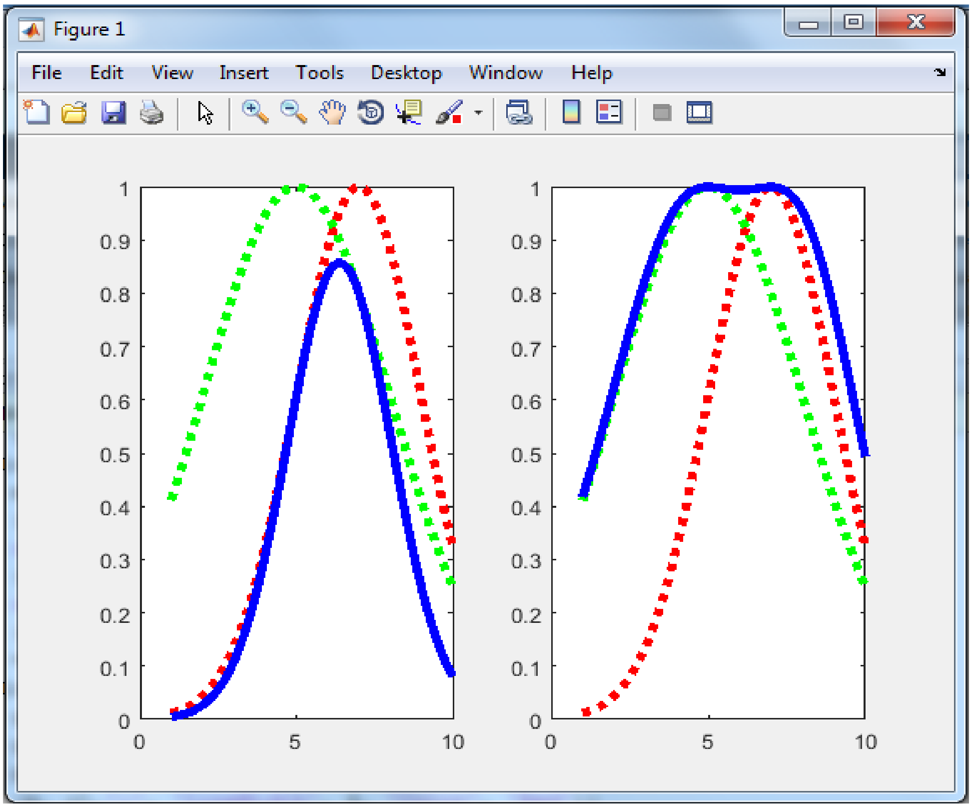
Рис. 7. Минимаксная интерпретация пересечения (а) и объединения (б) нечетких множеств

7. Построим вероятностную интерпретацию конъюнктивных и дизъюнктивных операторов.

В рамках данной интерпретации конъюнктивный оператор представляет собой вычисление алгебраического произведения (prod), а дизъюнктивный – алгебраической суммы (probor).

```
x = 1:0.1:10;
subplot (1, 2, 1);
y1 = gaussmf (x, [2 7]);
y2 = gaussmf (x, [3 5]);
y3 = prod ([y1; y2]);
plot (x, y1,':', 'LineWidth', 4, 'Color', 'Red');
hold on;
plot (x, y2,':', 'LineWidth', 4, 'Color', 'Green');
hold on;
plot (x, y3, 'LineWidth', 4, 'Color', 'Blue');
subplot (1, 2, 2);
y4 = probor ([y1; y2]);
plot (x, y1,':', 'LineWidth', 4, 'Color', 'Red');
```

```
hold on;
plot (x, y2,':', 'LineWidth', 4, 'Color', 'Green');
hold on;
plot (x, y4, 'LineWidth', 4, 'Color', 'Blue');
```



(а) (б)

Рис. 8. Вероятностная интерпретация пересечения (а) и объединения (б) нечетких множеств

8. Построим дополнение нечеткого множества, которое описывает некоторое размытое суждение и представляет собой математическое описание вербального выражения, отрицающего это нечеткое множество.

В данном случае значение результирующей ФП получается путём вычитания исходной из единицы.

```
x = 0:0.1:10;
y1 = gaussmf (x, [2 6]);
y2 = 1-y1;
plot (x, y1,':', 'LineWidth', 4);
hold on;
plot (x, y2, 'LineWidth', 4);
```

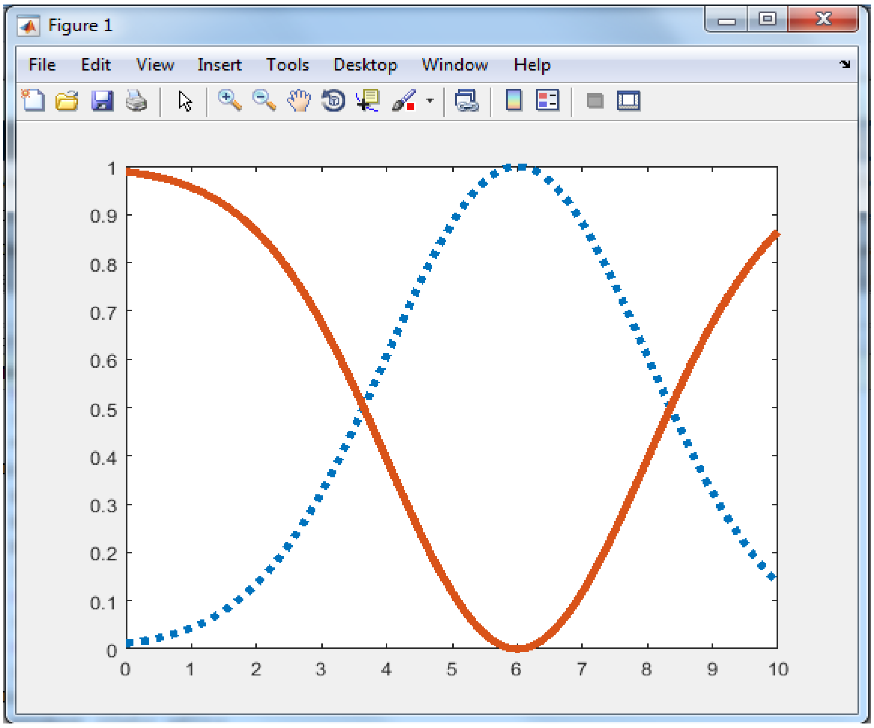


Рис. 9. Дополнение нечеткого множества

Построение нечёткой аппроксимирующей системы в пакете Fuzzy Logic Toolbox

Командой (функцией) *Fuzzy* из режима командной строки запускается основная интерфейсная программа пакета *Fuzzy Logic* – редактор нечеткой системы вывода (*Fuzzy Inference System Editor, FIS Editor, FIS*-редактор).

Вид открывающегося при этом окна приведен на рис.16.

Главное меню редактора содержит позиции:

File – работа с файлами моделей (их создание, сохранение, считывание и печать);

Edit – операции редактирования (добавление и исключение входных и выходных переменных);

View – переход к дополнительному инструментарию.

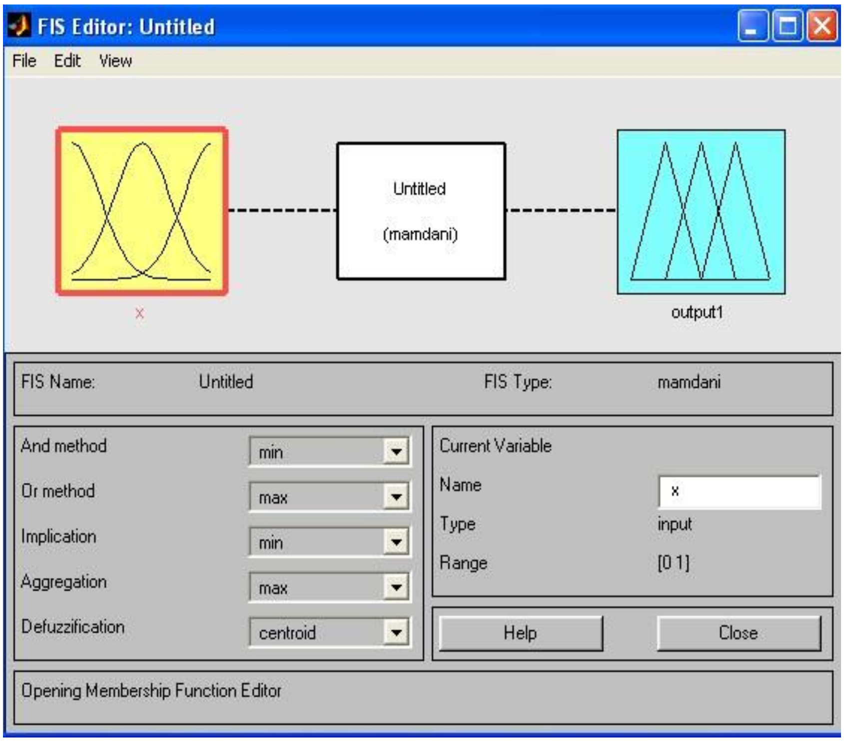


Рис. 10. Вид окна *Fis Editor*

Попробуем сконструировать нечеткую систему, отображающую зависимость между переменными *x* и *y*, заданную с помощью табл.1 (легко видеть, что представленные в таблице данные отражают зависимость $y = x^2$).

Таблица 1 – Значения *x* и *y*

x	-1	0,6	0	0,4	1
y	1	0,36	0	0,16	1

Щелкнем левой кнопкой мыши по блоку, озаглавленному *input* 1 (вход 1). Затем в правой части редактора в поле, озаглавленном *Name* (Имя), вместо *input* 1 введем обозначение нашего аргумента, т.е. *x*. Обратим внимание, что если теперь сделать где-нибудь (вне блоков редактора) однократный щелчок мыши, то имя отмеченного блока изменится на *x*; то же достигается нажатием после ввода клавиши *Enter*.

Требуемые действия отобразим следующими пунктами.

1. В позиции меню *File* выбираем опцию *New Sugeno FIS* (новая система типа *Sugeno*), при этом в блоке, отображаемом белым квадратом, в верхней части окна редактора появится надпись *Untitled2* (*Sugeno*).

2. Дважды щелкнем по этому блоку. Перед нами откроется окно редактора функций принадлежности – *Membership Function Editor* (см. рис.17). Войдем в позицию меню *Edit* данного редактора и выберем в нем опцию *Add MFs* (*Add Membership Funcions* – Добавить функций принадлежности). При этом появится диалоговое окно (рис.18), позволяющее задать тип (*MF type*) и количество (*Number of MFs*) функций принадлежности (в данном случае все относится к входному сигналу, т. е. к переменной *x*). Выберем гауссовы функции принадлежности (*gaussmf*), а их количество зададим равным пяти – по числу значений аргумента в табл.1. Подтвердим ввод информации нажатием кнопки ОК, после чего произойдет возврат к окну редактора функций принадлежности.

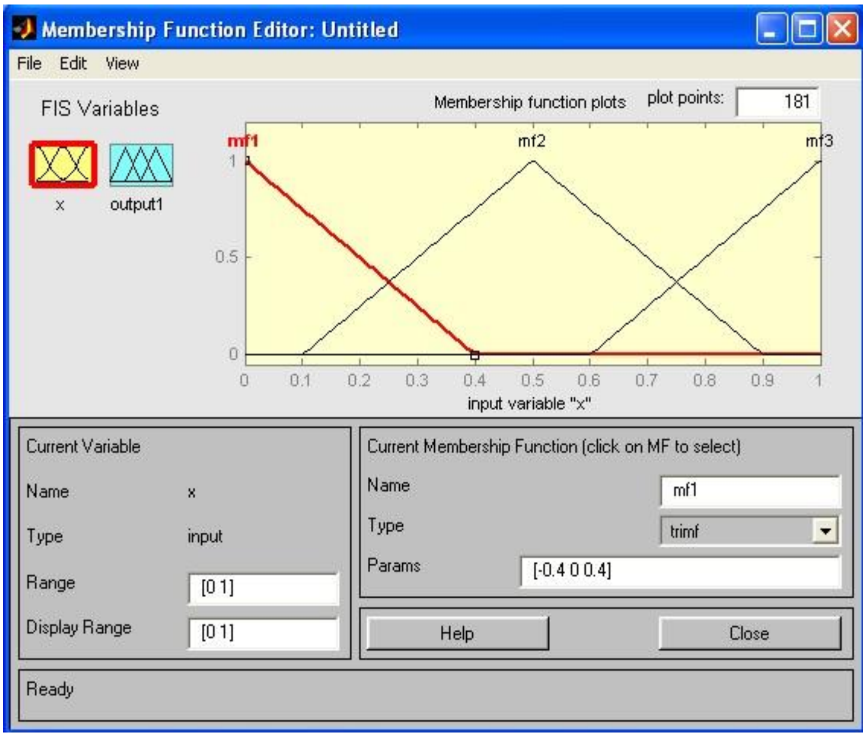


Рис. 11. Окно редактора функций принадлежности

3. В поле *Range* (Диапазон) установим диапазон изменения *x* от -1 до +1, т.е. диапазон, соответствующий табл. 1. Щелкнем затем левой кнопкой мыши где-нибудь в поле редактора (или нажмем клавишу ввода Enter). Обратим внимание, что после этого произойдет соответствующее изменение диапазона в поле *Display Range* (Диапазон дисплея).

4. Обратимся к графикам заданных нами функций принадлежности, изображенным в верхней части окна редактора функций принадлежности. Заметим, что для успешного решения поставленной задачи необходимо чтобы ординаты максимумовэтих функций совпадали с заданными значениями аргумента *x*. Для левой, центральной и правой функций такое условие выполнено, но две другие необходимо «подвинуть» вдоль оси абсцисс. «Передвижка» делается весьма просто: подводим курсор к нужной кривой и щелкаем левой кнопкой мыши. Кривая выбирается, окрашиваясь в красный цвет, после чего с помощью курсора ее и можно подвинуть в нужную сторону (более точную установку можно провести, изменяя числовые значения в поле *Params* (Параметры) – в данном случае каждой функции принадлежности соответствуют два параметра, при этом первый определяет размах кривой, а второй – положение ее центра). Для выбранной кривой, кроме этого, в поле Name можно изменять имя (завершая ввод каждого имени нажатием клавиши Enter).

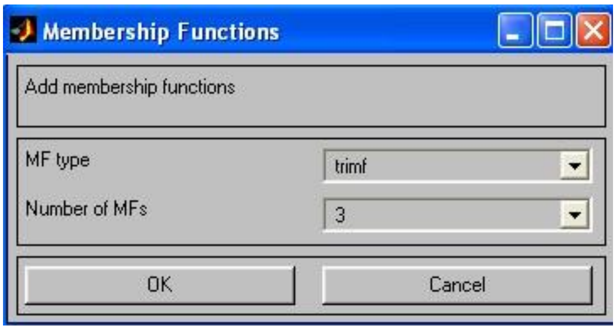


Рис.12. Диалоговое окно задания типа и количества функций принадлежности

Проделаем требуемые перемещения кривых и зададим всем пяти кривым новые имена, например:

- самой левой – *bn*,
- следующей – *n*,
- центральной – *z*,
- следующей за ней справа – *p*,
- самой левой – *bp*.

Нажмем кнопку *Close* и выйдем из редактора функций принадлежности, возвратившись при этом в окно редактора нечеткой системы (*FIS Editor*).

6. Сделаем однократный щелчок левой кнопкой мыши по голубому квадрату (блоку), озаглавленному *output* l (выход 1). В окошке *Name* заменим имя *output* l на *y* (как в пункте 2).

7. Дважды щелкнем по отмеченному блоку и перейдем к программе – редактору функций принадлежности. В позиции меню *Edit* выберем опцию *Add MFs*. Появляющееся диалоговое окно вида рис. 3 позволяет задать теперь в качестве функций принадлежности только линейные (*linear*) или постоянные (*constant*) – в зависимости от того, какой алгоритм *Sugeno* (1-го или 0-го порядка) мывыбираем. Если в вашем компьютере установлена версия, в которой нет данных функций принадлежностей, то можно оставить по умолчанию – *trimf*. Это, конечно, повлияет на результат, поэтому можно поэкспериментировать, изменяя тип функций принадлежности.

В рассматриваемой задаче необходимо выбрать постоянные функции принадлежности с общим числом 4 (по числу различных значений *y* в табл.1). Подтвердим введенные данные нажатием кнопки ОК, после чего произойдет возврат в окно редактора функций принадлежности.

Рис. 13. Параметры функций принадлежности переменной *y*

8. Обратим внимание, что здесь диапазон (*Range*) изменения, устанавливаемый по умолчанию – [0, 1], менять не нужно. Изменим лишь имена функций принадлежности (их графики при использовании алгоритма *Sugeno* для выходных переменных не приводятся), например, задав их как соответствующие числовые значения *y*, т.е. 0, 0.16, 0.36, 1; одновременно эти же числовые значение введем в поле *Params* (рис.19). Затем закроем окно нажатием кнопки *Close* и вернемся в окно *FIS*-редактора.

9. Дважды щелкнем левой кнопкой мыши по среднему (белому) блоку, при этом раскроется окно еще одной программы – редактора правил (*Rule Editor*). Введем соответствующие правила. При вводе каждого правила необходимо обозначить соответствие между каждой функцией принадлежности аргумента *x* и числовым значением *y*. Кривая, обозначенная нами *bn*, соответствует $x = -1$, т.е. $y = 1$. Выберем, поэтому в левом поле (с заголовком *x isbn)*, а в правом 1 и нажмем кнопку *Add rule* (Добавить правило). Введенное правило появится в окне правил и будет представлять собой запись:

If (*x* is *bn*) *then* (*y* is 1)

Аналогично поступим для всех других значений *x*, в результате чего сформируется набор из 5 правил (см. рис.20). Закроем окно редактора правил и возвратимся в окно *FIS*-редактора. Построение системы закончено и можно начать эксперименты по ее исследованию. Заметим, что большинство опций выбиралось нами по умолчанию.

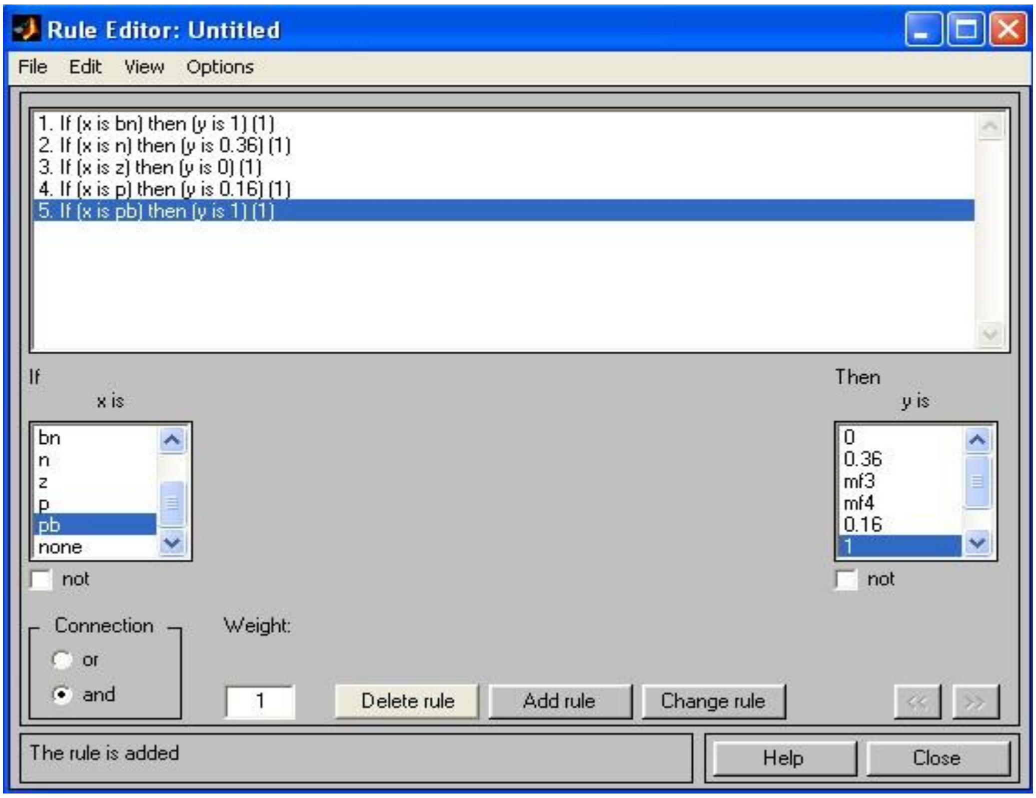


Рис. 14. Окно редактора правил

10. Предварительно сохраним на диске (используя пункты меню *File/Save to disk as...*) созданную систему под каким-либо именем, например, *Proba*.

11. Выберем позицию меню *View*. Как видно из выпадающего при этом подменю, с помощью пунктов *Edit membership functions* и *Edit rules* можно совершить переход к двум выше рассмотренным программам – редакторам функций принадлежности и правил (то же

можно сделать и нажатием клавиш *Ctrl*+2 или *Ctrl*+3). Но сейчас нас будут интересовать два других пункта – *View rules* (Просмотр правил) и *View surface* (Просмотр поверхности). Выберем пункт *View rules*, при этом откроется окно (рис.21) еще одной программы – просмотра правил (*Rule Viewer*).

12. В правой части окна в графической форме представлены функции принадлежности аргумента ж, в левой – переменной выхода y с пояснением механизма принятия решения. Красная вертикальная черта, пересекающая графики в правой части окна, которую можно перемещать с помощью курсора, позволяет изменять значения переменной входа (это же можно делать, задавая числовые значения в поле *Input* (Вход)), при этом соответственно изменяются значения y в правой верхней части окна. Зададим, например, x – 0.5 в поле *Input* и нажмем затем клавишу ввода (*Enter*). Значение y сразу изменится и станет равным 0.202. Таким образом, с помощью построенной модели и окна просмотра правил можно решать задачу интерполяции, т.е. задачу, решение которой и требовалось найти. Изменение аргумента путем перемещения красной вертикальной линии очень наглядно демонстрирует, как система определяет значения выхода.

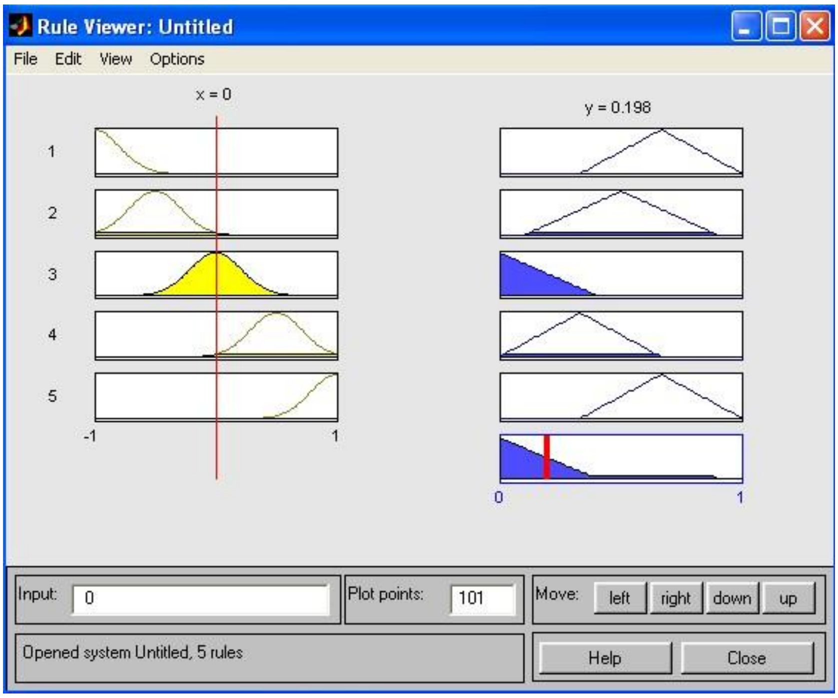


Рис. 15. Окно просмотра правил

13. Закроем окно просмотра правил и выбором пункта меню *View/View surface* перейдем к окну просмотра поверхности отклика (выхода), в нашем случае – к просмотру кривой *y(x)* (рис.22). Видно, что смоделированное системой по таблице данных (табл.1) отображение не очень-то напоминает функцию x^2. Ну что ж, ничего удивительного в этом нет: число экспериментальных точек невелико, да и параметры функций принадлежности (для *x)* выбраны, скорее всего, неоптимальным образом. Ниже мы рассмотрим возможность улучшения качества подобной модели.

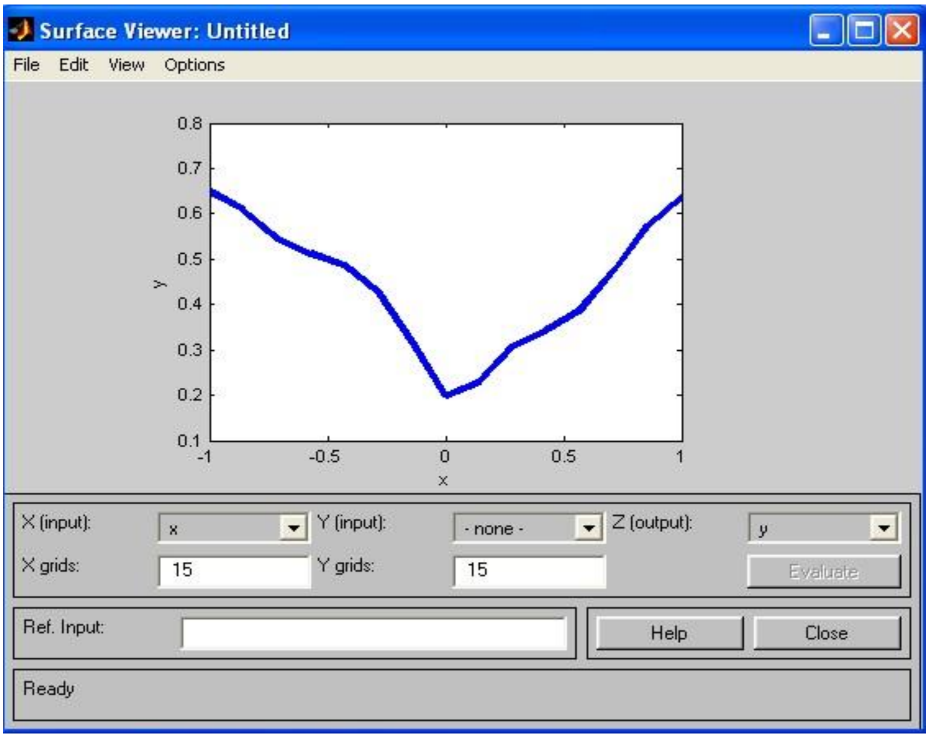


Рис. 16. Окно просмотра поверхности отклика

В заключение рассмотрения примера отметим, что с помощью вышеуказанных программ-редакторов на любом этапе проектирования нечеткой модели в нее можно внести необходимые коррективы, вплоть до задания какой-либо особенной пользовательской функции принадлежности. Из опций, устанавливаемых в FIS-редакторе по умолчанию при использовании алгоритма Sugeno, можно отметить:

- логический вывод организуется с помощью операции умножения (*prod*);
- композиция – с помощью операции логической суммы (вероятностного ИЛИ, *probor*);
- приведение к четкости – дискретным вариантом центроидного метода (взвешенным средним, *wtaver*). Используя соответствующие поля в левой нижней части окна *FIS*-редактора, данные опции можно, при желании, изменить.

Задание

1. Ознакомиться с основными видами функций принадлежности нечётких множеств.
2. Ознакомиться со стандартными видами функций принадлежности, входящими в пакет Fuzzy Logic Toolbox.
3. Ознакомиться с возможностями системы MATLAB для работы с нечёткими множествами.
4. Построить графики всех описанных функций принадлежности со своими параметрами.
5. Для одной из функций принадлежности привести пример использования.
6. Построить графики, иллюстрирующие операции объединения и пересечения (двумя методами), а также операцию дополнения.
7. Сконструировать нечёткую систему, отображающую зависимость между переменными *x* и *y*, заданную с помощью табл.2. По результатам работы определить тип кривой.

Список индивидуальных данных

Таблица 2 – Варианты заданий

Варианты	Значение аргумента и функции					
1	x	-1	-0.5	0	0.2	1
	y	1	0.25	0	0.4	1
2	x	-1	-0.6	0.2	0.4	1
	y	-1	-1.67	5	2.5	1
3	x	-1	-0.5	0	0.3	1
	y	-1	-0.13	0	0.27	1
4	x	-1	-0.6	0	0.3	1
	y	0	0.8	1	0.95	0
5	x	-1	-0.5	0	0.2	1
	y	1	-0.125	0	0.008	1
6	x	-1	-0.6	0.2	0.4	1
	y	0	-0.64	-0.96	-0.84	0
7	x	-1	-0.5	0	0.3	1
	y	-3	-2	-1	-0.4	1
8	x	-1	-0.6	0	0.3	1
	y	0.5	0.09	0	0.0225	0.5
9	x	-1	-0.5	0	0.2	1
	y	0.5	0.03125	0	0.0008	0.5
10	x	-1	-0.6	0.2	0.4	1
	y	-1	2.78	25	6.25	1
11	x	-1	-0.5	0	0.3	1
	y	-1	1.8	3	3.6	5
12	x	-1	-0.6	0	0.3	1
	y	-1	-0.216	0	0.027	1
13	x	-1	-0.5	0	0.2	1
	y	-2	-1.5	-1	-0.8	0
14	x	-1	-0.6	0.2	0.4	1
	y	-2	-1.2	0.4	2.5	1
15	x	-1	-0.5	0	0.3	1
	y	0	0.25	0.5	0.65	1
16	x	-1	-0.5	0	0.3	1
	y	-1	-1.75	-1	-0.31	0
17	x	-1	-0.5	0	0.2	1
	y	0	0.25	1	1.44	4
18	x	-1	-0.6	0.2	0.4	1
	y	-0.67	-0.2	0.07	0.13	0.67
19	x	-1	-0.5	0	0.3	1
	y	0.67	-0.34	0.5	0.18	1
20	x	-1	-0.6	0	0.3	1
	y	-2	-1.6	0	0.7	0

Контрольные вопросы

1. Что такое функция принадлежности?
2. Чем различаются нечёткая и лингвистическая переменные?
3. Что такое нечёткое отношение?
4. Какими свойствами обладают нечёткие отношения?

5. Какие функции принадлежности вам известны?
6. Каким образом можно сконструировать нечёткую систему, отображающую зависимость между двумя переменными?
7. Для чего, по-вашему, нужен аппарат нечёткой логики?