

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КРИВОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ**

З В І Т

**Лабораторна робота №5
з дисципліни
«Сучасні методи та моделі
інтелектуальних систем керування»**

Виконавець:

аспірант групи АКІТР-23-1а

Косей М.П.

Керівник:

викладач

Тиханський М. П.

2024

Лабораторна робота № 5

Тема. Порівняльне дослідження роботи алгоритмів Мамдані та Сугено в однотипних додатках

Мета: освоїти принципи роботи алгоритмів Мамдані та Сугено. Навчитися задавати властивості системи залежно від поставленої задачі та алгоритму розв'язання.

Короткі теоретичні відомості

Алгоритми нечіткого виведення різняться, головним чином, видом використовуваних правил, логічних операцій і різновидом методу дефазифікації. Розроблені моделі нечіткого виведення Мамдані, Сугено, Ларсена, Цукамото. При розгляді алгоритмів для спрощення припустимо, що базу знань організують два нечітких правила вигляду:

P_1 : якщо $x \in A_1$ та $y \in B_1$, то $z \in C_1$,

P_2 : якщо $x \in A_2$ та $y \in B_2$, то $z \in C_2$,

де x і y – імена вхідних змінних, z – ім'я змінної виведення, $A_1, A_2, B_1, B_2, C_1, C_2$ – деякі задані функції належності, при цьому чітке значення z_0 необхідно визначити на основі наведеної інформації та чітких значень x_0 і y_0 .

Алгоритм Мамдані (Mamdani)

Алгоритм Мамдані є одним з перших, який знайшов застосування в системах нечіткого виведення. Він був запропонований 1975 р. англійським математиком Е. Мамдані (Ebrahim Mamdani) як метод для керування паровим двигуном. Формально *алгоритм Мамдані* може бути визначений таким чином.

1. Процедура фазифікації: визначаються ступені істинності, тобто значення функцій належності для лівих частин кожного правила (передумов): $A_1(x_0), A_2(x_0), B_1(y_0), B_2(y_0)$.

2. Нечітке виведення: знаходяться рівні відтинання для передумов кожного з правил з використанням операції мінімуму:

$$\alpha_1 = A_1(x_0) \wedge B_1(y_0),$$

$$\alpha_2 = A_2(x_0) \wedge B_2(y_0),$$

де через « \wedge » позначена операція логічного мінімуму (min), потім знаходяться «зрізані» функції належності

$$C_1'(z) = (\alpha_1 \wedge C_1(z)),$$

$$C_2'(z) = (\alpha_2 \wedge C_2(z)).$$

3. Композиція: з використанням операції максимуму (max, позначення: « \vee ») виконується об'єднання знайдених зрізаних функцій,

що приводить до отримання підсумкової нечіткої підмножини для змінної виходу з функцією належності

$$\mu_{\Sigma}(z) = C(z) = C_1'(z) \vee C_2'(z) = (\alpha_1 \wedge C_1(z)) \vee (\alpha_2 \wedge C_2(z)).$$

4. Приведення до чіткості (для знаходження z_0) проводиться, наприклад, центроїдним методом (як x – координата центра ваги функції належності підсумкової нечіткої підмножини для змінної виходу):

$$z_0 = \frac{\int_{\Omega} z \cdot \mu_{\Sigma}(z) dz}{\int_{\Omega} \mu_{\Sigma}(z) dz}$$

Алгоритм ілюструється рис. 2.1:

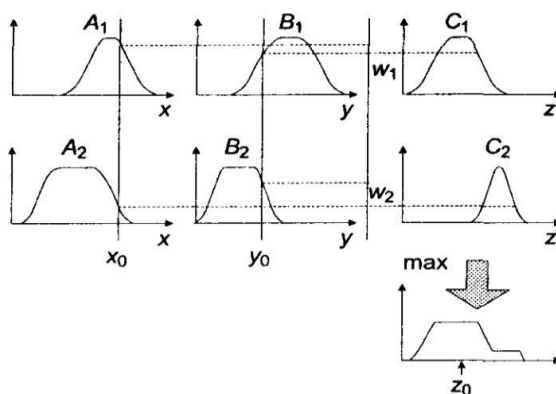


Рисунок 2.1 – Графічна реалізація

Алгоритм Сугено (Sugeno)

Формально алгоритм Сугено, запропонований Сугено і Такагі, може бути визначений таким чином.

1. Перший етап – як в алгоритмі Мамдані.
2. На другому етапі знаходяться $\alpha_1 = A_1(x_0) \wedge B_1(y_0)$, $\alpha_2 = A_2(x_0) \wedge B_2(y_0)$ та індивідуальні виходи правил:

$$z^*_{*1} = a_1 x_0 + b_1 y_0,$$

$$z^*_{*2} = a_2 x_0 + b_2 y_0,$$

3. На третьому етапі визначається чітке значення змінної виведення:

$$z_0 = \frac{\alpha_1 z^*_{*1} + \alpha_2 z^*_{*2}}{\alpha_1 + \alpha_2}$$

Алгоритм ілюструється на рисунку 2.2:

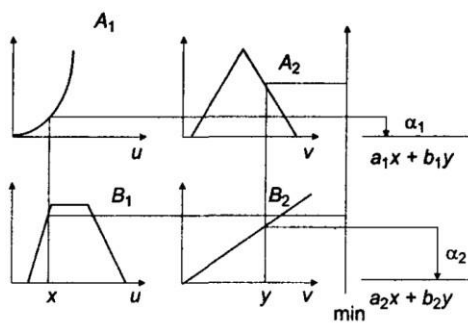
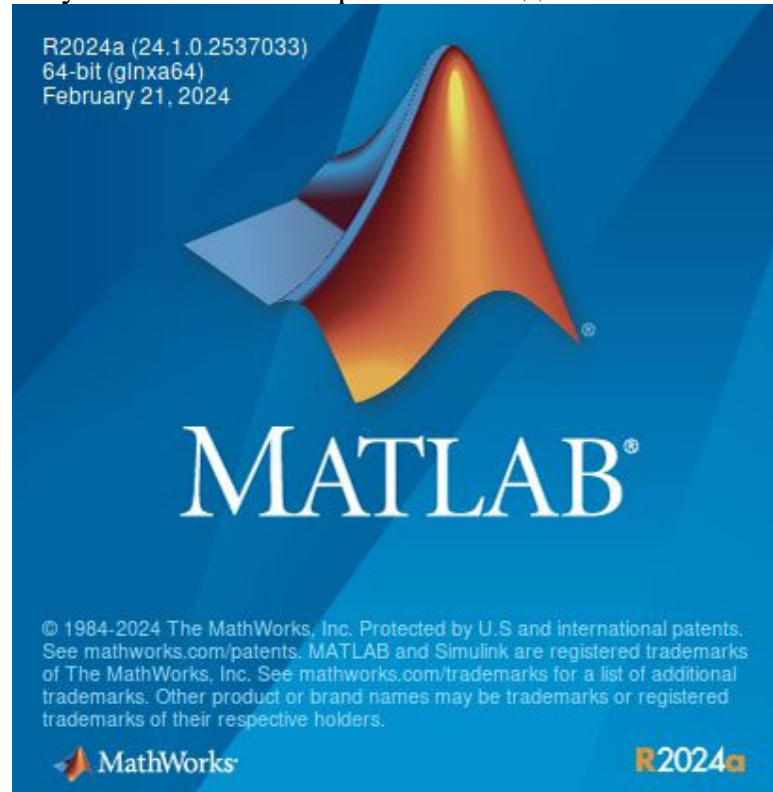


Рисунок 2.2 – Графічна реалізація алгоритму Сугено

2) Практична частина

Використовуємо **MATLAB** версія R2024a для Linux.



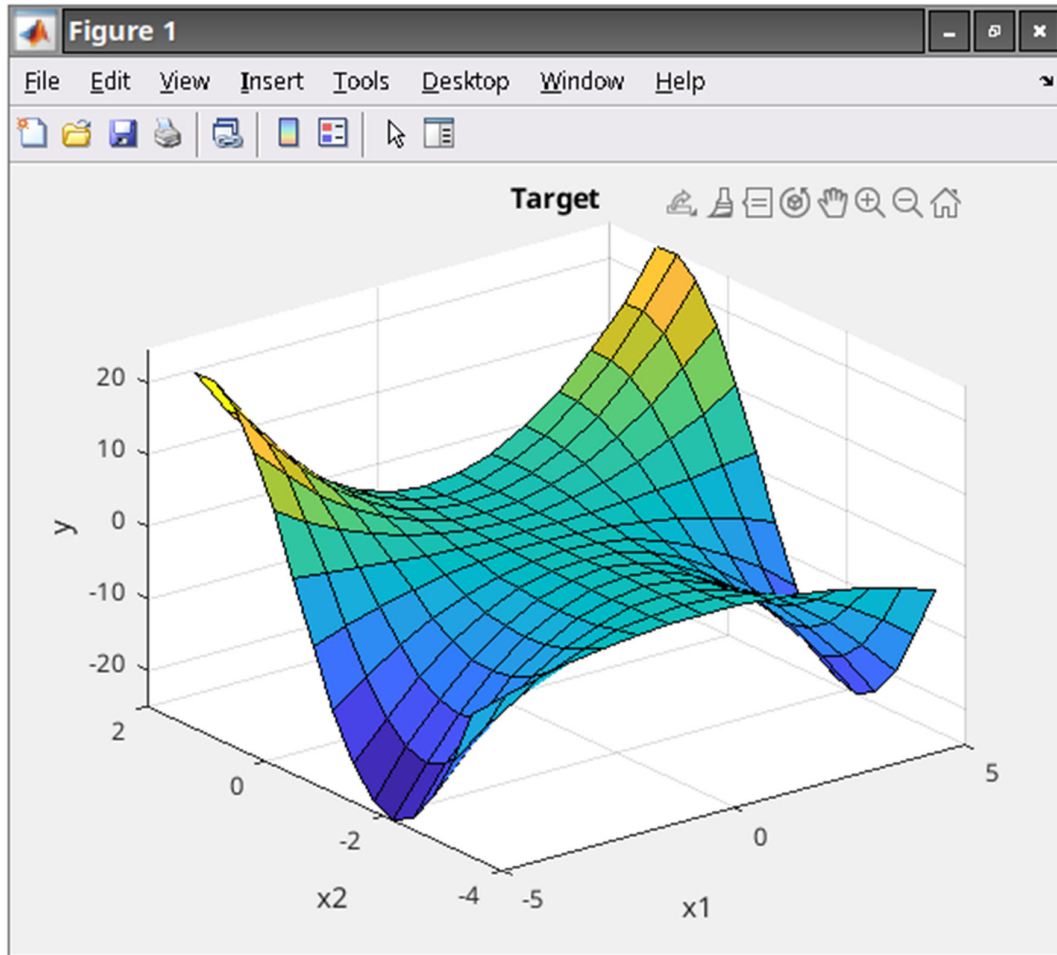
Порівняємо алгоритми Мамдані і Сугено на прикладі створення системи нечіткого логічного виведення, що моделює функцію $y = x_1^2 \cdot \cos(x_2 - 1)$ в межах $x_1 \in [-5, 5]$ і $x_2 \in [-3.5, 1.2]$.

Проектування системи нечіткого логічного виведення необхідно проводити на основі графічного зображення функції.

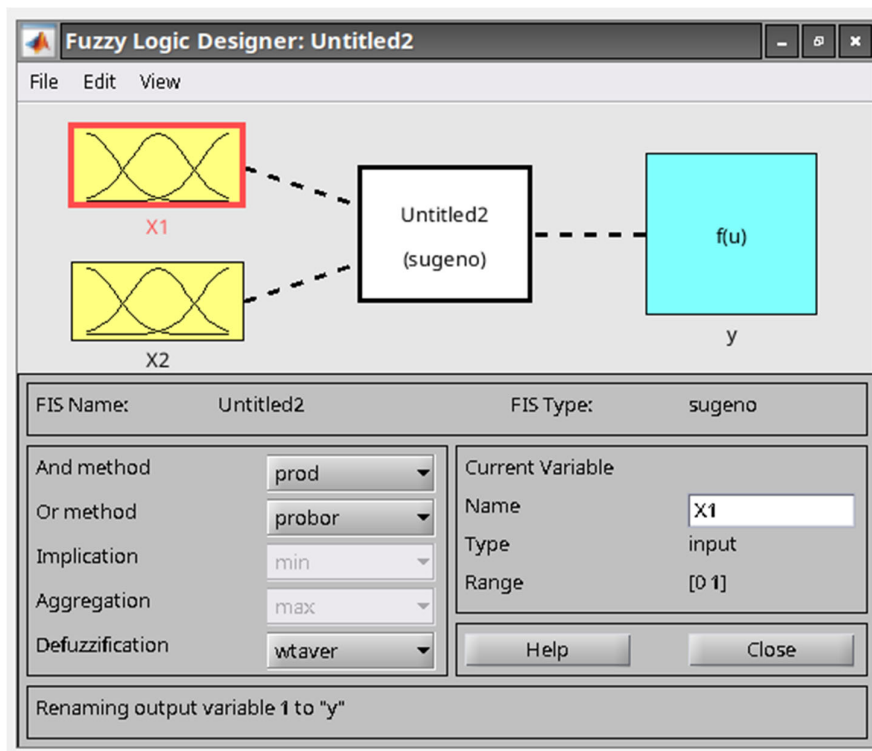
Складемо наступну програму, прописавши її в m-файлі:

```
Editor - /home/max11mus/Documents/LABS/LAB4/Function.m
Function.m
1 % Побудова графіка функції  $y = x_1^2 \cdot \cos(x_2 - 1)$  в межах  $x_1 \in [-5, 5]$ 
2 % і  $x_2 \in [-3.5, 1.2]$ .
3 n = 15; % кількість точок
4 x1 = linspace(-5, 5, n); % задання параметрів змінної x1
5 x2 = linspace(-3.5, 1.2, n); % задання параметрів змінної x2
6 y = zeros(n, n); % формування нульового масиву
7 % розміром n x n для вихідної змінної
8 for j = 1:n
9     y(j, :) = x1.^2 .* cos(x2(j) - 1);
10 end
11 surf(x1, x2, y); % зображення поверхні функції
12 xlabel('x1');
13 ylabel('x2');
14 zlabel('y');
15 title('Target');
16
```

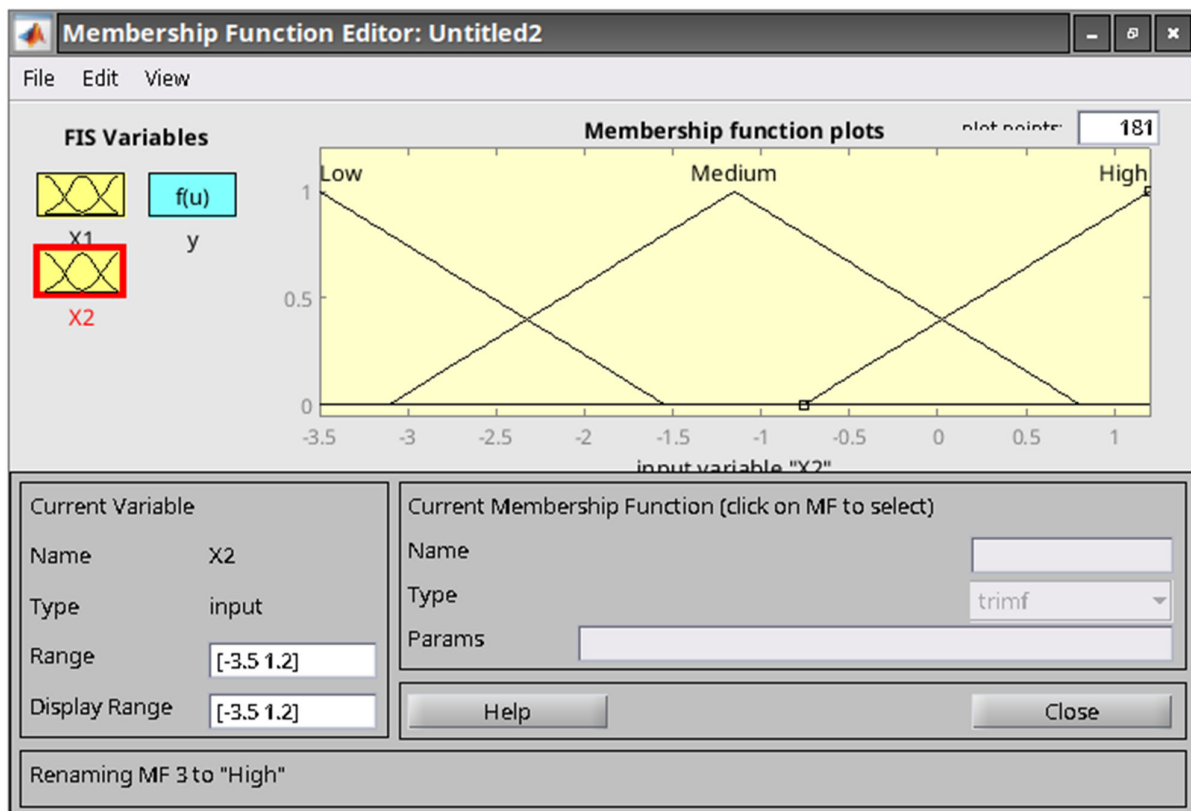
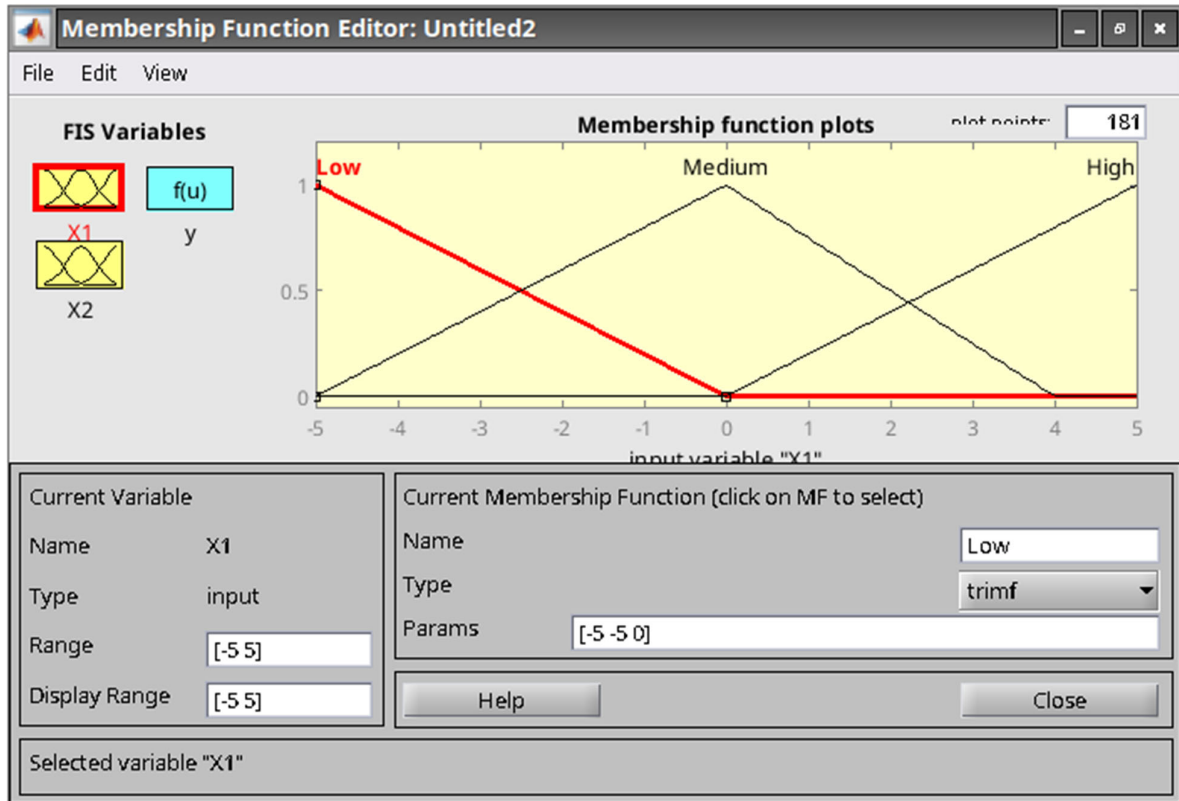
Створити еталонне тривимірне зображення заданої функції створення.



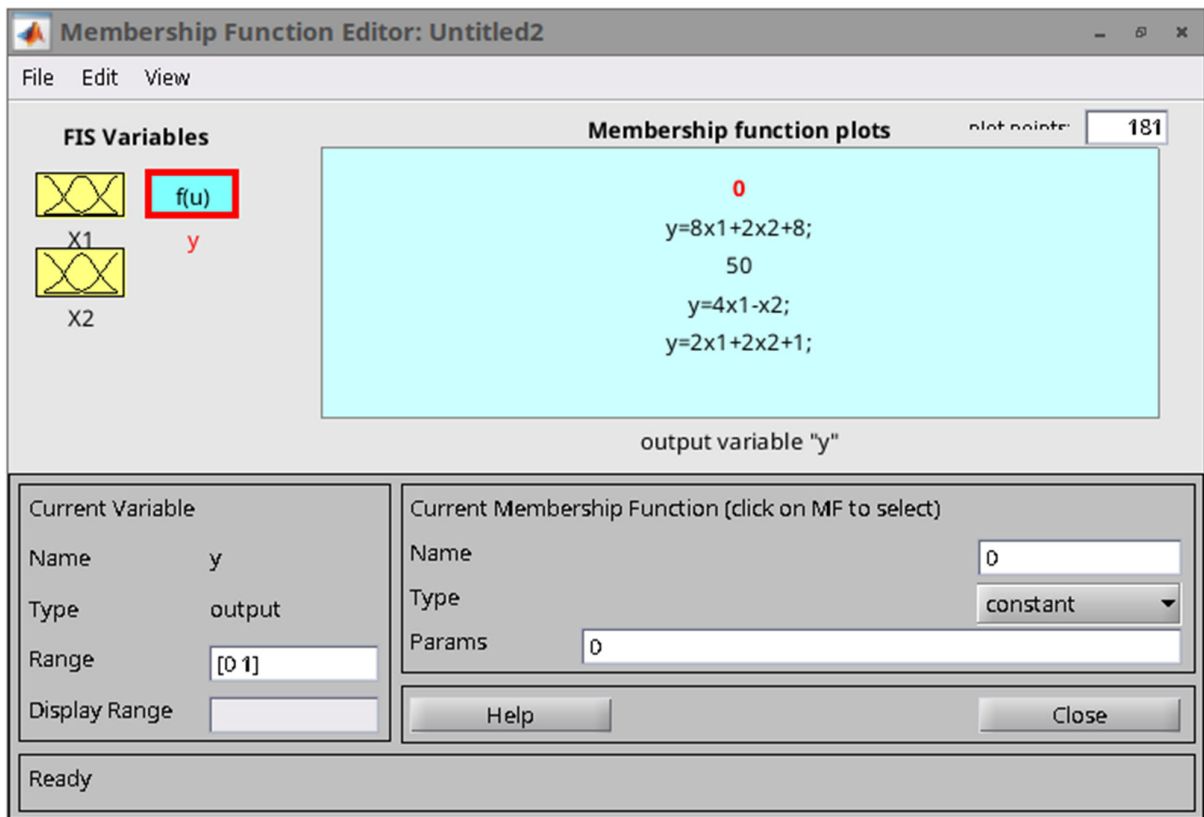
Реалізація першої системи Сугено:



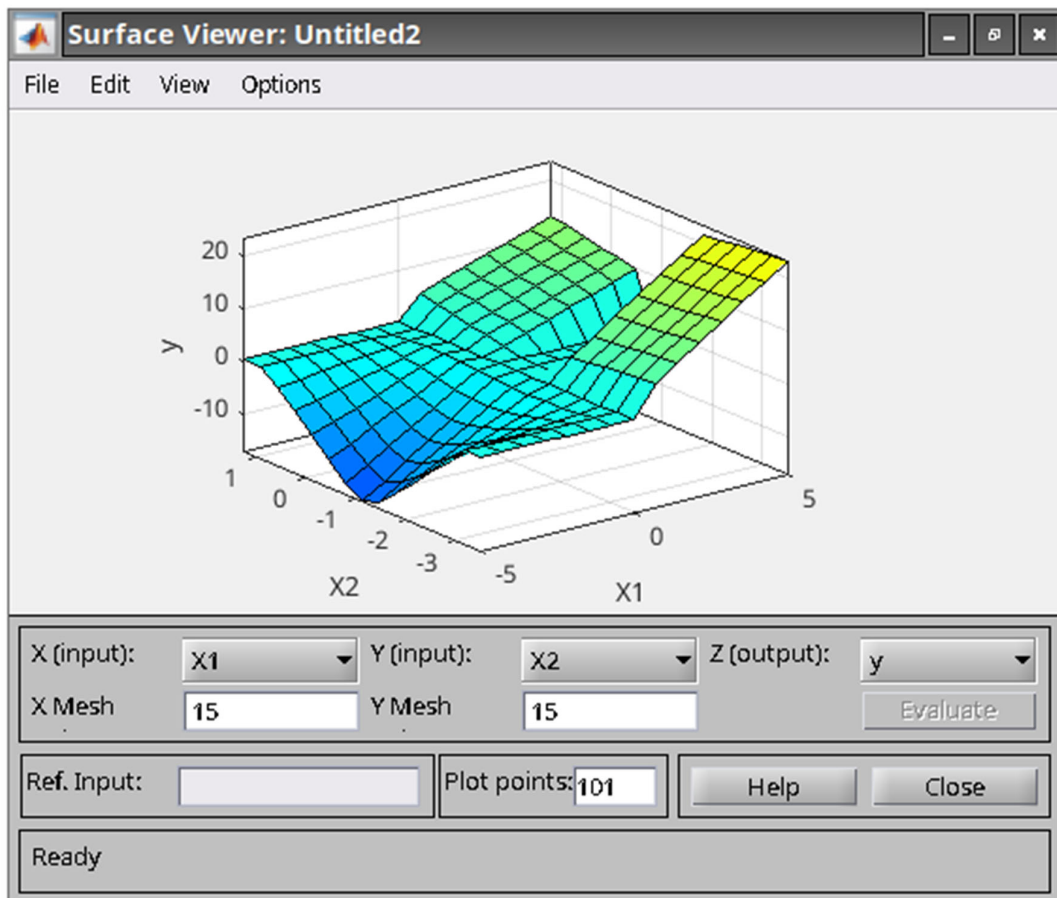
Налаштовуємо вхідні змінні



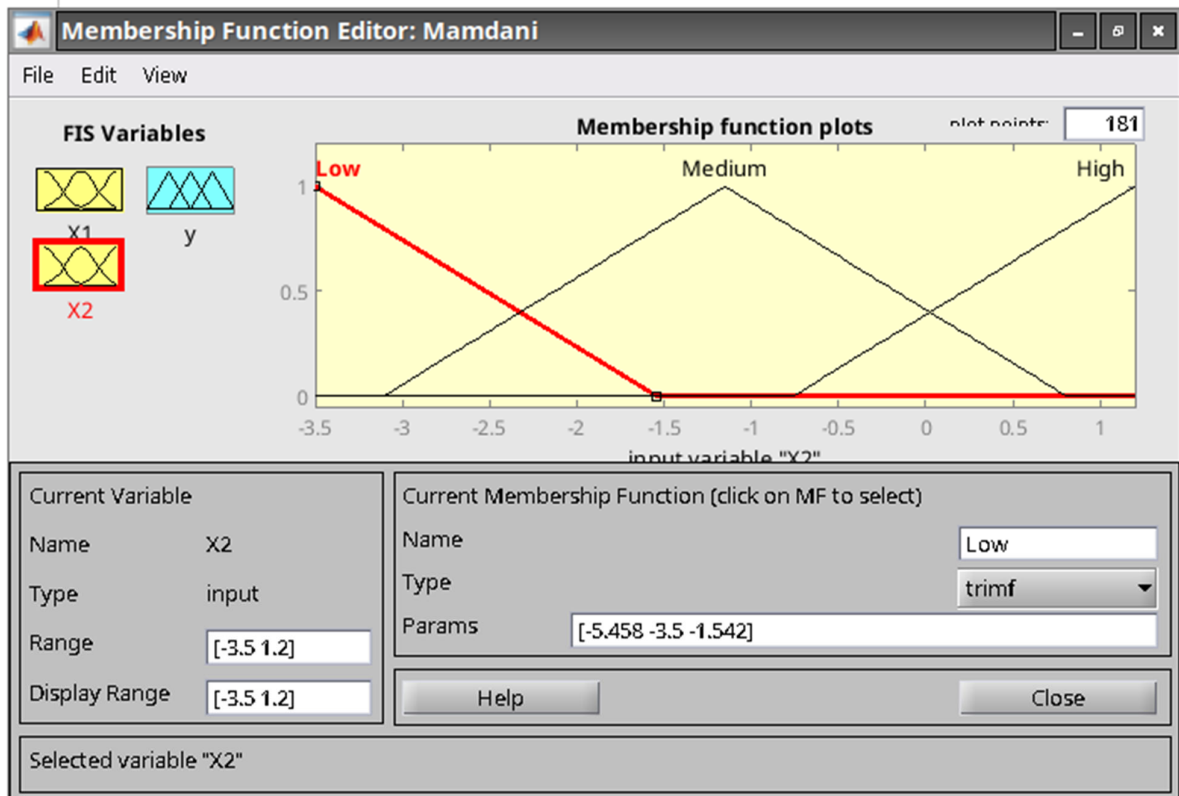
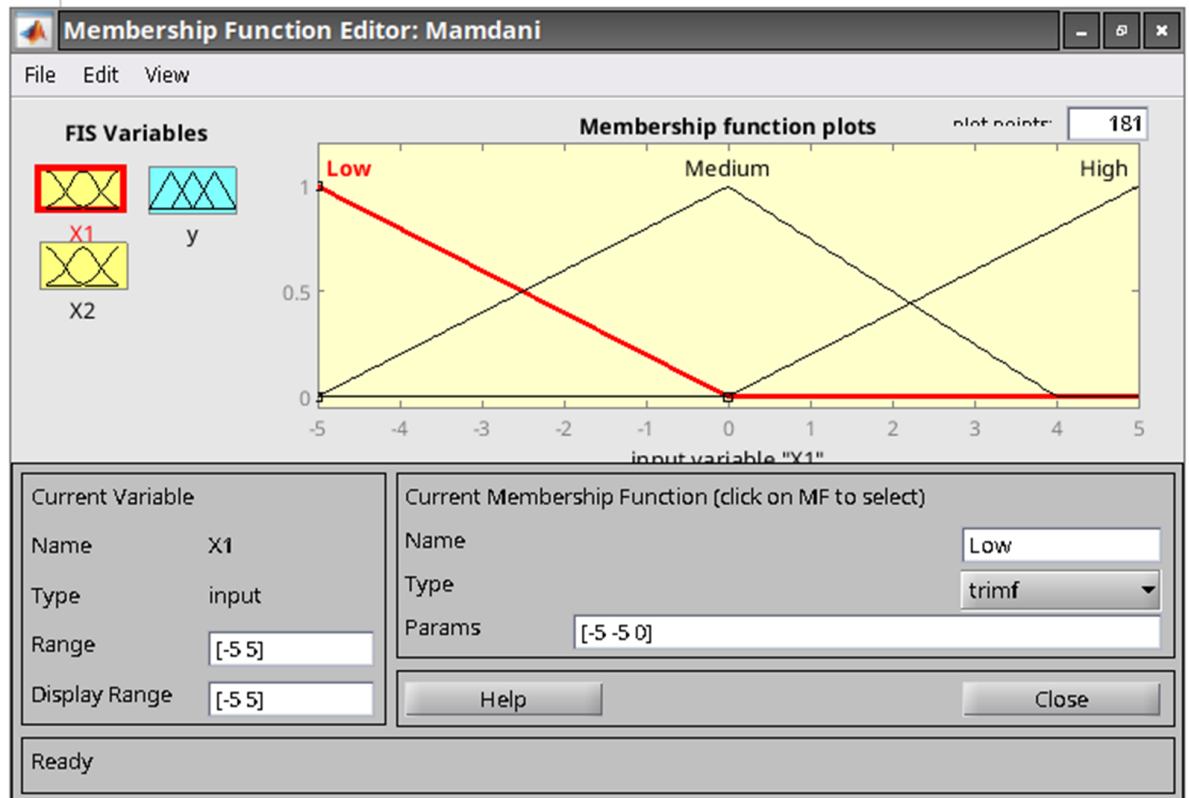
Налаштовуємо правила

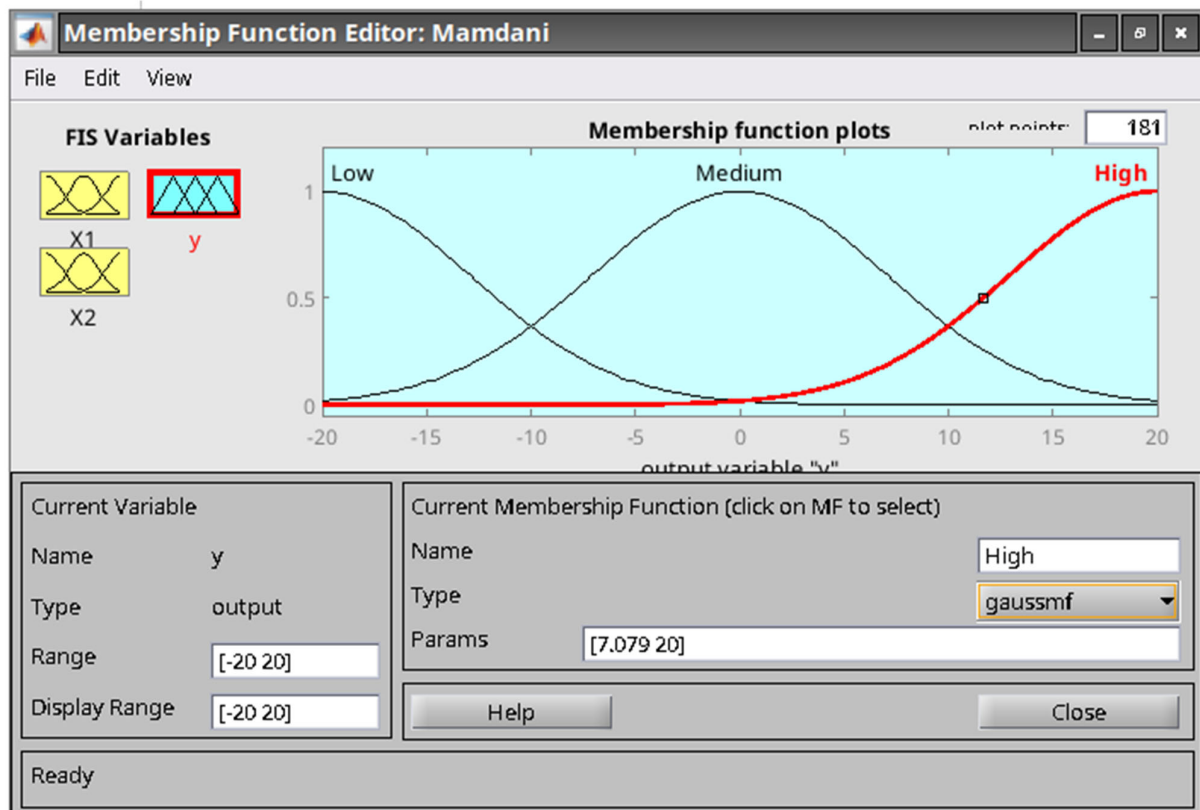


Будуємо графік поверхні



Аналогічно для системи Мамдані повторюємо тіж свмі кроки





Rule Editor: Mamdani

File Edit View Options

1. If (X1 is Low) and (X2 is Low) then (y is High) (1)
2. If (X1 is Medium) and (X2 is Medium) then (y is Low) (1)
3. If (X1 is High) and (X2 is High) then (y is High) (1)
4. If (X1 is Low) and (X2 is High) then (y is Medium) (1)
5. If (X1 is High) and (X2 is Low) then (y is Medium) (1)

If X1 is and X2 is Then y is

Low Low Low

Medium Medium Medium

High High High

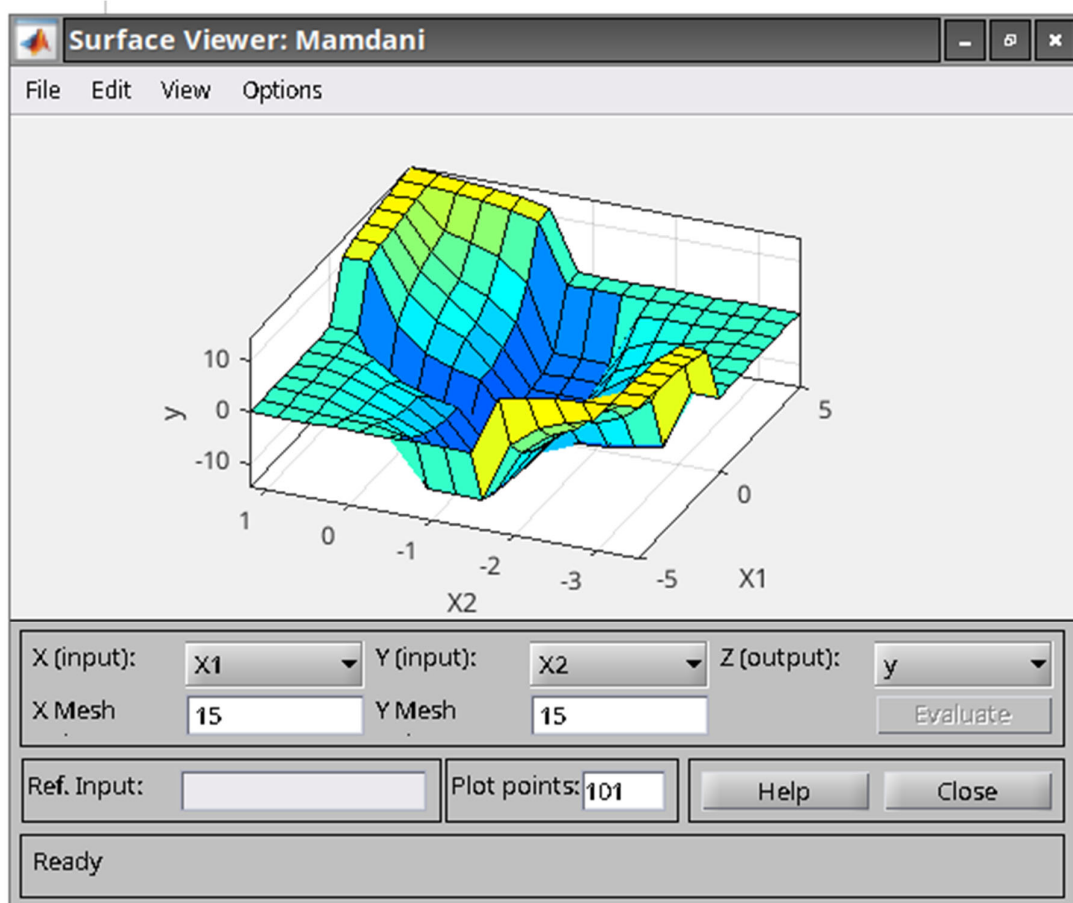
none none none

☐ not ☐ not ☐ not

Connection: ☐ or ☒ and Weight: 1

Delete rule Add rule Change rule

Ready Help Close



ВИСНОВКИ

В результаті виконаної лабораторної дослідили роботу алгоритмів Мамдані та Сугено в одностипних додатках.

Усі матеріали викладенні у репозиторії GitHub, за посиланням <https://github.com/Max11mus/LAB5-Modern-Methods-and-Models-of-Intelligent-Control-Systems>.