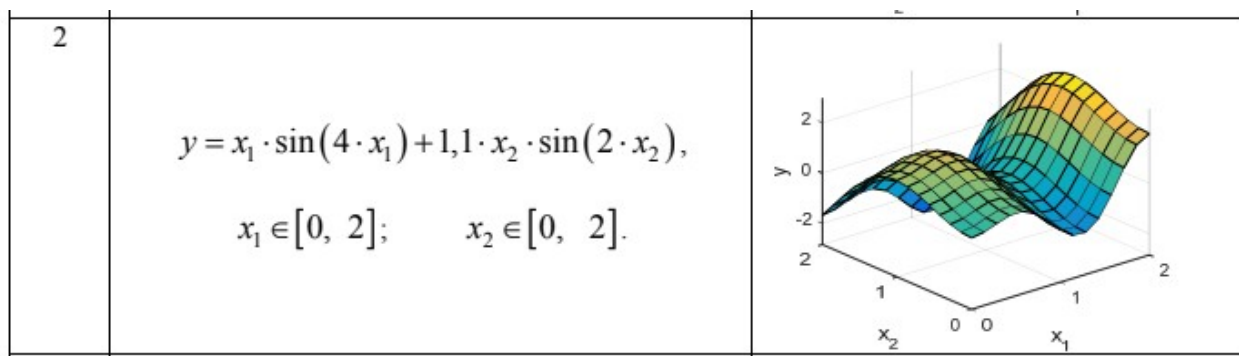


1. Борискин М.А.
2. 3341506/90401
3. 22.11.2020
4. Разработка систем нечёткого вывода в режиме командной строки с использованием пакета FuzzyLogicToolbox.
5. Изучение особенностей и приобретение навыков разработки систем нечёткого вывода в режиме командной строки с использованием пакета расширения Fuzzy Logic Toolbox в вычислительной среде MatLab.

Вариант 2:



6. m-программа синтеза системы нечёткого вывода в режиме командной строки согласно варианту задания:

```
% Разработка системы нечёткого вывода
% в режиме командной строки
(LR_5_KomSintFIS.m)
%=====
=====
clear all      % очистка памяти (leaving the workspace empty)
clc           % очистка командного окна (Clear Command Window)
%-----
-----
% Будем разрабатывать в режиме командной строки систему нечёткого
вывода,
% которая ранее (LR_1) была создана в среде GUI Fuzzy Logic
Toolbox
% и сохранена на диске под именем firstM.fis
% -----
% fuzzy
fis = readfis('2varM');          % загрузка нечёткой системы с диска
% fuzzy(fis)
% getfis(fis);
% showfis(fis);
```

```

%
-----
% РАЗРАБОТКА СИСТЕМЫ НЕЧЁТКОГО ВЫВОДА

fisK = newfis('firstMK');          % созд.новой сис.неч.выв. с именем
'firstMK'

fisK.input(1).name = 'x1';          % имя первой входной переменной
fisK.input(1).range = [0 2];        % границы первой входной переменной

fisK.input(1).mf(1).name = 'Низкий'; % имя перв.функц.прин.
перв.вх.перем.
fisK.input(1).mf(1).type = 'trimf'; % тип перв.функц.прин.
перв.вх.перем.
fisK.input(1).mf(1).params = [-0.8333 0 0.8333]; %      параметры
перв.функц.прин. перв.вх.перем.

fisK.input(1).mf(2).name = 'Средний'; % вторая функц. прин.
перв.вх.перем.
fisK.input(1).mf(2).type = 'trimf';
fisK.input(1).mf(2).params = [0.1667 1 1.833];

fisK.input(1).mf(3).name = 'Высокий'; % третья функц. прин.
перв.вх.перем.
fisK.input(1).mf(3).type = 'trimf';
fisK.input(1).mf(3).params = [1.167 2 2.833];

fisK.input(2).name = 'x2';          % вторая входная переменная
fisK.input(2).range = [0 2];

fisK.input(2).mf(1).name = 'Низкий';
fisK.input(2).mf(1).type = 'trimf';
fisK.input(2).mf(1).params = [-0.8333 0 0.8333];

fisK.input(2).mf(2).name = 'Средний';
fisK.input(2).mf(2).type = 'trimf';
fisK.input(2).mf(2).params = [0.1667 1 1.833];

fisK.input(2).mf(3).name = 'Высокий';
fisK.input(2).mf(3).type = 'trimf';
fisK.input(2).mf(3).params = [1.167 2 2.833];

fisK.output(1).name = 'y';          % выходная переменная
fisK.output(1).range = [-3 3];

fisK.output(1).mf(1).name = 'Низкий';
fisK.output(1).mf(1).type = 'gaussmf';
fisK.output(1).mf(1).params = [0.637 -3];

```

```

fisK.output(1).mf(2).name = 'Ниже-среднего';
fisK.output(1).mf(2).type = 'gaussmf';
fisK.output(1).mf(2).params = [0.637 -1.5];

fisK.output(1).mf(3).name = 'Средний';
fisK.output(1).mf(3).type = 'gaussmf';
fisK.output(1).mf(3).params = [0.637 2.776e-17];

fisK.output(1).mf(4).name = 'Выше-среднего';
fisK.output(1).mf(4).type = 'gaussmf';
fisK.output(1).mf(4).params = [0.637 1.5];

fisK.output(1).mf(5).name = 'Высокий';
fisK.output(1).mf(5).type = 'gaussmf';
fisK.output(1).mf(5).params = [0.637 3];

% Задаём базу правил
fisK.rule(1).antecedent = [1 1]; % усл.перв.прав.:
ном.функц.прин.вх.перем.
fisK.rule(1).connection = 1; % логич.опер.перв.прав.: 1-AND,
2-OR
fisK.rule(1).consequent = [3]; % закл.перв.прав.:
ном.фун.прин.вых.перем.
fisK.rule(1).weight = 1; % вес первого правила

fisK.rule(2).antecedent = [2 2]; % второе правило
fisK.rule(2).connection = 1;
fisK.rule(2).consequent = [3];
fisK.rule(2).weight = 1;

fisK.rule(3).antecedent = [1 3]; % 3 правило
fisK.rule(3).connection = 1;
fisK.rule(3).consequent = [2];
fisK.rule(3).weight = 1;

fisK.rule(4).antecedent = [3 1]; % 4 правило
fisK.rule(4).connection = 1;
fisK.rule(4).consequent = [4];
fisK.rule(4).weight = 1;

fisK.rule(5).antecedent = [3 3]; % 5 правило
fisK.rule(5).connection = 1;
fisK.rule(5).consequent = [3];
fisK.rule(5).weight = 1;

fisK.rule(6).antecedent = [2 3]; % 6 правило
fisK.rule(6).connection = 1;
fisK.rule(6).consequent = [1];
fisK.rule(6).weight = 1;

```

```

fisK.rule(7).antecedent = [3 2];    % 7 правило
fisK.rule(7).connection = 1;
fisK.rule(7).consequent = [5];
fisK.rule(7).weight = 1;

fisK.rule(8).antecedent = [1 2];    % 8 правило
fisK.rule(8).connection = 1;
fisK.rule(8).consequent = [4];
fisK.rule(8).weight = 1;

% -----
writefis(fisK, 'firstMK_2var')      % сохранение разработанной FIS на
дискете
% fuzzy(fisK)
%
-----

% СРАВНЕНИЕ ДВУХ СИСТЕМ НЕЧЁТКОГО ВЫВОДА (ПРОВЕРКА ИДЕНТИЧНОСТИ)

n = 15;          % количество точек дискретизации
x1 = linspace(0, 2, n);
x2 = linspace(0, 2, n);

yM = zeros(n, n);
yMK = zeros(n, n);
for i = 1:n
    yM(i,:) = evalfis([x1; ones(size(x1))*x2(i)], fis);
    yMK(i,:) = evalfis([x1; ones(size(x1))*x2(i)], fisK);
end

% -----
h1 = figure(1);
set(h1, 'Position', [13    553    524    407])
surf(x1, x2, yM)
axis([min(x1)  max(x1) ...
      min(x2)  max(x2) ...
      min(min(yM))  max(max(yM))]);
view(-40,30)
xlabel('x_1');  ylabel('x_2');  zlabel('y');
title('Исходная система нечёткого вывода Мамдани - firstM.fis')

h2 = figure(2);
set(h2, 'Position', [539    553    524    407])
surf(x1, x2, yMK)
axis([min(x1)  max(x1) ...
      min(x2)  max(x2) ...
      min(min(yMK))  max(max(yMK))]);
view(-40,30)
xlabel('x_1');  ylabel('x_2');  zlabel('y');
title('Созданная в режиме командной строки FIS - firstMK_2var.fis')

```

```

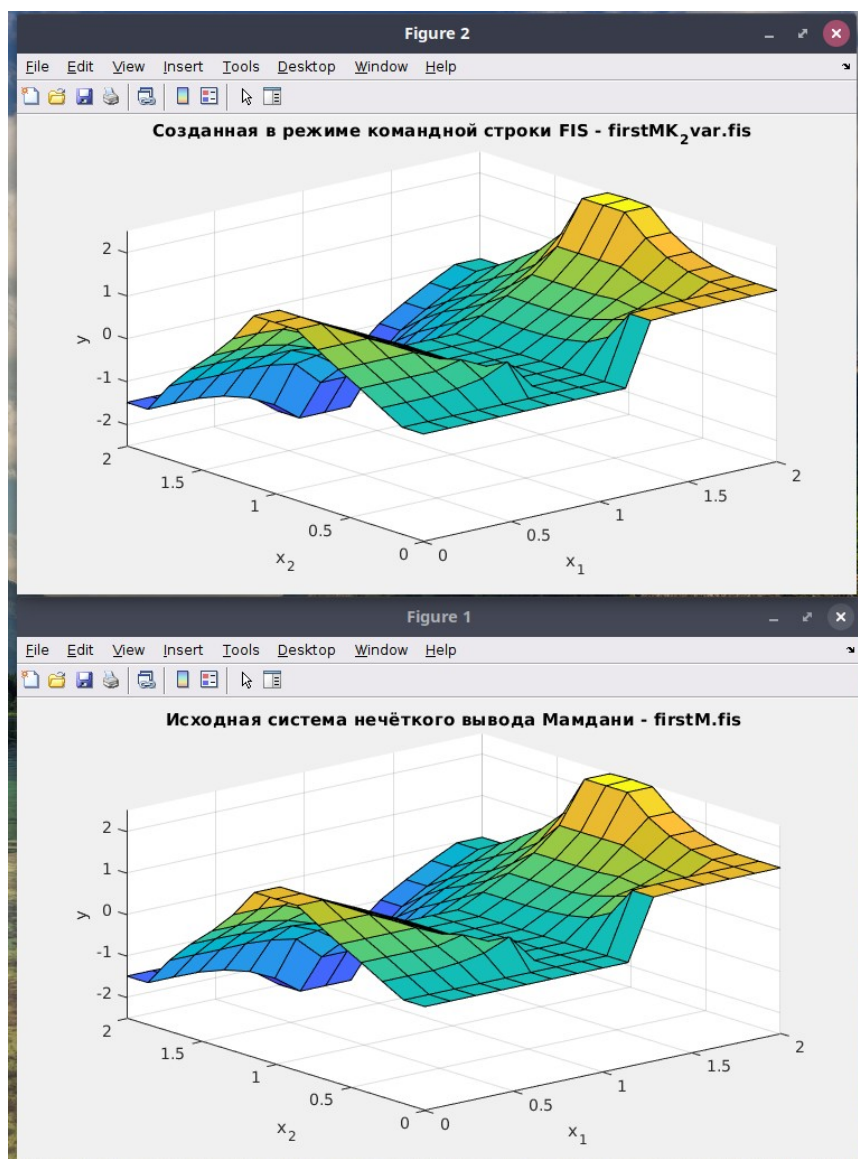
% -----
% Максимальное отклонение выходного значения FIS

maxER = max(max(yM-yMK));

disp(' ')
disp('Макс. откл. выхода синтезир. в режиме ком.стр. FIS от
исходной:')
disp([' maxER = ', num2str(maxER)])
disp(' ')
%
-----
-----

```

8. Графические зависимости выходных сигналов исходной и вновь разработанной систем нечёткого вывода от их входных сигналов в области определения. Величина максимального отклонения выходного сигнала разработанной системы от выхода исходной системы.



```
In LR 5 KomSintFIS (line 129)

Макс. откл. выхода синтезир. в режиме ком.стр. FIS от исходной:
maxER = 0

>>
```

## 8. Выводы по работе:

В результате сравнения получена ошибка: 0%. В данном случае правила для поверхности просты в составлении и задавались аналогично GUI-подходу.

На мой взгляд – удобнее работать в текстовом режиме, так как можно контролировать что откуда происходит и охватывать взглядом бОльший кусок кода, нежели при работе в интерактивном режиме. Однако приложение Fuzzy Logic Designer из-за автоматизации своих процессов не допускает ошибки, в отличие от работы в режиме командной строки. Таким образом, дебажить при использовании GUI намного легче. И сделать ошибку намного труднее. В этом плюс GUI. Также в случае более сложных систем, составление нечеткой системф без приложения отнимет гораздо больше времени. А точность сильно уменьшится.

Однако, главный вывод – составить нечеткую систему в Matlab можно и без использования Fuzzy Logic Designer.