

## Лабораторная работа №2

### Нечёткие системы

**Цель работы:** изучить методы построения нечётких систем в задачах математического моделирования и принятия решений с использованием инструментария нечёткой логики программной среды MATLAB.

**Продолжительность работы:** 4 часа.

### Теоретические сведения

#### Системы нечёткой логики

Базовая конфигурация простой системы нечёткой логики представлена на рис. 1. Базис нечётких правил состоит из набора нечётких IF-THEN-правил, а механизм нечёткого вывода на основе нечёткой логики использует эти правила для отображения нечётких множеств из входного множества высказываний  $X$  в нечёткие множества из множества высказываний  $Y$  на выходе системы. Нечёткие IF-THEN-правила имеют вид:

$$IF (x_1 IS F_1^{(p)}) AND \dots AND (x_n IS F_n^{(p)}) THEN y IS G^{(p)}, \quad (1)$$

где  $F_i^{(p)}$  и  $G^{(p)}$  - нечёткие множества,  $\bar{x} = (x_1, \dots, x_n)^T \in X$  и  $y \in Y$  - входные и выходная переменные соответственно.

Главным недостатком простых систем нечёткой логики является то обстоятельство, что её входы и выходы – нечёткие множества, тогда как в большинстве технических систем входы и выходы принимают реальные значения.

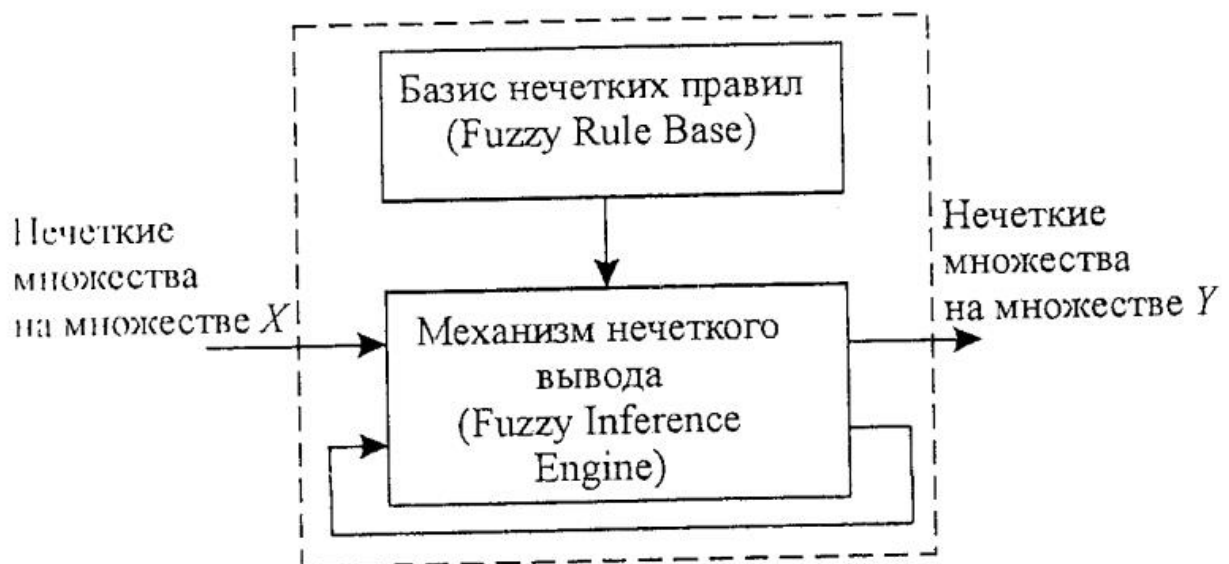


Рис. 1. Конфигурация простой системы нечёткой логики

#### Нечёткие модели

Нечёткие модели – математические модели, в основе вычисления которых лежит нечёткая логика. Они применяются, когда предмет исследования имеет слабую

Этапы построения модели:

1. Определение входных и выходных параметров модели.

3. Выбор одного из методов нечёткого логического вывода.

Два основных метода нечёткого вывода – методы Мамдани и Сугено.

$$\begin{aligned}
& IF \left( x_1 IS A_1^{(1)} \right) AND \left( x_2 IS A_2^{(1)} \right) AND ... AND \left( x_n IS A_n^{(1)} \right), \\
& \quad THEN (Y IS d_1)(W_1) \\
& OR \\
& .. ... .. \\
& OR \\
& IF \left( x_1 IS A_1^{(M)} \right) AND \left( x_2 IS A_2^{(M)} \right) AND ... AND \left( x_n IS A_n^{(M)} \right), \\
& \quad THEN (Y IS d_M)(W_M), 
\end{aligned} \tag{2}$$

Введём новое обозначение  $\mu_t(v)$  – функция принадлежности входной или выходной нечёткой переменной  $v$  нечёткому терму  $t$ .

$$\mu_{d_j}(X) = \bigvee_{j=1}^M W_j \cdot \bigwedge_{i=1}^n |\mu_{A_i^{(j)}}(X_i)|, \quad (3)$$

После нахождения всех  $\mu_{d_j}$  получим  $M$  новых функций принадлежности, которые в совокупности будут образовывать новое нечёткое множество, которое мы обозначим  $\tilde{Y}$ , соответствующее входному вектору  $X$ .

$$\tilde{Y} = \frac{\mu_{d_1}(X)}{d_1} + \frac{\mu_{d_2}(X)}{d_2} + \dots + \frac{\mu_{d_M}(X)}{d_M}. \quad (4)$$

Далее необходимо:

- После этого мы получаем результирующее нечёткое множество, дефазификация которого и даст нам точный выход системы.

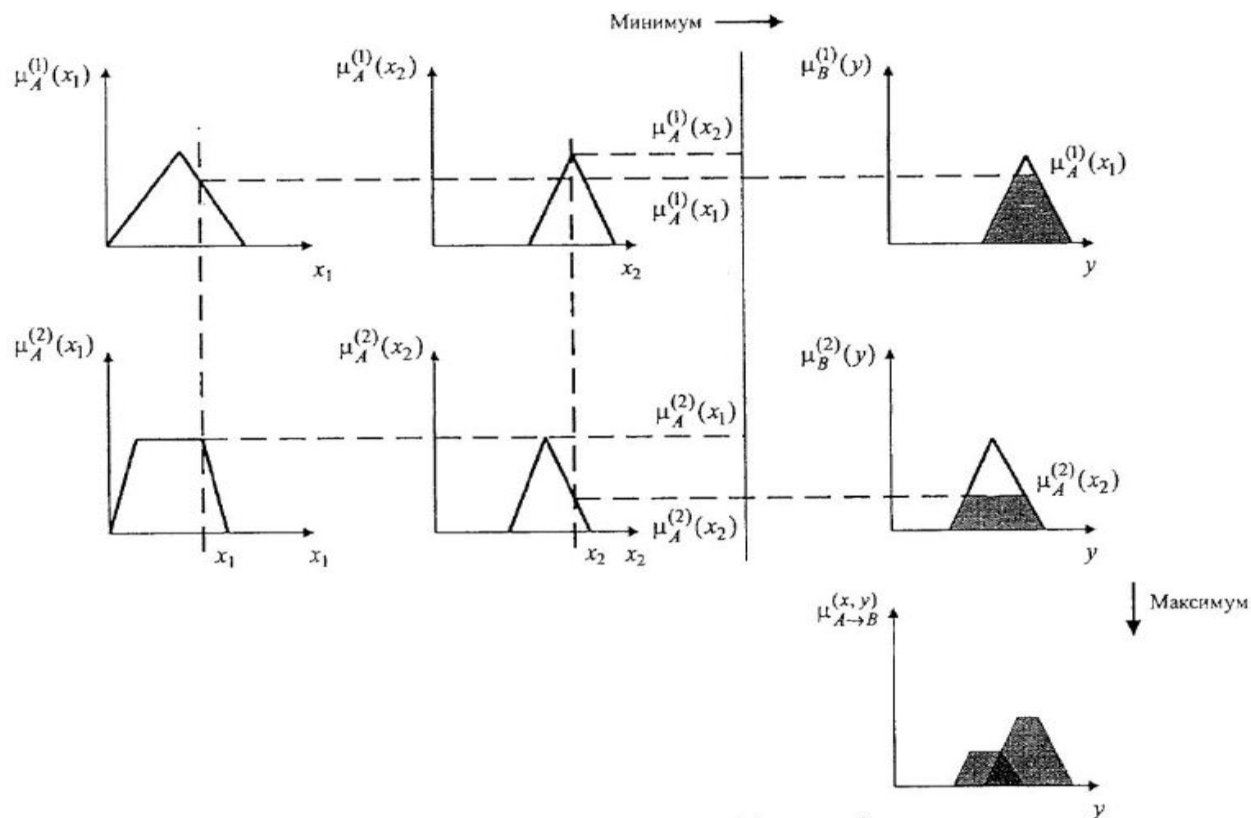


Рис. 2. Пример системы вывода Мамдани-Заде

[illegible]

База знаний Сугено аналогична базе знаний Мамдани за исключением заключений правил  $d_j$ , которые задаются не нечёткими термами, а линейной функцией от входов:

$$d_j = p_{j0} + \sum_{i=1}^n p_{ji} x_i. \quad (6)$$

Пусть  $\mu_t(v)$  – функция принадлежности входной или выходной нечёткой переменной  $v$  нечёткому терму  $t$ .

В нечётком логическом выводе Сугено наиболее часто используются вероятностное ИЛИ как s-норма и произведение как t-норма.

Дальнейшее определение нечёткого множества  $\tilde{Y}$  вычисляется по формуле (4).

Обратим внимание, что в отличие от результата вывода Мамдани, приведённое выше нечёткое множество является обычным нечётким множеством первого порядка, заданным на множестве чётких чисел. Результирующее значение выхода определяют дефазификацией выходного нечёткого множества посредством нахождения взвешенного среднего или взвешенной суммы.

## Пример нечёткой системы

**Задача.** Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи определения количества линий в службе поддержки (учитывать количество обслуживаемых клиентов, среднюю частоту обращения в службу одного клиента, среднее время обслуживания одной заявки, квалификацию персонала и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).

Алгоритм выбора числа линий может быть задан следующими правилами:

- 1) если количество обслуживаемых клиентов большое, то количество линий большое;
- 2) если количество обслуживаемых клиентов среднее, то количество линий среднее;
- 3) если количество обслуживаемых клиентов малое, то количество линий малое;
- 4) если частота обращений высокая, то количество линий большое;
- 5) если частота обращений средняя, то количество линий среднее;
- 6) если частота обращений низкая, то количество линий малое;
- 7) если время обслуживания большое, то количество линий большое;
- 8) если время обслуживания среднее, то количество линий среднее;
- 9) если время обслуживания малое, то количество линий малое;
- 10) если квалификация персонала высокая, то количество линий малое;
- 11) если квалификация персонала средняя, то количество линий среднее;
- 12) если квалификация персонала низкая, то количество линий высокое;

Пусть количество обслуживаемых клиентов измеряется в штуках (от 0 до 1000), средняя частота обращения каждого из них в раза в сутки (от 0 до 100), время обслуживания заявки – в минутах (от 0 до 100), квалификация персонала – в баллах пятибалльной шкалы (2 – очень плохо, 5 – отлично, возможны дробные баллы), количество линий – в штуках (от 0 до 10). Зададим лингвистические переменные формулами.

«Малое» количество клиентов ( $N$ ):

$$m_{\text{малое}}(N) = \begin{cases} 1, & N \leq 300, \\ \frac{600 - N}{300}, & 300 < N < 600, \\ 0, & N \geq 600. \end{cases} \quad (7)$$

«Среднее» количество клиентов (N):

$$m_{\text{среднее}}(N) = \begin{cases} 0, N < 200 \text{ или } N \geq 800, \\ \frac{N - 200}{300}, 200 \leq N < 500, \\ \frac{800 - N}{300}, 500 \leq N < 800. \end{cases} \quad (8)$$

«Большое» количество клиентов (N):

$$m_{\text{большое}}(N) = \begin{cases} 0, N < 500, \\ \frac{N - 500}{300}, 500 \leq N < 800, \\ 1, N \geq 800. \end{cases} \quad (9)$$

«Низкая» частота обращений (f):

$$m_{\text{низкая}}(f) = \begin{cases} 1, f \leq 30, \\ \frac{60 - f}{30}, 30 < f < 60, \\ 0, f \geq 60. \end{cases} \quad (10)$$

«Средняя» частота обращений (f):

$$m_{\text{средняя}}(f) = \begin{cases} 0, f < 20 \text{ или } f \geq 80, \\ \frac{f - 20}{30}, 20 \leq f < 50, \\ \frac{80 - f}{30}, 50 \leq f < 80. \end{cases} \quad (11)$$

«Высокая» частота обращений (f):

$$m_{\text{высокая}}(f) = \begin{cases} 0, f < 50, \\ \frac{f - 50}{30}, 50 \leq f < 80, \\ 1, f \geq 80. \end{cases} \quad (12)$$

«Малое» время обслуживания (t):

$$m_{\text{малое}}(t) = \begin{cases} 1, t \leq 30, \\ \frac{60 - t}{30}, 30 < t < 60, \\ 0, t \geq 60. \end{cases} \quad (13)$$

«Среднее» время обслуживания (t):

$$m_{\text{среднее}}(t) = \begin{cases} 0, t < 20 \text{ или } t \geq 80, \\ \frac{t-20}{30}, 20 \leq t < 50, \\ \frac{80-t}{30}, 50 \leq t < 80. \end{cases} \quad (14)$$

«Большое» время обслуживания (t):

$$m_{\text{большое}}(t) = \begin{cases} 0, t < 50, \\ \frac{t-50}{30}, 50 \leq t < 80, \\ 1, t \geq 80. \end{cases} \quad (15)$$

«Низкая» квалификация персонала (q):

$$m_{\text{низкая}}(q) = \begin{cases} 1, q \leq 2, \\ \frac{3-q}{1}, 2 < q < 3, \\ 0, q \geq 3. \end{cases} \quad (16)$$

«Средняя» квалификация персонала (q):

$$m_{\text{средняя}}(q) = \begin{cases} 0, q < 3 \text{ или } q \geq 4, \\ \frac{q-3}{0,5}, 3 \leq q < 3,5, \\ \frac{4-q}{0,5}, 3,5 \leq q < 4. \end{cases} \quad (17)$$

«Высокая» квалификация персонала (q):

$$m_{\text{высокая}}(q) = \begin{cases} 0, q < 4, \\ \frac{q-4}{1}, 4 \leq q < 5, \\ 1, q \geq 5. \end{cases} \quad (18)$$

«Малое» количество линий (L):

$$m_{\text{малое}}(L) = \begin{cases} 1, L \leq 3, \\ \frac{6-L}{3}, 3 < L < 6, \\ 0, L \geq 6. \end{cases} \quad (19)$$

«Среднее» количество линий (L):

$$m_{\text{среднее}}(L) = \begin{cases} 0, L < 2 \text{ или } L \geq 8, \\ \frac{L-2}{3}, 2 \leq L < 5, \\ \frac{8-L}{3}, 5 \leq L < 8. \end{cases} \quad (20)$$

«Большое» количество линий (L):

$$m_{\text{большое}}(L) = \begin{cases} 0, L < 5, \\ \frac{L-5}{3}, 5 \leq L < 8, \\ 1, L \geq 8. \end{cases} \quad (21)$$

Рассмотрим теперь, как нечеткая экспертная система определяет количество линий. Пусть  $N = 450$ ,  $f = 55$ ,  $t = 33$ ,  $q = 3,75$ .

$$\begin{aligned} &N: \\ &m_{\text{малое}}(450) = 0,5; \\ &m_{\text{среднее}}(450) = 0,83; \\ &m_{\text{большое}}(450) = 0. \\ &f: \\ &m_{\text{низкая}}(55) = 0,17; \\ &m_{\text{средняя}}(55) = 0,83; \\ &m_{\text{высокая}}(55) = 0,17. \\ &t: \\ &m_{\text{малое}}(33) = 0,9; \\ &m_{\text{среднее}}(33) = 0,43; \\ &m_{\text{большое}}(33) = 0. \\ &q: \\ &m_{\text{низкая}}(3,75) = 0; \\ &m_{\text{средняя}}(3,75) = 0,5; \\ &m_{\text{высокая}}(3,75) = 0. \end{aligned} \quad (22)$$

Модификация для N:

```
>> x=[0:0.1:10];
yN1=ones([1,30]).*0.5;
yN1=[yN1, (6-x(31:60))./6];
yN1=[yN1, zeros([1,41])];
plot(x,yN1, 'Linewidth',4)
hold on
yN2=zeros([1,20]);
yN2=[yN2, (x(21:50)-2)./3.*0.83];
yN2=[yN2, (8-x(51:80))./3.*0.83];
yN2=[yN2, zeros([1,21])];
plot(x,yN2, 'Linewidth',4)
grid on
hold on
yN3= zeros([1,101]);
plot(x,yN3, 'Linewidth',4)
```

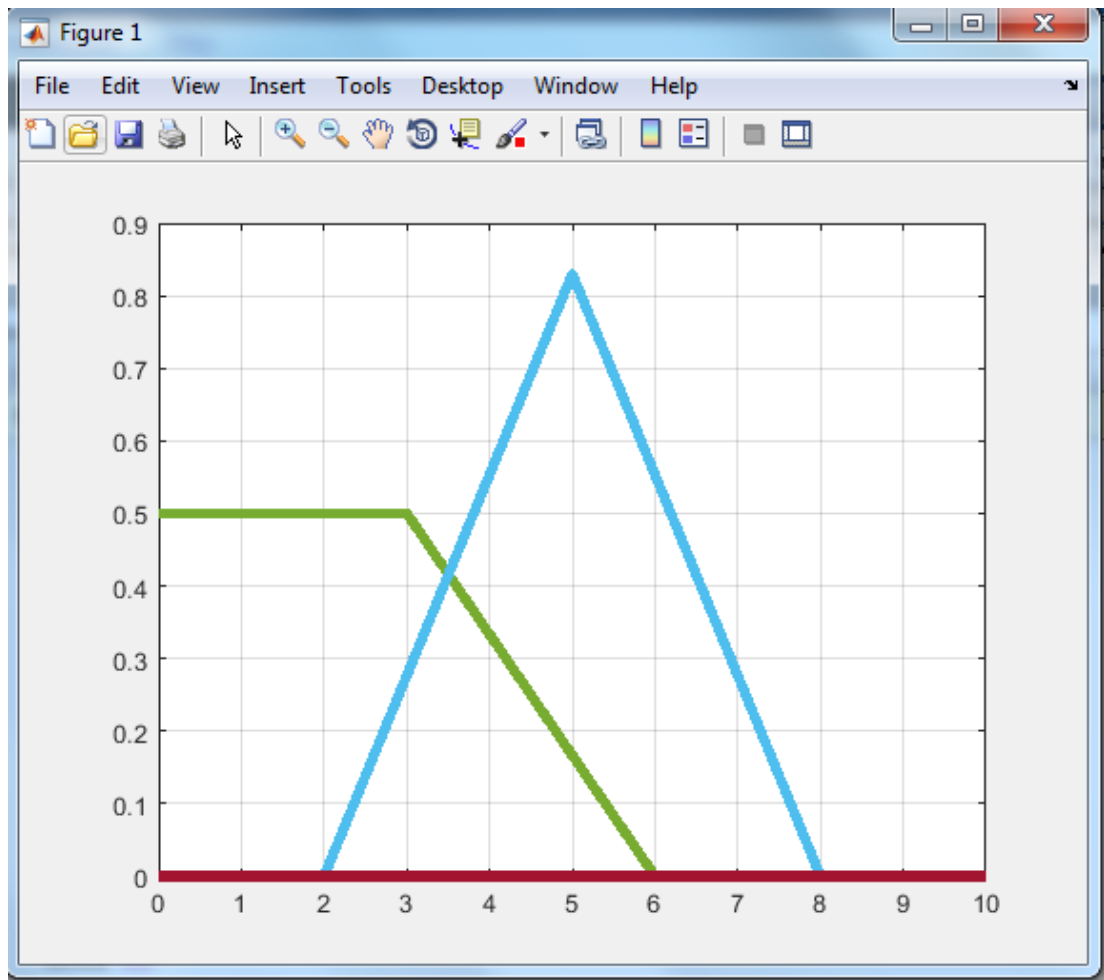


Рис.3. Модификация нечётких подмножеств для N

Модификация для f:

```
>> x=[0:0.1:10];
yf1=ones([1,30]).*0.17;
yf1=[yf1, (6-x(31:60))./3.*0.17];
yf1=[yf1, zeros([1,41])];
plot(x,yf1, 'Linewidth',4)
hold on
yf2=zeros([1,20]);
yf2=[yf2, (x(21:50)-2)./3.*0.83];
yf2=[yf2, (8-x(51:80))./3.*0.83];
yf2=[yf2, zeros([1,21])];
plot(x,yf2, 'Linewidth',4)
grid on
hold on
yf3=zeros([1,50]);
yf3=[yf3, (x(51:80)-5)./3.*0.17];
yf3=[yf3,ones([1,21]).*0.17];
plot(x,yf3, 'Linewidth',4)
```



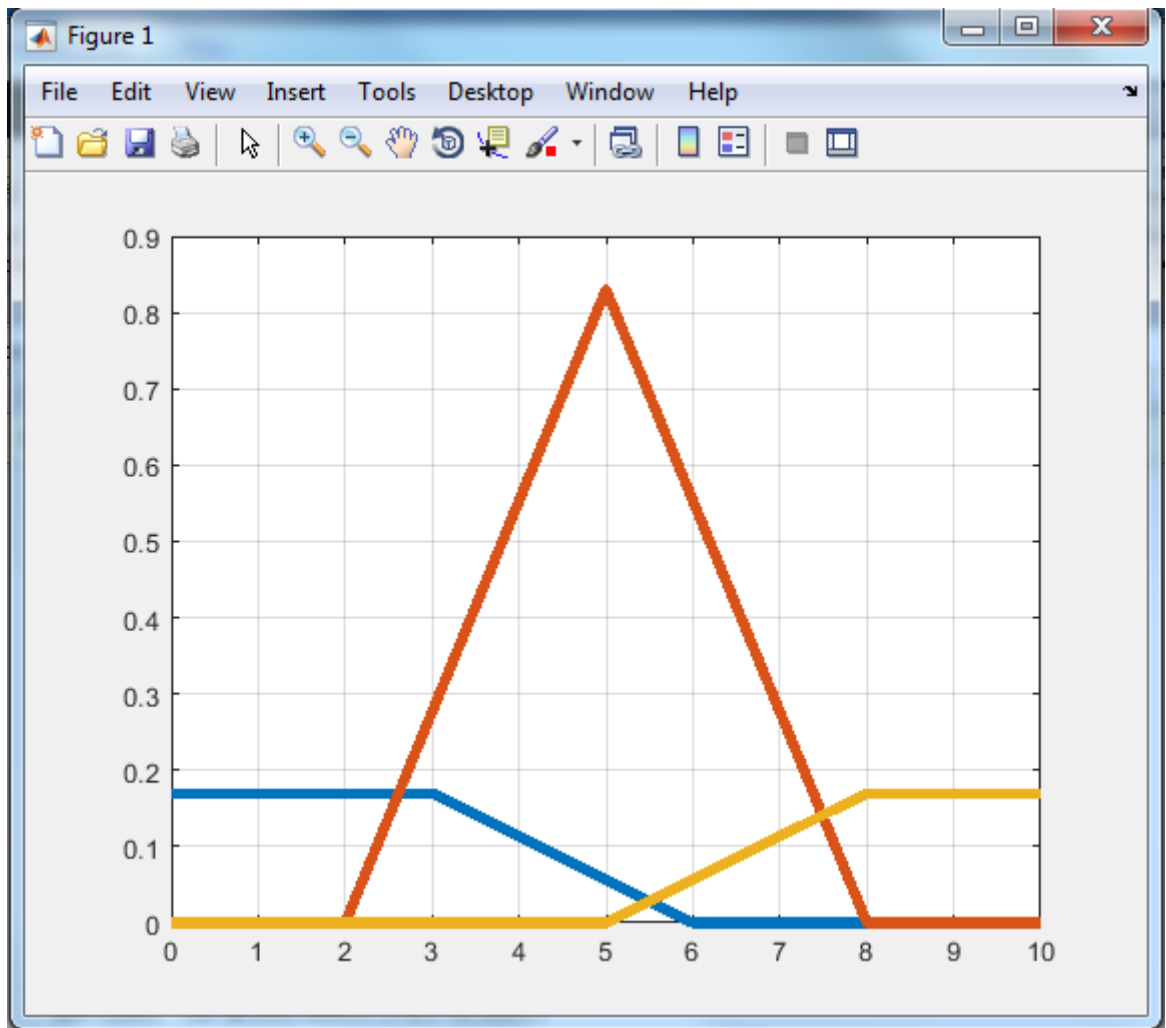


Рис.4. Модификация нечётких подмножеств для  $f$

Модификация для  $t$ :

```
>> x=[0:0.1:10];
yt1=ones([1,30]).*0.9;
yt1=[yt1, (6-x(31:60))./3.*0.9];
yt1=[yt1, zeros([1,41])];
plot(x,yt1, 'Linewidth',4)
hold on
yt2=zeros([1,20]);
yt2=[yt2, (x(21:50)-2)./3.*0.43];
yt2=[yt2, (8-x(51:80))./3.*0.43];
yt2=[yt2, zeros([1,21])];
plot(x,yt2, 'Linewidth',4)
grid on
hold on
yt3=zeros([1,101]);
plot(x,yt3, 'Linewidth',4)
```

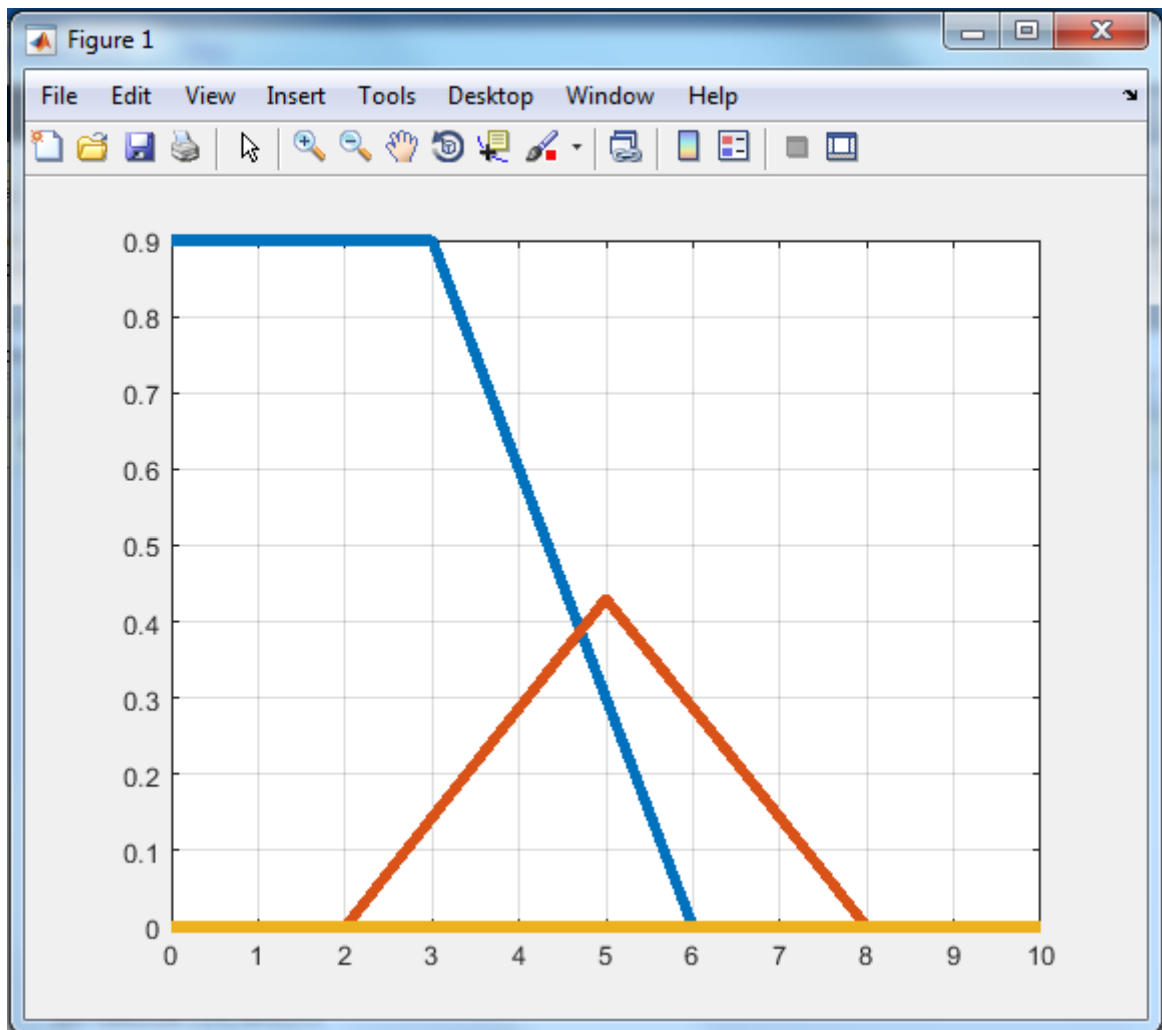


Рис.5. Модификация нечётких подмножеств для  $t$

Модификация для  $q$ :

```
>> x=[0:0.1:10];
yq1= zeros([1,101]);
plot(x,yq1, 'Linewidth',4)
hold on
yq2=zeros([1,20]);
yq2=[yq2, (x(21:50)-2)./3.*0.5];
yq2=[yq2, (8-x(51:80))./3.*0.5];
yq2=[yq2, zeros([1,21])];
plot(x,yq2, 'Linewidth',4)
grid on
hold on
yq3=zeros([1,101]);
plot(x,yq3, 'Linewidth',4)
```

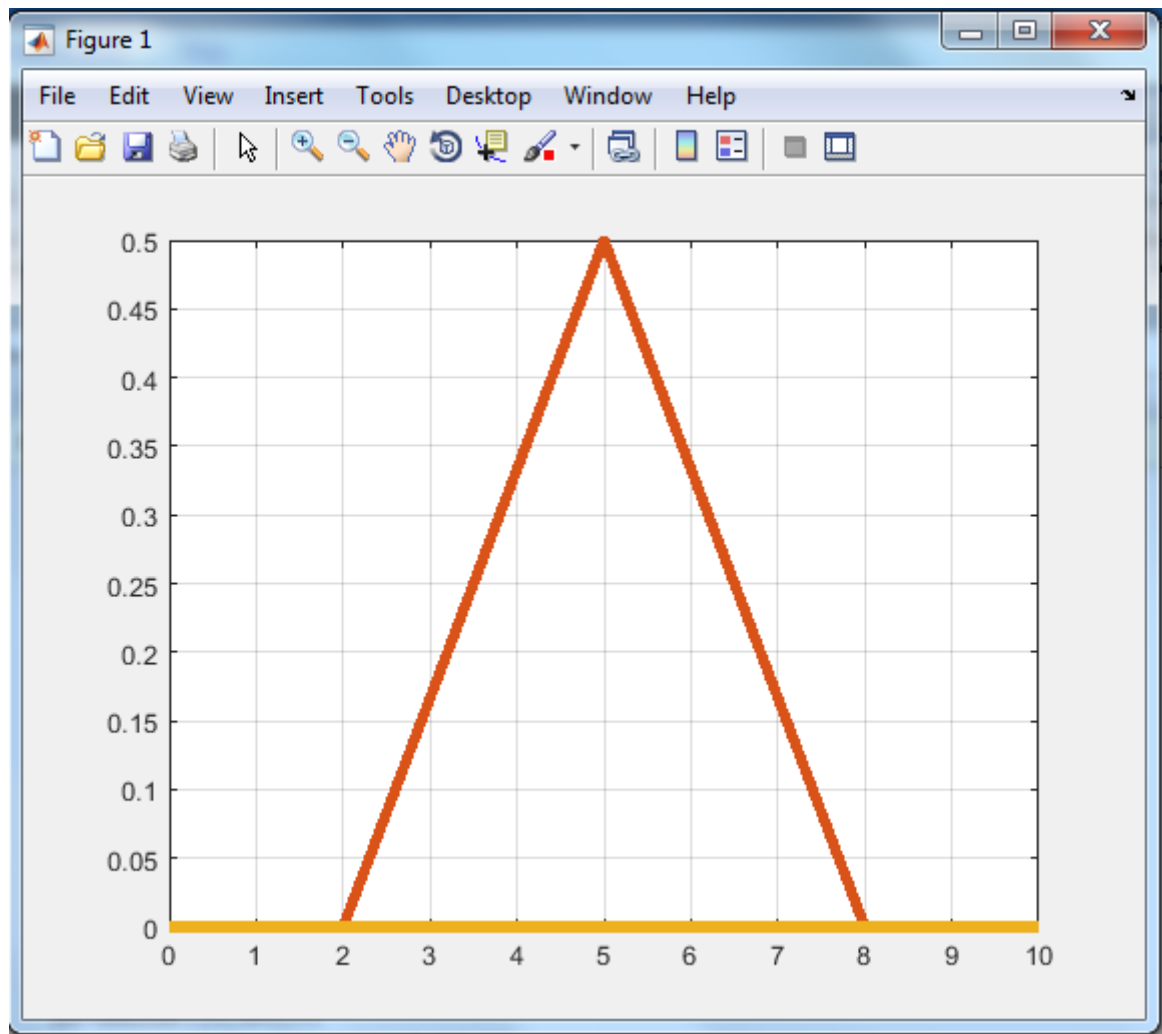


Рис.6. Модификация нечётких подмножеств для  $q$

График суперпозиции (максимума функций принадлежности) получим следующим образом. Для лингвистических переменных, имеющих обратную пропорциональную зависимость с выходом, отразим модифицированные графики слева направо.

```
>> A=[yN1;yN2;yN3;yf1;yf2;yf3;yt1;yt2;yt3;fliplr(yq1);fliplr(yq2);fliplr(yq3)];  
>> y=max(A);  
>> plot(x,y,'Linewidth',4)  
>> grid on
```

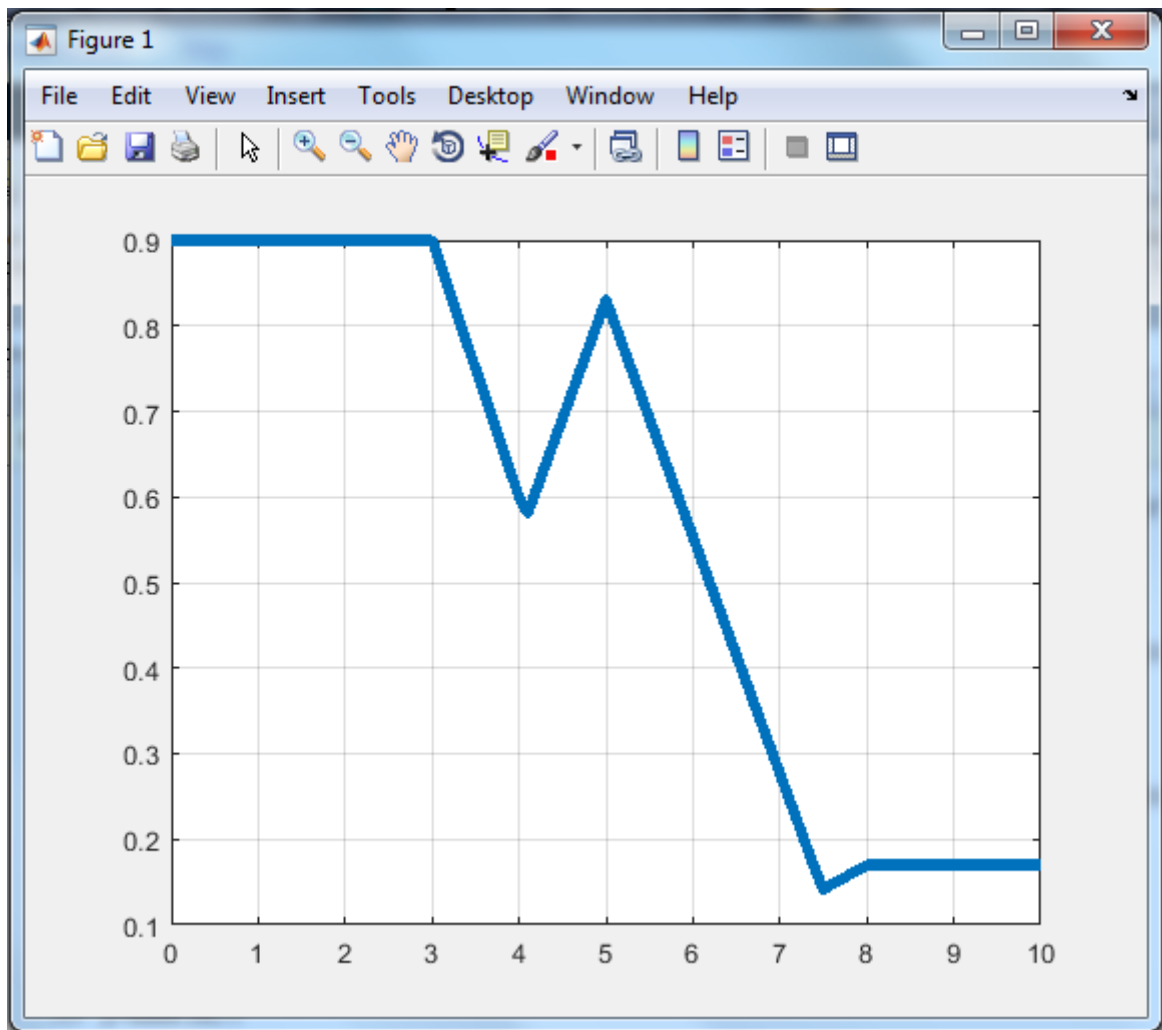


Рис.7. Суперпозиция модифицированных нечётких подмножеств для L

Теперь необходимо осуществить переход от суперпозиции множеств к скалярному значению. Скаляризацию произведем методом "центра тяжести":

$$c = \frac{\sum_{i=1}^k x_i m(x_i)}{\sum_{i=1}^k m(x_i)}, \quad (23)$$

где  $x_i$  - выходная переменная, имеющая  $k$  дискретных отсчётов,  $m(x_i)$  - значение результирующей функции принадлежности.

```

sum1=0;
sum2=0;
for (i=1:101)
sum1=sum1+x(i)*y(i);
sum2=sum2+y(i);
end
c=sum1/sum2
c =
    3.5615

```

Так как число линий – целое, округлим ответ до 4.

## Реализация в среде MATLAB

Откроем пакет Fuzzy Logic и добавим в систему недостающие переменные.

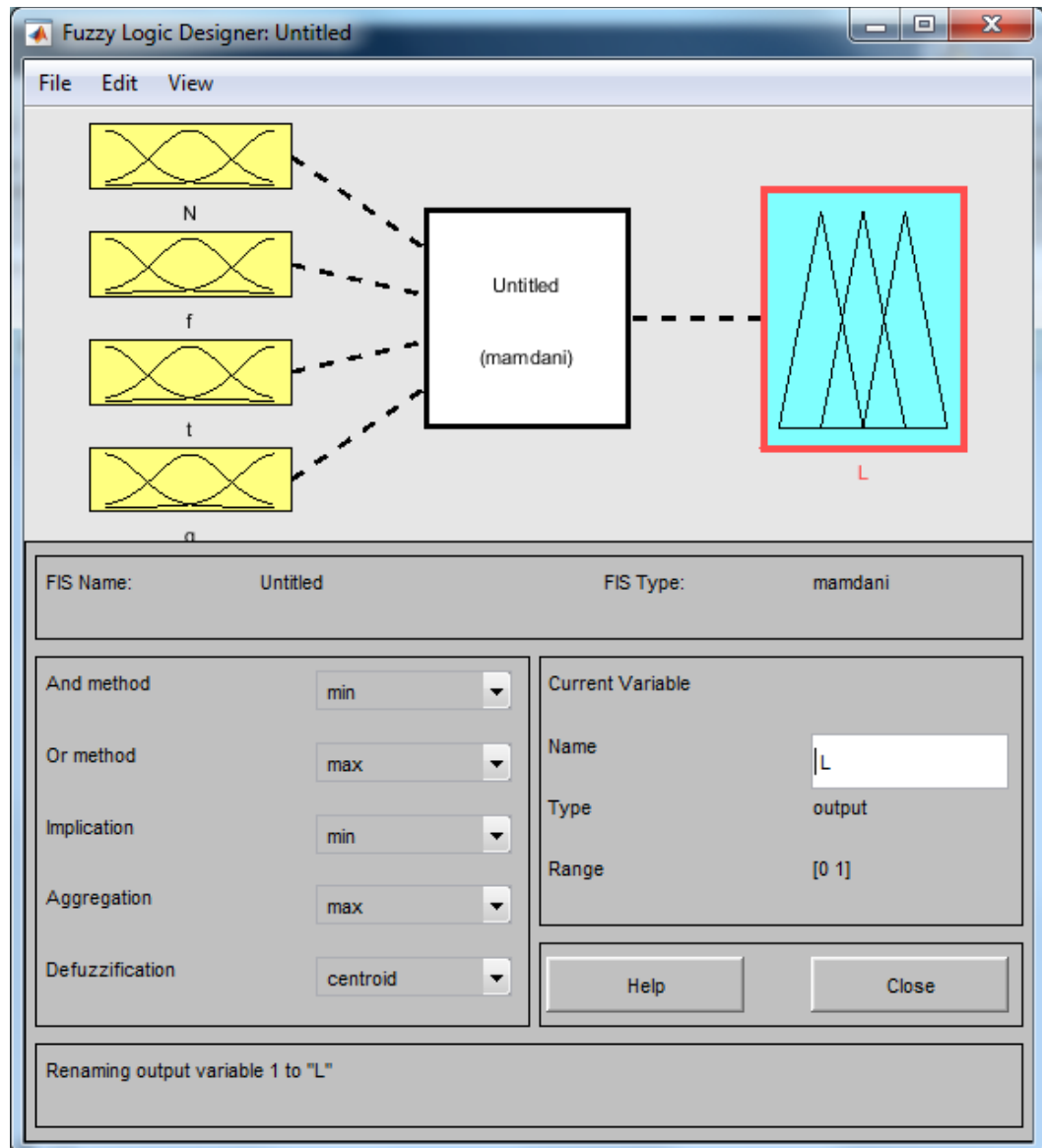


Рис. 8. Окно Fuzzy Logic Toolbox после добавления переменных

Зададим нечеткие функции принадлежности и диапазон значений для переменных.

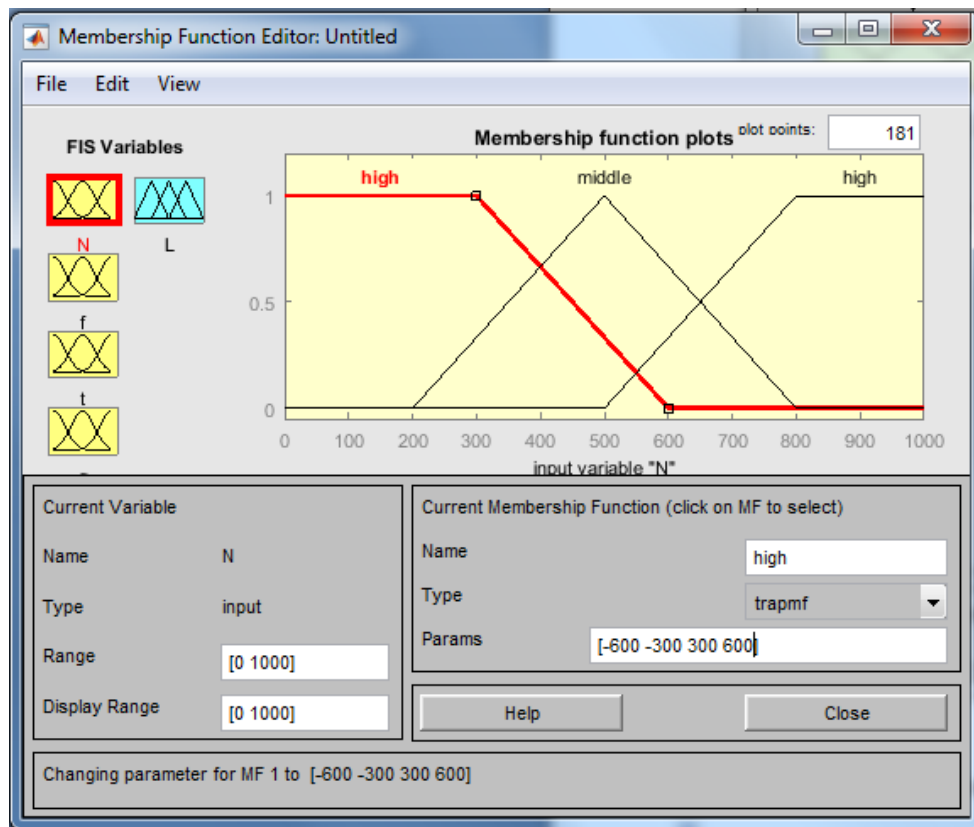


Рис. 9. Окно задания функций принадлежности и диапазона для переменных

Зададим правила вывода при помощи вкладки Edit→Rules.

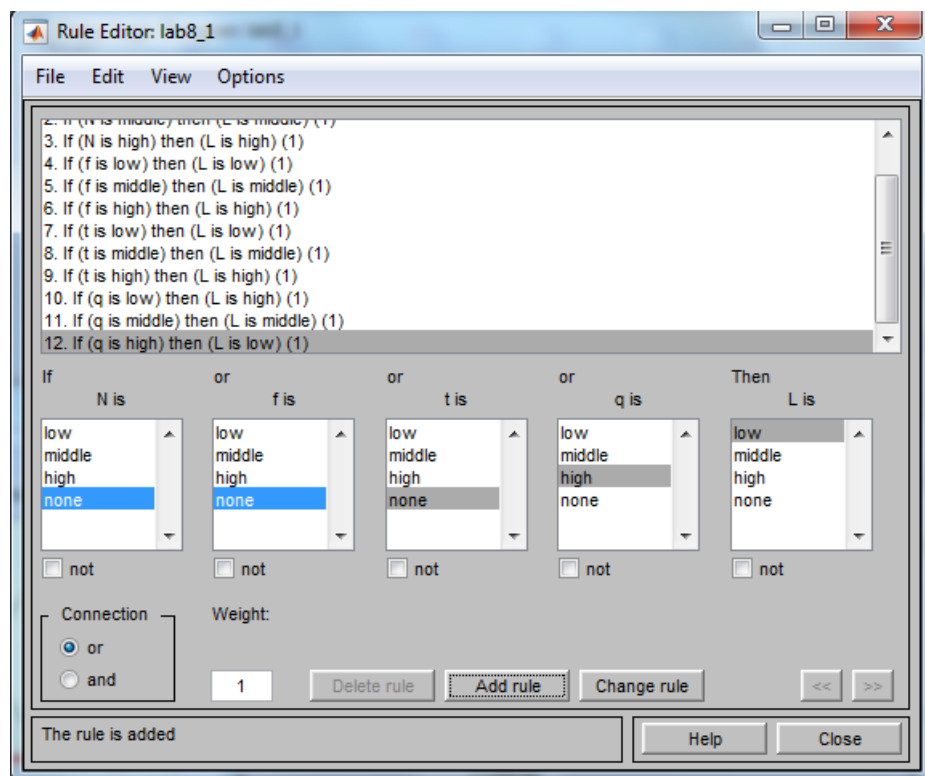


Рис. 10. Окно редактора после задания правил вывода

Получим отклик системы для наших входных значений.

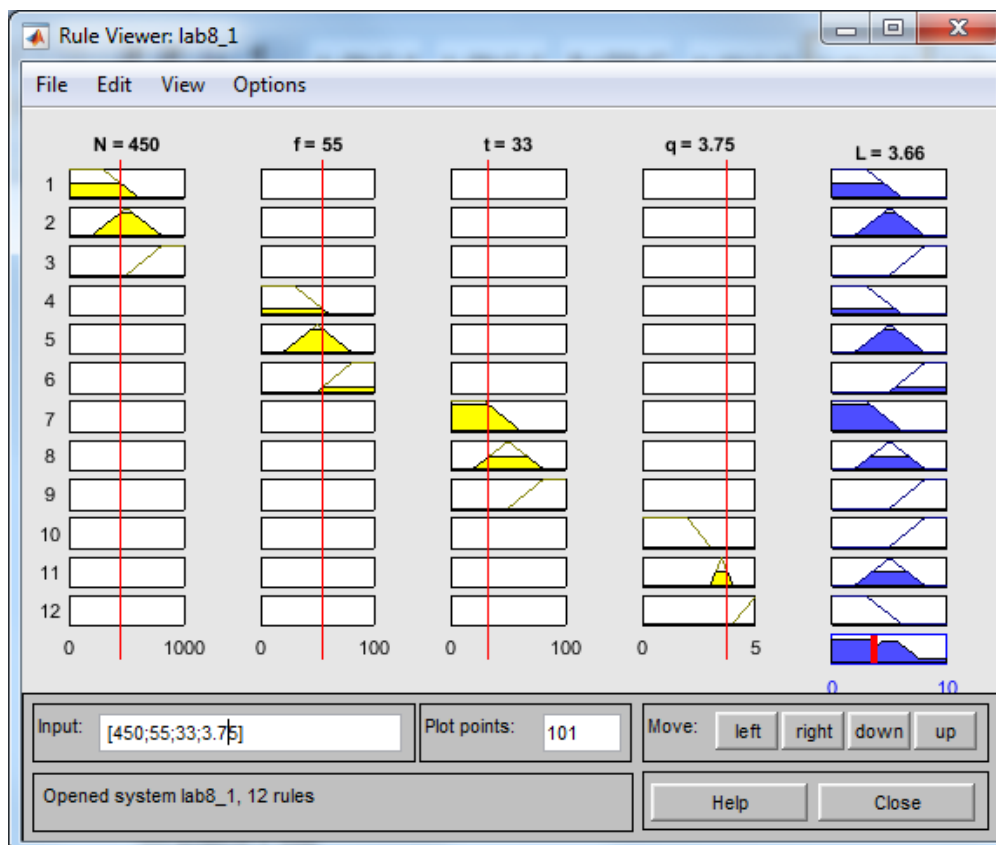


Рис. 11. Визуализация правил вывода

Как видно, ответ практически совпал с полученным вручную. Разница объясняется тем, что мы использовали разные методы фазификации.

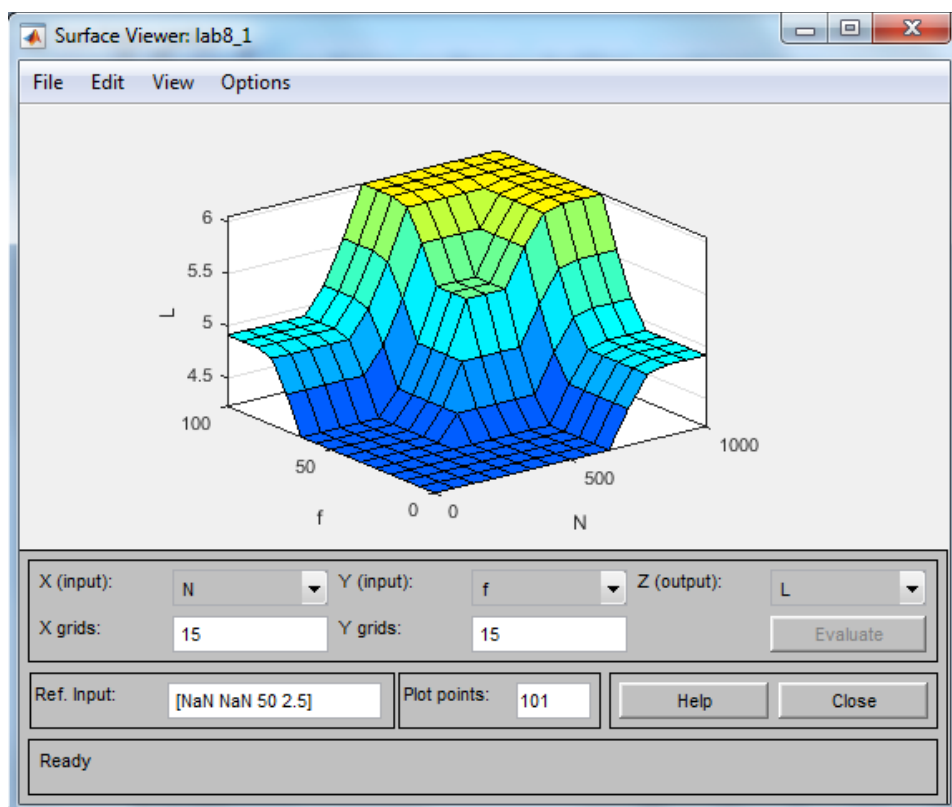


Рис. 12. Отклик системы

## Задание

Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи согласно варианту, проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).

### Список индивидуальных данных

1) Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи закупок (соотношения цены, качества, объема закупок и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).

2) Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи распределения нагрузок спортсмена (соотношение нагрузок, физического состояния, потребляемых калорий и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).

3) Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи управления транспортным средством (регулировка скорости с учетом передачи, погодных условий, интенсивности потока и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).

4) Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи управления транспортным средством (управление рулем, газом, тормозом при въезде в гараж), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).

5) Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи регулирования теплоснабжения (соотношение среднесуточной температуры, ветра, размера здания и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).

6) Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи регулирования реверсного движения на мосту (учитывать время, интенсивность потока, день недели и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).

7) Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи подбора специй для блюда (соотношение количества и остроты специй, рецептуры, предпочтений едока, объема пищи и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).

8) Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи подбора объема блюд (учитывать калорийность, вкусовые предпочтения, количество едоков и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).

9) Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи подачи электроэнергии в условиях экономии (учет времени суток, типа помещений, количества людей, типа оборудования и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).

10) Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи подбора интенсивности занятий (учитывать начальный уровень подготовки, объем учебного материала, количество человек в группе, необходимый уровень



усвоения и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).

11) Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи расчета потребления бензина (учитывать тип совершаемых маневров, уровень подготовки водителя, состояние автомобиля, тип автомобиля и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).

12) Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи регулирования системы орошения (учитывать время года, количество выпадающих осадков, вид орошаемой культуры и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).

13) Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи настройки аудиосистемы (мощность колонок, их количество, размер помещения, назначение установки и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).

14) Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи выбора дозы снотворного (количество препарата, действие препарата, восприимчивость к выбранному препарату, цель и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).

15) Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи планирования объема производства продукции (с учетом возможной прибыли, необходимых ресурсов, платежеспособности населения, рынка сбыта и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).

16) Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи регулирования кондиционера (учитывать его мощность, объем помещения, температуру окружающей среды, необходимую температуру в помещении и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).

17) Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи распределения нагрузки между компьютерами при использовании их в кластерах (учитывать характеристики компьютеров, их количество, количество параллельного кода, характеристики сети и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).

18) Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи выбора складского помещения (учитывать площадь склада, количество и размеры продукции, удаленность от места производства и точек реализации, свойства продукции и характеристики помещений и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).

19) Построить нечеткую базу знаний (использовать не менее 3 лингвистических переменных) для задачи выбора комплектующих для компьютера (учитывать цену, потребности пользователя, совместимость, сроки использования и т.д.), проверить ее на полноту и произвести нечеткий вывод для конкретных значений (выбрать случайным образом).

## Контрольные вопросы

1. Что такое базис нечётких правил?

2. В чём состоят основные особенности системы нечёткой логики с фазификатором и дефазификатором?
3. Чем различаются логический вывод Мамдани и Сугено?
4. Как отразить обратную зависимость между входными и выходной переменными при построении системы нечёткого вывода вручную?
5. Какие шаги следует предпринять для построения системы нечёткого вывода в MATLAB?