Министерство науки и высшего образования Российской Федерации

Санкт-Петербургский политехнический университет Петра Великого

Институт машиностроения, материалов и транспорта

Высшая школа автоматизации и робототехники

**ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №5**

**Разработка систем нечёткого вывода в режиме командной строки  
с использования пакета Fuzzy Logic Toolbox**

по дисциплине «Нечёткие системы обработки информации  
в мехатронике и робототехнике»

Выполнил

студент гр. 3341506/10401 Паньков И. С.

Проверил

ассистент Абросимов Э. А.

«\_\_\_» \_\_\_\_\_\_\_ 2022 г.

Санкт-Петербург

2022

# Цель работы

Цель работы — приобрести навыки разработки систем нечёткого вывода в режиме командной строки среды MATLAB с использованием пакета расширения Fuzzy Logic Toolbox.

# Задание

1. Ознакомиться с порядком разработки систем нечёткого вывода в режиме командной строки среды MATLAB по учебно-методическому пособию.
2. При помощи пакета расширения Fuzzy Logic Toolbox разработать в режиме командной строки систему нечёткого вывода типа Мамдани (аналогичную той, которая была получена в лабораторной работе №1) для аппроксимации нелинейной зависимости, описываемой следующей функцией:



1. Для проверки идентичности исходной и вновь разработанной систем нечёткого вывода построить поверхности «входы – выход» данных систем. Также определить максимальное отклонение выходных сигналов вновь разработанной и исходной систем.
2. Сделать выводы из проделанной работы, в которых отразить:

* область возможного применения пакета Fuzzy Logic Toolbox в режиме командной строки для решения прикладных задач;
* сравнительную оценку трудозатрат при разработке систем нечёткого вывода с использованием пакета Fuzzy Logic Toolbox в интерактивном и командном режимах.

1. Подготовить отчёт по лабораторной работе.

# Ход работы

Исходная нелинейная зависимость описывает поверхность, график которой представлен на рисунке Рисунок 1.



Рисунок — Поверхность исходной нелинейной зависимости

Для аппроксимации зависимости ранее была разработана система нечёткого вывода типа Мамдани. Поскольку поверхность достаточно сложна для описания, было решено создавать систему сразу с пятью термами как входных, так и выходных переменных: negative-big, negative-middle, zero, positive-middle и positive-big. В результате исследования зависимости точности аппроксимации от типа функции принадлежности было решено использовать гауссовы функции принадлежности. Графики функций принадлежности термов входных и выходных переменных ,  и  представлен на рисунке Рисунок 2.

а)

б)

в)



Рисунок — Функции принадлежности термов входных  
и выходных переменных: а) —, б) — , в) — 

Для системы была определена следующая база правил нечётких продукций:

1. Если  zero и  zero, то  positive-big;
2. Если  negative-middle и  negative-middle, то  zero;
3. Если  positive-middle и  positive-middle, то  zero;
4. Если  negative-big и  positive-big, то  zero;
5. Если  positive-big и  negative-big, то  zero;
6. Если  negative-big и  negative-big, то  negative-middle;
7. Если  positive-big и  positive-big, то  negative-middle;
8. Если  negative-middle и  positive-middle, то  negative-big;
9. Если  positive-middle и  negative-middle, то  negative-big;
10. Если  zero и  negative-big, то  negative-big;
11. Если  zero и  positive-big, то  negative-big;
12. Если  negative-big и  zero, то  negative-big;
13. Если  positive-big и  zero, то  negative-big.

В итоге была получена система нечёткого вывода для аппроксимации исходной зависимости со среднеквадратичной ошибкой , поверхность «входы – выход» которой представлена на рисунке Рисунок 3.



Рисунок — Поверхность «входы – выход» системы нечёткого вывода

Создадим систему нечёткого вывода, аналогичную уже созданной, используя режим командной строки среды MATLAB. Листинг скрипта на языке MATLAB для загрузки исходной системы нечёткого вывода, создания новой системы нечёткого вывода, а также тестирования и сравнения этих систем представлен в листинге Листинг 1.

Листинг — Скрипт для загрузки исходной системы нечёткого вывода и создания новой  
системы нечёткого вывода, их тестирования и сравнения

|  |  |
| --- | --- |
| 001  002  003  004  005  006  007  008  009  010  011  012  013  014  015  016  017  018  019  020  021  022  023  024  025  026  027  028  029  030  031  032  033  034  035  036  037  038  039  040  041  042  043  044  045  046  047  048  049  050  051 | clc; clear; close all;  n = 25;  x1min = 0;  x1max = pi;  x2min = -1;  x2max = 1;  ymin = 0;  ymax = 1;  x1 = linspace(x1min, x1max, n);  x2 = linspace(x2min, x2max, n);  x = reshape(cat(3, repmat(x1, length(x2), 1)', ...  repmat(x2, length(x1), 1)), [], 2, 1);  y = sin(x1 - 2 \* x2').^2 .\* exp(-abs(x2'));  print\_surface\_plot(x1, x2, y, 'Original Function', 'original\_function.emf');  name = {'negative-big','negative-middle','zero','positive-middle','positive-big'};  count = length(name);  fis1 = readfis('../model/mamdani\_gaussmf\_5in\_gaussmf\_5out.fis');  fis2 = mamfis('Name', 'mamdani\_gaussmf\_5in\_gaussmf\_5out\_custom.fis');  input1 = fisvar([x1min, x1max], 'Name', 'x1');  dx1 = x1max - x1min;  for k = 1 : count  params = [dx1 / (2 \* count), x1min + dx1 \* (k - 1) / (count - 1)];  gaussmf = fismf('gaussmf', params, 'Name', name{k});  input1.membershipFunctions(k) = gaussmf;  end  input2 = fisvar([x2min, x2max], 'Name', 'x2');  dx2 = x2max - x2min;  for k = 1 : length(name)  params = [dx2 / (2 \* count), x2min + dx2 \* (k - 1) / (count - 1)];  gaussmf = fismf('gaussmf', params, 'Name', name{k});  input2.membershipFunctions(k) = gaussmf;  end  output = fisvar([ymin, ymax], 'Name', 'y');  dy = ymax - ymin;  for k = 1 : length(name)  params = [dy / (2 \* count), ymin + dy \* (k - 1) / (count - 1)];  gaussmf = fismf('gaussmf', params, 'Name', name{k});  output.membershipFunctions(k) = gaussmf;  end |

Продолжение листинга Листинг 1

|  |  |
| --- | --- |
| 052  053  054  055  056  057  058  059  060  061  062  063  064  065  066  067  068  069  070  071  072  073  074  075  076  077  078  079  080 | fis2.rules = [  fisrule('x1 == zero & x2 == zero => y = positive-big' );  fisrule('x1 == negative-middle & x2 == negative-middle => y = zero' );  fisrule('x1 == positive-middle & x2 == positive-middle => y = zero' );  fisrule('x1 == negative-big & x2 == positive-big => y = zero' );  fisrule('x1 == positive-big & x2 == negative-big => y = zero' );  fisrule('x1 == negative-big & x2 == negative-big => y = negative-middle');  fisrule('x1 == positive-big & x2 == positive-big => y = negative-middle');  fisrule('x1 == negative-middle & x2 == positive-middle => y = negative-big' );  fisrule('x1 == positive-middle & x2 == negative-middle => y = negative-big' );  fisrule('x1 == zero & x2 == negative-big => y = negative-big' );  fisrule('x1 == zero & x2 == positive-big => y = negative-big' );  fisrule('x1 == negative-big & x2 == zero => y = negative-big' );  fisrule('x1 == positive-big & x2 == zero => y = negative-big' );  ];  y1 = reshape(evalfis(fis1, x), length(x1), length(x2))';  rmse1 = sqrt(sum(sum((y - y1).^2)) / numel(y1));  print\_surface\_plot(x1, x2, y1, 'Default Mamdani Gauss MF Surface', ...  'mamdani\_gauss\_5in\_gauss\_5out\_surface\_default.emf', rmse1);  y2 = reshape(evalfis(fis2, x), length(x1), length(x2))';  rmse2 = sqrt(sum(sum((y - y2).^2)) / numel(y2));  print\_surface\_plot(x1, x2, y2, 'Custom Mamdani Gauss MF Surface', ...  'mamdani\_gauss\_5in\_gauss\_5out\_surface\_custom.emf', rmse2);  writefis(fis2, strcat('../model/', fis2.name));  fprintf('dy\_max = %.4g\n', max(max(abs(y2 - y1)))); |

Поверхности «входы – выход» исходной и вновь созданной систем нечёткого вывода представлены на рисунке Рисунок 4.



а)

б)

Рисунок — Поверхности «входы – выход» систем нечёткого вывода:  
а) — исходная система, б) — вновь созданная система

Как видно по полученным поверхностям и среднеквадратичной ошибке аппроксимации, системы нечёткого вывода идентичны. При этом абсолютное отклонение между выходами систем составило , что сравнимо с погрешностью чисел с плавающей запятой двойной точности ().

# Вывод

Пакет Fuzzy Logic Toolbox в режиме командной строки среды MATLAB предоставляется абсолютно те же возможности по созданию систем нечёткого вывода, что и в интерактивном режиме. Пусть в режиме командной строки пользователь лишён инструментов для визуализации базы правил и зависимости между входами и выходами системы, ему остаётся доступен широкий выбор между свободными функциями и классами пакета расширения для создания новых систем и редактирования существующих.

Более того, путём написания скриптов пользователем могут быть нивелированы некоторые рутинные действия при создании систем нечёткого вывода, например добавление и/или изменение функций принадлежности термов входных и выходных переменных. При этом редактирование базы правил нечётких продукций и вовсе представляется более удобным именно из режима командной строки в сравнении с инструментами, доступными в интерактивном режиме.

Таким образом, вопрос удобства и оценки трудозатрат остаётся дискуссионным при сравнении режимов работы с пакетом Fuzzy Logic Toolbox. При это возможен некоторый компромисс при разработке систем нечёткого вывода, когда «каркас» системы создаётся с помощью скрипта в режиме командной строки, а более тонкая настройка производится уже в интерактивном режиме.

*23:39*

*17 ноября 2022 г.*