

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

ІНТЕЛЕКТУАЛЬНІ СИСТЕМИ АВТОМАТИЧНОГО КЕРУВАННЯ

**РЕАЛІЗАЦІЯ СИСТЕМ FUZZY-КЕРУВАННЯ
З ВИКОРИСТАННЯМ ПРОГРАМНИХ ЗАСОБІВ
ІНТЕГРОВАНОГО СЕРЕДОВИЩА MATLAB**

КОМП'ЮТЕРНИЙ ПРАКТИКУМ

*Рекомендовано Методичною радою КПІ ім. Ігоря Сікорського
як навчальний посібник для здобувачів ступеня магістра за освітньою програмою
"Інжиніринг інтелектуальних електротехнічних та мехатронних комплексів"
спеціальності 141 "Електроенергетика, електротехніка та електромеханіка"*

Електронне мережне навчальне видання

Київ
КПІ ім. Ігоря Сікорського
2024

УДК: 004.89:681.51] (076)

I-73

Укладачі: *Данілін Олександр Валерійович*, канд. техн. наук, доц.
Бойченко Сергій Валерійович, докт. техн. наук, проф.
Босак Алла Василівна, канд. техн. наук, доц.
Майданський Іван Ярославович, ст. викладач

Рецензент *Денисюк С.П.*, докт. техн. наук, проф.
професор кафедри електропостачання НН ІЕЕ

Відповідальний редактор *Бєлоха Г.С.*, канд. техн. наук, доц.

***Гриф надано Методичною радою КПП ім. Ігоря Сікорського
(протокол № 6 від 28.03.2024 р.)
за поданням Вченої ради НН ІЕЕ (протокол № 9 від 28.02.2024 р.)***

I-73 **Інтелектуальні системи автоматичного керування.** Реалізація систем fuzzy-керування з використанням програмних засобів інтегрованого середовища Matlab [Електронний ресурс] : комп. практикум : навч. посіб. для здобувачів ступеня магістра за освітньою програмою "Інжиніринг інтелектуальних електротехнічних та мехатронних комплексів" спец. 141 "Електроенергетика, електротехніка та електромеханіка" / КПП ім. Ігоря Сікорського ; уклад.: О.В. Данілін, С.В. Бойченко, А.В. Босак, І.Я. Майданський. – Електрон. текст. дані (1 файл). – Київ : КПП ім. Ігоря Сікорського, 2024. – 48 с.

У представленому посібнику викладено основні положення щодо виконання завдань комп'ютерного практикуму, тематика яких обіймає розділи курсу «Інтелектуальні системи автоматичного керування».

У комп'ютерному практикумі наведено методики дослідження математичних моделей одноканальних систем fuzzy-керування (нечіткого логічного виведення), розглянуто процеси фазифікації та дефазифікації нечітких множин, запропоновано методи складання алгоритмів моделювання нечітких систем типу Mamdani та Sugeno, а також отримання графічних результатів комп'ютерного моделювання з використанням спеціалізованих програмних засобів інтегрованого середовища Matlab.

Навчальне видання призначене для здобувачів ступеня магістра за освітньою програмою "Інжиніринг інтелектуальних електротехнічних та мехатронних комплексів" спеціальності 141 "Електроенергетика, електротехніка та електромеханіка" та може бути також корисним у навчальному процесі інших закладів вищої освіти під час підготовки професійних та наукових магістрів з електроенергетики, електротехніки та електромеханіки.

УДК: 004.89:681.51] (076)

Реєстр. № НП 23/24-371. Обсяг 2,4 авт. арк.
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
проспект Берестейський, 37, м. Київ, 03056
<https://kpi.ua>

Свідectво про внесення до Державного реєстру видавців, виготовлювачів
і розповсюджувачів видавничої продукції ДК № 5354 від 25.05.2017 р.

© О.В. Данілін, С.В. Бойченко, А.В. Босак, І.Я. Майданський
© КПП ім. Ігоря Сікорського, 2024

ЗМІСТ

Вступ	4
1. Інтерактивно-графічний модуль Fuzzy Logic Toolbox.....	5
1.1. Графічний інтерфейс редактора FIS-систем	6
1.2. Редактор нечіткої системи (Fuzzy Logic Designer).....	6
1.3. Редактор функцій приналежності (MF Editor).....	11
1.4. Редактор бази нечітких правил (Rule Editor)	15
1.5. Візуалізація процесу нечіткого логічного виведення (Rule Viewer)	18
1.6. Візуалізація поверхні "входи-вихід" (Surface Viewer)	20
2. Реалізація одноканальної системи fuzzy-керування типу Mamdani	24
3. Реалізація одноканальної системи fuzzy-керування типу Sugeno	28
4. Інтеграція fuzzy-системи у додаток Simulink.....	33
5. Робота з модулем Fuzzy Logic Toolbox у режимі командного рядку	36
5.1. Побудова графіків функцій приналежності	36
5.2. Виконання процесу дефазифікації.....	38
5.3. Відображення структури нечіткої моделі у вигляді системних команд	39
5.4. Відображення структури нечіткої моделі у вигляді блок-схеми	40
5.5. Відображення графіків функцій приналежності	41
5.6. Відображення графіку поверхні "входи-вихід"	41
5.7. Перетворення системи типу Mamdani у систему типу Sugeno	42
5.8. Заміна форми функцій приналежності еквівалентними	43
5.9. Виконання основних нечітких арифметичних операцій	44
6. Спеціальні Simulink-блоки побудови fuzzy-систем	45
Список використаних літературних джерел	48

ВСТУП

Одним з найцікавіших та найперспективніших напрямів розвитку інтелектуальних систем автоматичного та програмного управління електротехнічними комплексами є використання математичного апарату теорії нечіткої логіки (Fuzzy Logic) для створення спеціальних пристроїв, що дозволяють досить точно керувати різними технологічними процесами.

Актуальність цієї технології та її перевага перед відомими класичними методами теорії автоматичного керування полягає в тому, що поглиблюється тенденція збільшення складності математичних та фізичних моделей реальних процесів та систем, що пов'язано з бажанням підвищити їх адекватність та врахувати дедалі більше різних факторів, які впливають на процеси прийняття управлінських рішень. З одного боку, традиційні методи побудови моделей не призводять до задовільних результатів, коли вихідні дані проблеми, що підлягає вирішенню, свідомо є неточним або неповним. З іншого боку, прагнення отримати всю вичерпну інформацію для побудови точної математичної моделі складної реальної задачі призведе до втрати часу та коштів та може бути взагалі неможливою.

У таких випадках найбільш доцільно скористатися методами, які спеціально орієнтовані на побудову моделей, що враховують неповноту та неточність вихідних даних. Саме в таких ситуаціях технологія створення систем fuzzy-керування виявляється найбільш конструктивною.

У галузі управління електротехнічними комплексами моделювання систем fuzzy-керування дозволяє отримувати більш адекватні результати порівняно з результатами, що ґрунтуються на використанні традиційних аналітичних моделей та алгоритмів управління. Діапазон застосування нечітких методів з кожним роком розширюється, охоплюючи такі галузі, як проектування промислових роботів та побутових електроприладів, управління доменними печами та рухом поїздів метро, автоматичне розпізнавання звуків, зображень тощо.

Теорія нечіткої логіки та нечітких множин, яка є основою для реалізації методів fuzzy-керування, більш природно описує характер людського мислення та перебіг його міркувань, ніж традиційні формально-логічні системи. Саме тому вивчення та використання математичних засобів для представлення нечіткої вихідної інформації дозволяє будувати моделі, що найбільш адекватно відображають різні аспекти невизначеності.

Моделювання систем fuzzy-керування у середовищі Matlab здійснюється з використанням спеціального інтерактивного модуля Fuzzy Logic Toolbox, в якому реалізовано багато функціональних можливостей нечіткої логіки та нечіткого логічного виведення та крім того є змога інтегрування блок-схем нечітких систем у додаток Simulink.

Інтерактивне середовище Matlab, що містить спеціальні засоби нечіткого моделювання, дозволяє виконувати весь комплекс досліджень з розробки та впровадження fuzzy-систем. Саме з цих причин дана система була обрана як один із програмних засобів, в рамках якого можна реалізувати основні теоретичні концепції теорії нечітких множин та нечіткої логіки.

1. Інтерактивно-графічний модуль Fuzzy Logic Toolbox

Програмне середовище **Matlab** розроблено американською компанією **Math Works Inc.** та є однією з найбільш відомих спеціалізованих систем комп'ютерної математики, що містить спеціальні засоби нечіткого моделювання й дозволяє виконувати увесь спектр з дослідження, розробки та застосування моделей систем нечіткого логічного виведення.

Fuzzy Logic Toolbox – це пакет прикладних програм, що входить до складу середовища імітаційно-інтерактивного моделювання Matlab. Він дозволяє створювати системи нечіткого логічного виведення (системи fuzzy-керування) та нечіткої класифікації у рамках системи Matlab з можливістю їх інтегрування у **Simulink** – спеціальний програмний комплекс для моделювання систем, заданих функціональними блок-схемами.

До складу програмних засобів Fuzzy Logic Toolbox входять наступні програми, що дозволяють працювати в режимі інтерактивно-графічного інтерфейсу:

- редактор системи нечіткого логічного виведення Fuzzy Logic Designer (FIS Editor або FIS-редактор) разом із допоміжними програмами – редактором функцій приналежності (Membership Function Editor), редактором бази нечітких правил (Rule Editor), візуалізатором нечіткого логічного виведення (Rule Viewer) та візуалізатором поверхні "входи-вихід" (Surface Viewer);

- редактор гібридних нейро-нечітких (адаптивних) систем Adaptive Neuro – Fuzzy Inference System (ANFIS Editor або ANFIS -редактор);

- програма автоматичного знаходження центрів кластерів багатовимірних даних (Findcluster).

Набір даних програм надає користувачеві максимальні зручності для створення, редагування та використання різних систем fuzzy-керування.

Базовим поняттям пакету Fuzzy Logic Toolbox є FIS-структура – система нечіткого логічного виведення, що використовується для синтезу різних типів fuzzy-контролерів (регуляторів). FIS-структура містить усі необхідні дані для реалізації функціонального відображення поверхні "входи-вихід" на основі нечіткого логічного виведення згідно зі схемою, зображеною на рис. 1.1.

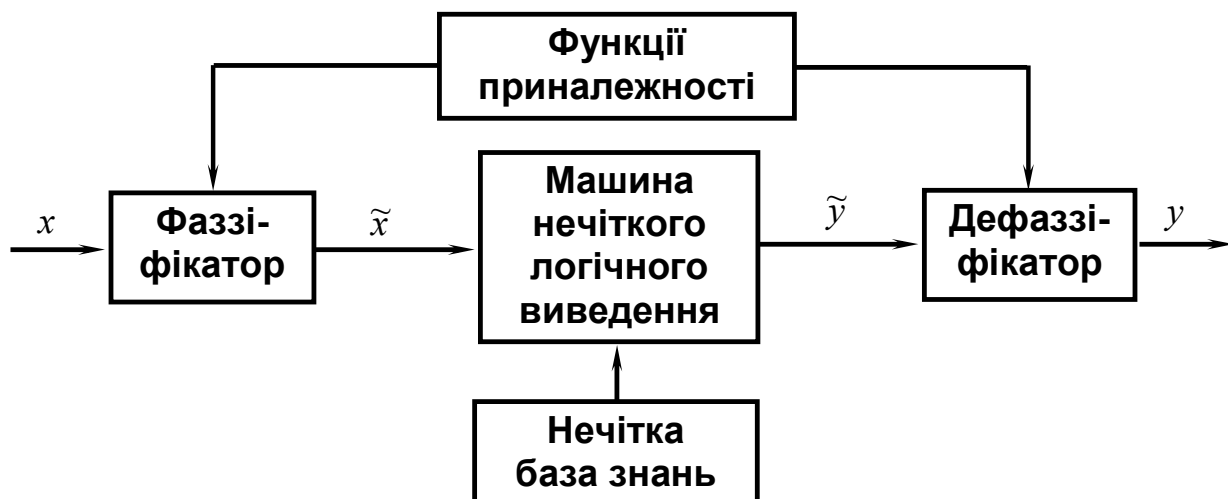


Рисунок 1.1 – Схема нечіткого логічного виводу

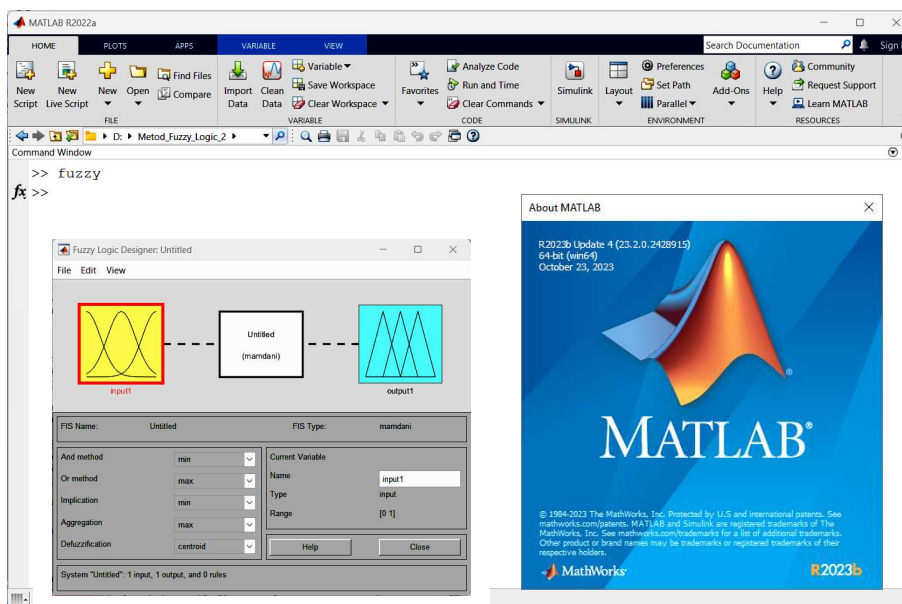
На рис. 1.1: x – вхідний вектор чітких значень; $\tilde{x}$ – вектор нечітких множин, що відповідає вхідному вектору x ; $\tilde{y}$ – результат логічного виведення у вигляді вектора нечітких множин; y – вихідний вектор чітких значень.

Структурно пакет Fuzzy Logic Toolbox містить наступні категорії програмних засобів:

- функції приналежності;
- інтерактивні модулі з графічним інтерфейсом користувача (GUI-модулі);
- функціонально-інтерактивні блоки для пакету Simulink;
- демонстраційні приклади (Fuzdemos).

За допомогою пакету Fuzzy Logic Toolbox можна конструювати FIS-системи двох типів – **Mamdani (Мамдані)** та **Sugeno (Сугено)**.

1.1. Графічний інтерфейс редактора FIS-систем



Редактор створення та редагування fuzzy-систем відкривається командою (функцією) **fuzzy** у вікні команд системи Matlab (рис. 1.2).

У даному посібнику використовується версія **MATLAB R2023b (Update 4)** з додатком **Fuzzy Logic Toolbox (Version 23.2)**.

Рисунок 1.2 – Запуск FIS-редактора

1.2. Редактор нечіткої системи (Fuzzy Logic Designer)

Редактор систем нечіткого логічного виведення (**Fuzzy Logic Designer, FIS Editor**) – основна інтерфейсна програма модулю Fuzzy Logic Toolbox, що призначена для створення, збереження, завантаження та виведення на друк різних систем нечіткого логічного виведення, а також для редагування наступних властивостей систем:

- тип нечіткої системи (Mamdani або Sugeno);
- ім'я системи;
- кількість вхідних та вихідних змінних;
- імена вхідних та вихідних змінних;
- параметри нечіткого логічного виведення.

Після виконання команди **fuzzy** на екрані з'являється інтерактивне графічне вікно, що зображено на рис. 1.3, де вказані функціональні призначення основних полів графічного вікна. За замовчуванням середовище Matlab відкриває для створення та редагування систему типу Mamdani.

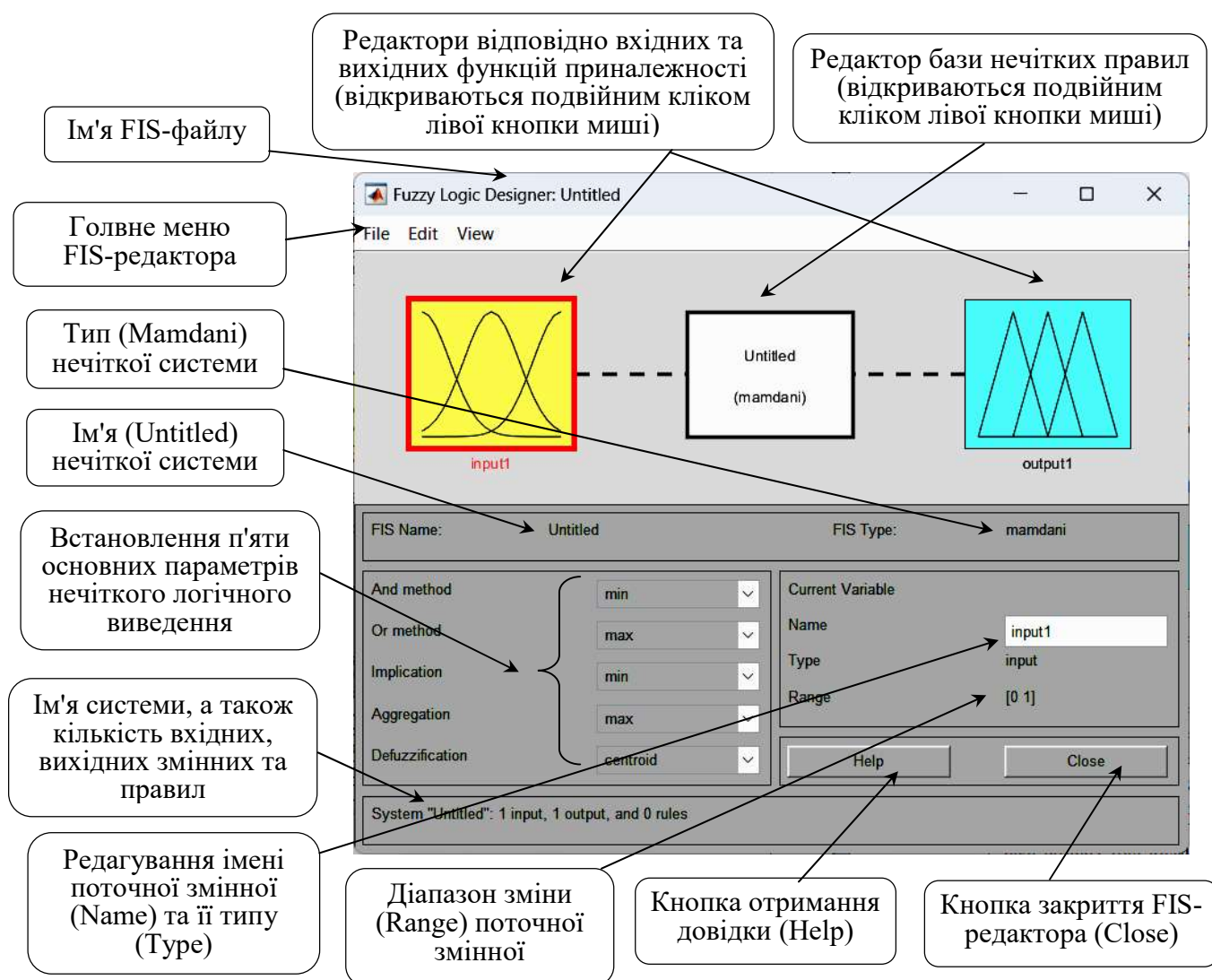


Рисунок 1.3 – Вигляд вікна FIS-редактора для системи типу Mamdani

У верхній частині FIS-редактора розташований рядок, що містить ім'я fis-файлу (**Fuzzy Logic Designer: Untitled**). Нижче знаходиться головне меню редактора, що містить три пункти (**File, Edit, View**). Під ним у загальному вигляді зображена функціональна схема синтезуємої системи нечіткого логічного виведення, що включає блоки завдання та редагування вхідних (**input1**) та вихідних (**output1**) функцій приналежності, а також блок редактора бази нечітких правил (**Untitled (Mamdani)**). Нижче розташовано поле, в якому відображено ім'я (**FIS Name: Untitled**) та тип (**FIS Type: mamdani**) системи нечіткого логічного виведення. Далі паралельно розміщено два поля: перше дозволяє встановити п'ять параметрів нечіткого логічного виведення (**And Method, Or Method, Implication, Aggregation, Defuzzification**), а у другому (**Current Variable**) задається ім'я (**Name**) поточної змінної (**input1**) та відображається її тип (**Type**), а також діапазон (**Range**) зміни (**[0 1]**). Крім того, друге поле містить кнопки (**Help**) та (**Close**), що призначені для виклику довідкової інформації та закриття FIS-редактора відповідно.

Докладно розглянемо всі три пункти головного меню **FIS-редактора**.

FILE

Цей пункт головного меню загальний для усіх GUI-модулів, що використовуються у системах нечіткого логічного виведення. Загальний вигляд пункту **File** зображено на рис. 1.4. З його допомогою здійснюється робота з файлами моделей (їх створення, збереження, зчитування та друк).

Розглянемо послідовно усі підпункти (команди) цього меню.

Команда **New FIS...** дозволяє користувачеві створити нову систему нечіткого логічного виведення.

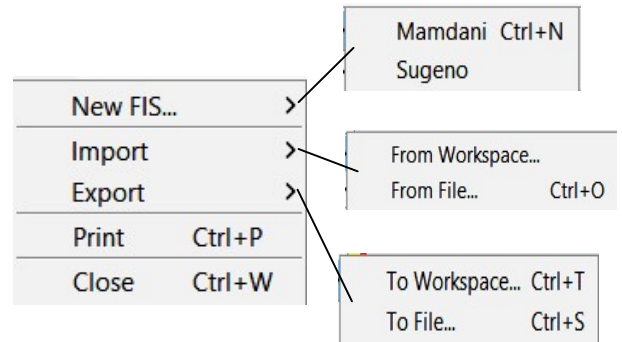


Рисунок 1.4 – Пункт головного меню File з вкладеними підпунктами

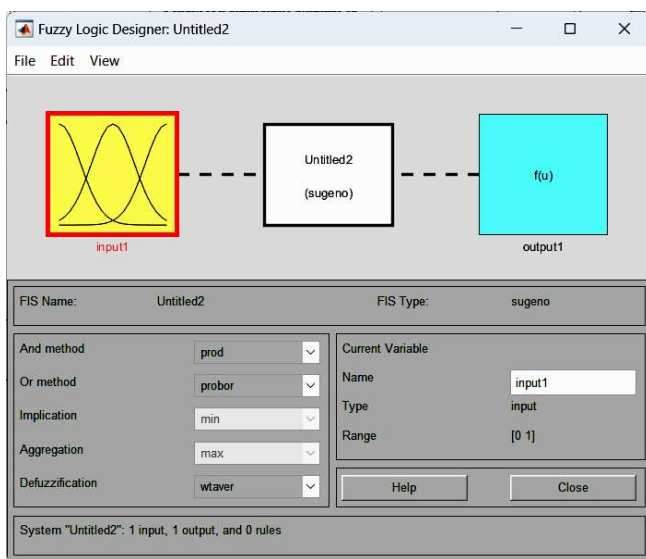


Рисунок 1.5 – Вигляд вікна FIS-редактора для системи типу Sugeno

Після вибору цієї команди можливі дві альтернативи: **Mamdani (Ctrl + N)** та **Sugeno**, що визначають тип створюваної нечіткої системи.

У разі вибору системи **Mamdani** відкривається вікно, що зображено на рис. 1.3 а для систем типу **Sugeno** відкриється вікно, що зображено на рис. 1.5. Графічний інтерфейс такого редактора ідентичний інтерфейсу редактора систем типу **Mamdani**.

За допомогою команди **Import** користувач може завантажити раніше створену систему нечіткого логічного виведення.

Після вибору цієї команди можливі дві альтернативи: **From Workspace...** та **From File (Ctrl + O)**, що дозволяють завантажити систему нечіткого логічного виведення з робочої області системи Matlab та з диска відповідно. Під час вибору команди **From Workspace...** з'явиться діалогове вікно, яке зображено на рис. 1.6, де необхідно вказати ідентифікатор (ім'я) системи нечіткого логічного виведення, що знаходиться у робочій області системи Matlab.

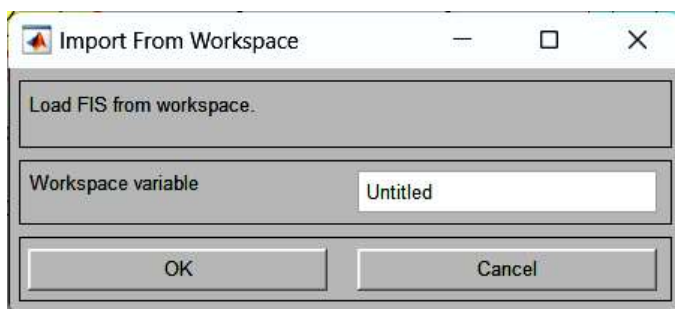


Рисунок 1.6 – Вікно завантаження нечіткої системи з робочої області Matlab

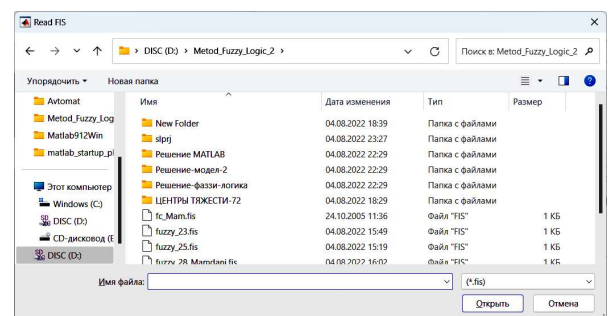


Рисунок 1.7 – Вікно завантаження нечіткої системи з диску

Під час вибору команди **From File (Ctrl+O)** з'явиться діалогове вікно (рис. 1.7), в якому необхідно обрати диск, папку та вказати ім'я файлу системи нечіткого логічного виведення. Файли нечітких систем мають розширення **.fis**. Завантажити систему нечіткого логічного виведення з диску можна також командою **fuzzy FIS_name** з вікна команд системи Matlab, де **FIS_name** – ім'я файлу нечіткої системи.

Після вибору команди **Export** можливі дві альтернативи: **To Workspace... (Ctrl+T)** та **To File (Ctrl+S)**, що дозволяють зберегти систему нечіткого логічного виведення у робочу область системи Matlab або на диск відповідно. У разі вибору команди **To Workspace... (Ctrl + T)** з'явиться діалогове вікно, що зображено на рис. 1.8, в якому необхідно вказати ідентифікатор (ім'я) системи нечіткого логічного виведення, під яким вона буде збережена у робочій області системи Matlab. Після вибору команди **To File** з'явиться діалогове вікно, що зображено на рис. 1.9, в якому необхідно вказати ім'я файлу нечіткої системи (за замовчуванням – безіменний **Untitled**).

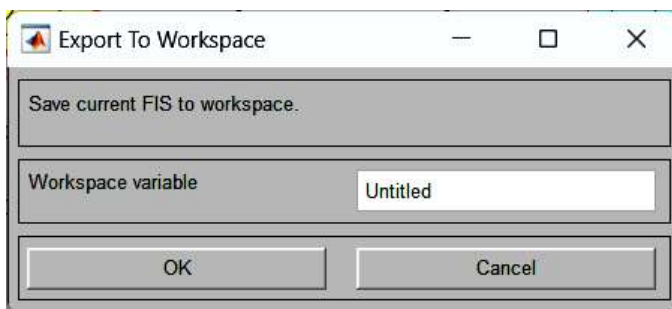


Рисунок 1.8 – Вікно експортування нечіткої системи у робочу область Matlab

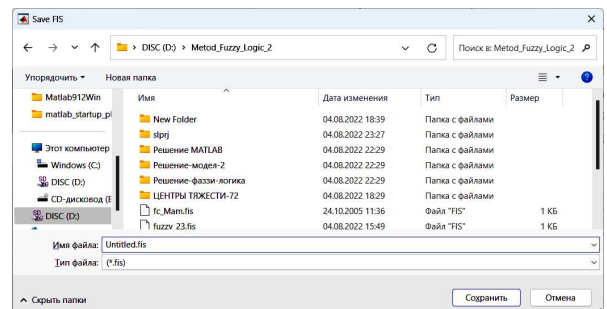


Рисунок 1.9 – Вікно запису нечіткої системи на диск

Команда **Print (Ctrl + P)** дозволяє вивести на друк копію вікна інтерфейсу FIS-редактора. Після виконання цієї команди з'явиться вікно, що зображено на рис. 1.10, у якому за потреби встановлюються параметри процесу друку та властивості принтера.

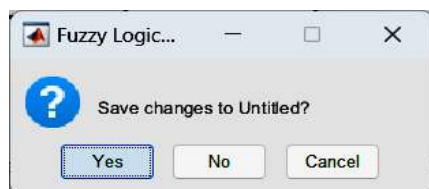


Рисунок 1.11 – Вікно запиту на збереження моделі перед закриттям

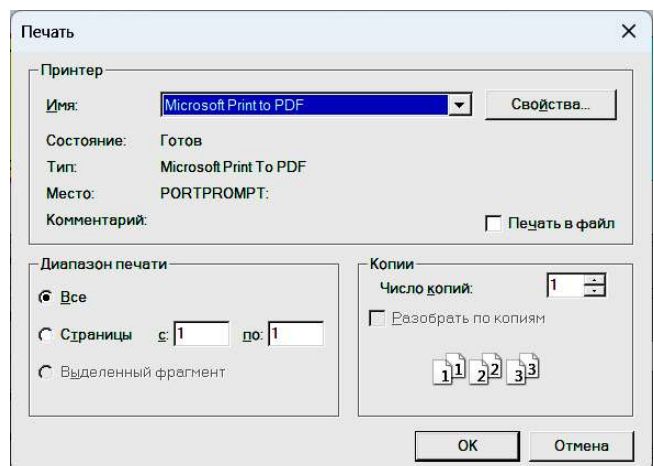


Рисунок 1.10 – Виведення на друк копії вікна FIS-редактора

Команда **Close (Ctrl + W)** закриває графічне вікно FIS-редактора. Після виконання цієї команди з'являється діалогове вікно (рис. 1.11), що вказує на необхідність збереження файлу нечіткої системи.

EDIT

Загальний вигляд пункту **Edit** головного меню FIS-редактора зображено на рис. 1.12. З його допомогою здійснюються операції редагування системи нечіткого логічного виведення (додавання та виключення вхідних та вихідних змінних, а також відкриття редактора функцій приналежності та бази нечітких правил).

Команда **Undo (Ctrl + Z)** дозволяє користувачеві скасувати раніше скоєну дію. Якщо вона не активна, отже, раніше не було виконано жодної дії.

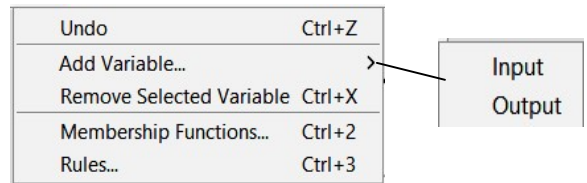


Рисунок 1.12 – Пункт головного меню **Edit** з вкладеним підпунктом

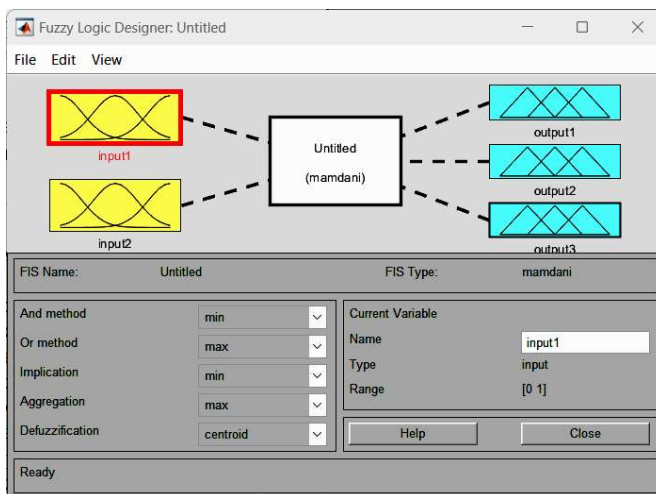


Рисунок 1.13 – Додавання у систему однієї вхідної та двох вихідних змінних

Команда **Add Variable...** дозволяє додавати до системи нечіткого логічного виведення додаткові змінні. Після вибору цієї команди можливі дві альтернативи: **Input** та **Output**, що дозволяють додати вхідну та вихідну змінну відповідно. Наприклад, під час додавання до нечіткої системи однієї вхідної **input2** та двох вихідних **output2** і **output3** змінних, функціональна схема системи у вікні FIS-редактора набуде вигляду, зображеного на рис. 1.13.

Команда **Remove Selected Variable (Ctrl+X)** видаляє поточну змінну з системи. Ознакою поточної змінної є червона межа прямокутника. Призначення поточної змінної відбувається за допомогою кліку лівою кнопкою миші на її полі.

Команда **Membership Function... (Ctrl+2)** відкриває редактор функцій приналежності, а команда **Rules... (Ctrl+3)** – редактор бази нечітких правил, що будуть детально розглянуті нижче.

VIEW

Цей пункт головного меню загальний для усіх GUI-модулів, що використовуються у системах нечіткого логічного виведення. Загальний вигляд пункту **View** головного меню FIS-редактора зображено на рис. 1.14. З його допомогою здійснюється перехід до додаткового інструментарію.

У цьому пункті всього дві команди: **Rules (Ctrl + 5)** та **Surface (Ctrl + 6)**, що дозволяють відкрити вікно візуалізації нечіткого логічного виведення та вікно поверхні "входи-вихід" відповідно.

Розглянемо п'ять основних параметрів нечіткої логічної системи, що встановлюються у лівій частині вікна FIS-редактора (рис. 1.3).

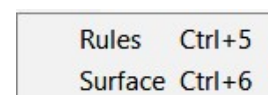


Рисунок 1.14 – Пункт головного меню **View**

МЕТОДИ ЛОГІЧНОГО ОБРОБЛЕННЯ ПАРАМЕТРІВ FUZZY-СИСТЕМИ

And Method дозволяє встановити наступні реалізації логічної операції ТА (кон'юнкції): **min** – метод мінімального значення; **prod** – метод добутку значень.

Також є можливість встановити власну реалізацію операції ТА. Для цього необхідно обрати команду **Custom...** та у вікні налаштувань вказати ім'я функції, що реалізує цю операцію (рис. 1.15).

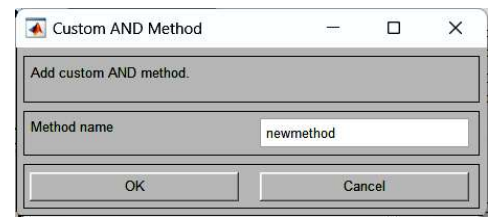


Рисунок 1.15 – Вікно з ім'ям функції, що реалізує логічну операцію ТА

Or Method – дозволяє встановити наступні реалізації логічної операції АБО (диз'юнкції): **max** – метод максимального значення; **probor** – метод ймовірнісного значення операції АБО; **Custom...** – дозволяє встановити власну реалізацію операції АБО (вікно налаштування ідентично рис. 1.15 за винятком назви методу).

Implication – дозволяє встановити наступні значення реалізації виведення (імплікації): **min** – метод мінімального значення; **prod** – метод добутку значень; **Custom...** – дозволяє встановити власну реалізацію імплікації (вікно налаштування ідентично рис. 1.15, за винятком назви методу). У системах типу Sugeno це меню не активне, через специфіку алгоритму таких систем (рис. 1.5).

Aggregation – дозволяє встановити наступні реалізації логічної операції поєднання функцій приналежності вихідної змінної (агрегування): **max** – метод максимального значення; **sum** – метод суми значень; **probor** – метод ймовірнісного значення операції АБО; **Custom...** – дозволяє встановити власну реалізацію цієї операції (вікно налаштування ідентично рис. 1.15, за винятком назви методу). У системах типу Sugeno це меню не активне, через специфіку алгоритму таких систем (рис. 1.5).

Defuzzification – дозволяє обрати потрібний метод дефазифікації.

Для систем типу Mamdani запрограмовані наступні методи дефазифікації: **centroid** – гравітаційний метод; **bisector** – метод медіани; **mom** – метод середнього з максимумів; **lom** – метод найбільшого з максимумів; **som** – метод найменшого з максимумів; **Custom...** – дозволяє встановити власний метод дефазифікації (вікно налаштування ідентично рис. 1.15, за винятком назви методу).

Для систем типу Sugeno запрограмовані наступні методи дефазифікації: **wtaver** – метод зваженого середнього значення; **wtsum** – метод зваженого сумарного значення.

1.3. Редактор функцій приналежності (MF Editor)

Редактор функцій приналежності (**Membership Function Editor**) призначений для завдання та редагування вхідних та вихідних функцій приналежності окремих термів системи нечіткого логічного виведення у

графічному режимі. За допомогою цього редактора можливе завдання наступної інформації про терми-множини вхідних та вихідних змінних:

- кількість термів;
- найменування термів;
- тип та параметри функцій приналежності, що необхідні для представлення лінгвістичних термів в якості нечітких множин.

Редактор функцій приналежності може бути викликаний з будь-якого GUI-модуля, що використовується у системах нечіткого логічного виведення.

Виклик редактора функцій приналежності здійснюється командою **Membership Function... (Ctrl +2)** з пункту **Edit** головного меню FIS-редактора. Також він може бути відкритий подвійним кліком лівої кнопки миші на полі вхідних або вихідних змінних у функціональній блок-схемі верхньої частини FIS-редактору (рис. 1.3).

Загальний вигляд редактора функцій приналежності із зазначенням функціонального призначення основних полів графічного вікна зображено на рис. 1.16.

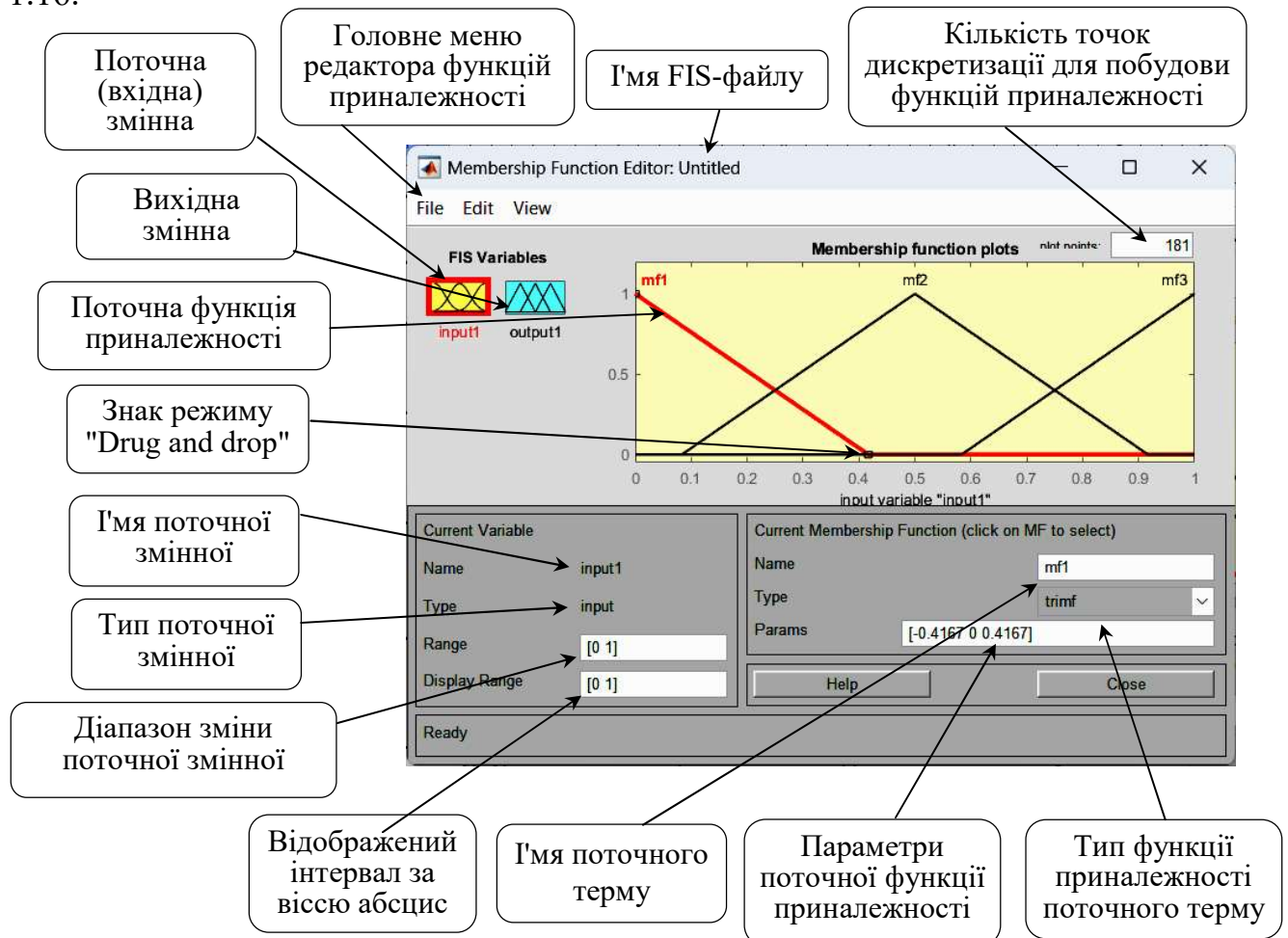


Рисунок 1.16 – Вигляд вікна редактора функцій приналежності

У верхній частині редактора функцій приналежності розташований рядок, що містить ім'я fis-файлу (**Membership Function Editor: Untitled**). Нижче знаходиться головне меню редактора, що містить три пункти (**File, Edit, View**).

Під ним відображені блоки вхідної (**input1**) та вихідної (**output1**) функцій приналежності, одна з яких є поточною (для обрання поточної змінної потрібно на необхідному блоці один раз клікнути лівою кнопкою миші).

Праворуч розташоване вікно для зазначення кількості точок дискретизації побудови графічного зображення функцій приналежності (**plot points: 181**). Нижче наведено графічні зображення функцій приналежності (**Membership function plots**), на рис. 1.16 це три трикутники, іменовані як **mf1**, **mf2** та **mf3** (поточним у цьому прикладі є терм функції приналежності **mf1**). Під графіком вказано ім'я поточної вхідної змінної (**input1**).

Нижче паралельно розміщені два поля. Перше містить інформацію про поточну змінну (**Current Variable**) – ім'я (**input1**) та тип (**input**) вхідної змінної. У цьому ж полі розташовані два вікна: (**Range**) та (**Display Range**), які призначені для завдання діапазону зміни поточної функції приналежності (**[0, 1]**) та діапазону вісі абсцис виведення її на екран (**[0, 1]**) відповідно.

Друге поле містить інформацію щодо поточної функції приналежності (**Current Membership Function**). Для вибору поточної функції потрібно клікнути один раз лівою кнопкою миші на відповідному термі (**click on MF to select**). У вікні (**Name**) відображається ім'я поточного терму функції приналежності (**mf1**). У вікні (**Type**) вкладено підменю, що дозволяє встановити тип функцій приналежності термів, що використовуються для лінгвістичної оцінки поточної змінної (**trimf**). На рис. 1.17 зображено меню **Type**, в якому вказані можливі типи користувацьких функцій приналежності нечітких множин.



Рисунок 1.17 – Меню Type

Нижче розташовано вікно (**Params**), що призначене для встановлення вектора параметрів поточного терму функції приналежності (**[-0.4 0 0.4]**).

Крім того, внизу знаходяться кнопки (**Help**) та (**Close**), що призначені для виклику довідкової інформації та закриття редактора функцій приналежності відповідно.

Слід зазначити, що додавання будь-яких чисельних значень у вікна для введення інформації потрібно закінчувати натисканням клавіші **Enter**.

Параметри функцій приналежності можна підбирати і у графічному режимі, змінюючи форму функцій приналежності за допомогою технології "**Drug and drop**". Для цього необхідно позиціонувати курсор миші на знаку режиму "**Drug and drop**" (рис. 1.16), натиснути ліву кнопку і, не відпускаючи її, змінити форму функції приналежності. Параметри функції приналежності будуть перераховуватися автоматично.

Розглянемо головне меню редактора функцій приналежності (**MF-редактора**). Пункти **File** та **View** однакові для усіх GUI-модулів, що використовуються у системах нечіткого логічного виведення. Вони докладно описані під час розгляду FIS-редактора.

EDIT

Загальний вигляд пункту **Edit** головного меню редактора функцій приналежності зображено на рис. 1.18. З його допомогою здійснюються операції редагування функцій приналежності.

Команда **Undo (Ctrl + Z)** дозволяє користувачеві скасувати раніше скоєну дію. Якщо вона не активна, отже, раніше не було виконано жодної дії.

Undo	Ctrl+Z
Add MFs...	
Add Custom MF...	
Remove Selected MF	
Remove All MFs	
FIS Properties...	Ctrl+1
Rules...	Ctrl+3

Рисунок 1.18 – Пункт головного меню Edit

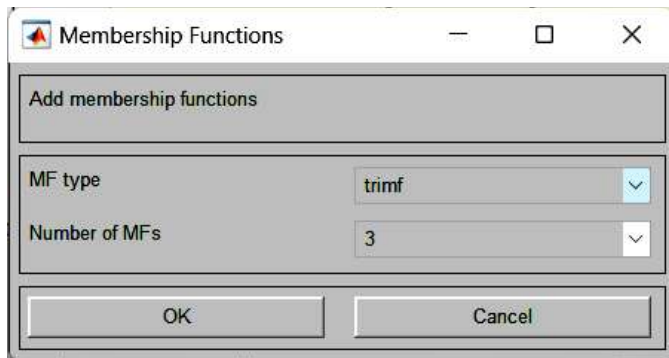


Рисунок 1.19 – Вікно додавання нового терму

Значення параметрів функцій приналежності будуть встановлені автоматично таким чином, щоб рівномірно вкрити область визначення змінної, заданої у вікні (**Range**), згідно з рис. 1.16. У разі зміни області визначення у вікні (**Range**) параметри функцій приналежності будуть масштабовані.

Команда **Add Custom MF...** дозволяє додати один лінгвістичний терм, функція якого відрізняється від вбудованих. Після вибору цієї команди з'явиться вікно налаштувань, що зображено на рис. 1.20 (**Add customized membership function**), в якому необхідно ввести ім'я лінгвістичного терму (**поле MF name**), тип функції приналежності (**поле M-File function name**) та її параметри (**поле Parameter list**).

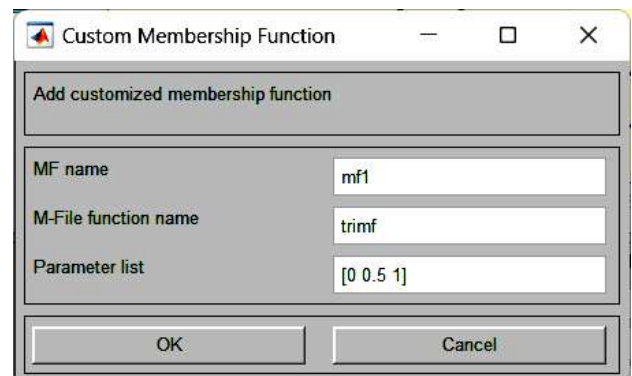


Рисунок 1.20 – Вікно додавання нового терму, що відрізняється від вбудованих

Команда **Remove Selected MF** видаляє поточний терм з термів-множин поточної змінної. Ознакою поточної змінної є червона межа прямокутника. Ознакою поточного терму є червоний колір його функції приналежності. Для обрання поточного терму необхідно позиціонувати курсор миші на відповідному графіку функції приналежності та клікнути лівою кнопкою.

Команда **Remove All MFs** видаляє усі терми з термів-множин поточної змінної.

Команда **FIS Properties... (Ctrl +1)** відкриває FIS-редактор (рис. 1.3), а команда **Rules... (Ctrl +3)** відкриває редактор бази нечітких правил (рис. 1.21).

1.4. Редактор бази нечітких правил (Rule Editor)

Редактор бази нечітких правил (**Rule Editor**) призначений для формування, редагування та модифікації окремих правил систем нечіткого логічного виведення. Редактор бази нечітких правил може бути викликаний з будь-якого GUI-модуля, що використовується для дослідження нечітких систем. З FIS-редактора та редактора функцій приналежності його можна викликати командою **Rules... (Ctrl +3)** з пункту **Edit** головного меню.

У FIS-редакторі (рис. 1.3) відкрити редактор бази нечітких правил можна також подвійним кліком лівої кнопки миші на прямокутнику з назвою системи нечіткого логічного виведення, що розташований у центрі графічного вікна.

Загальний вигляд редактора бази нечітких правил із зазначенням функціонального призначення основних полів графічного вікна зображено на рис. 1.21.

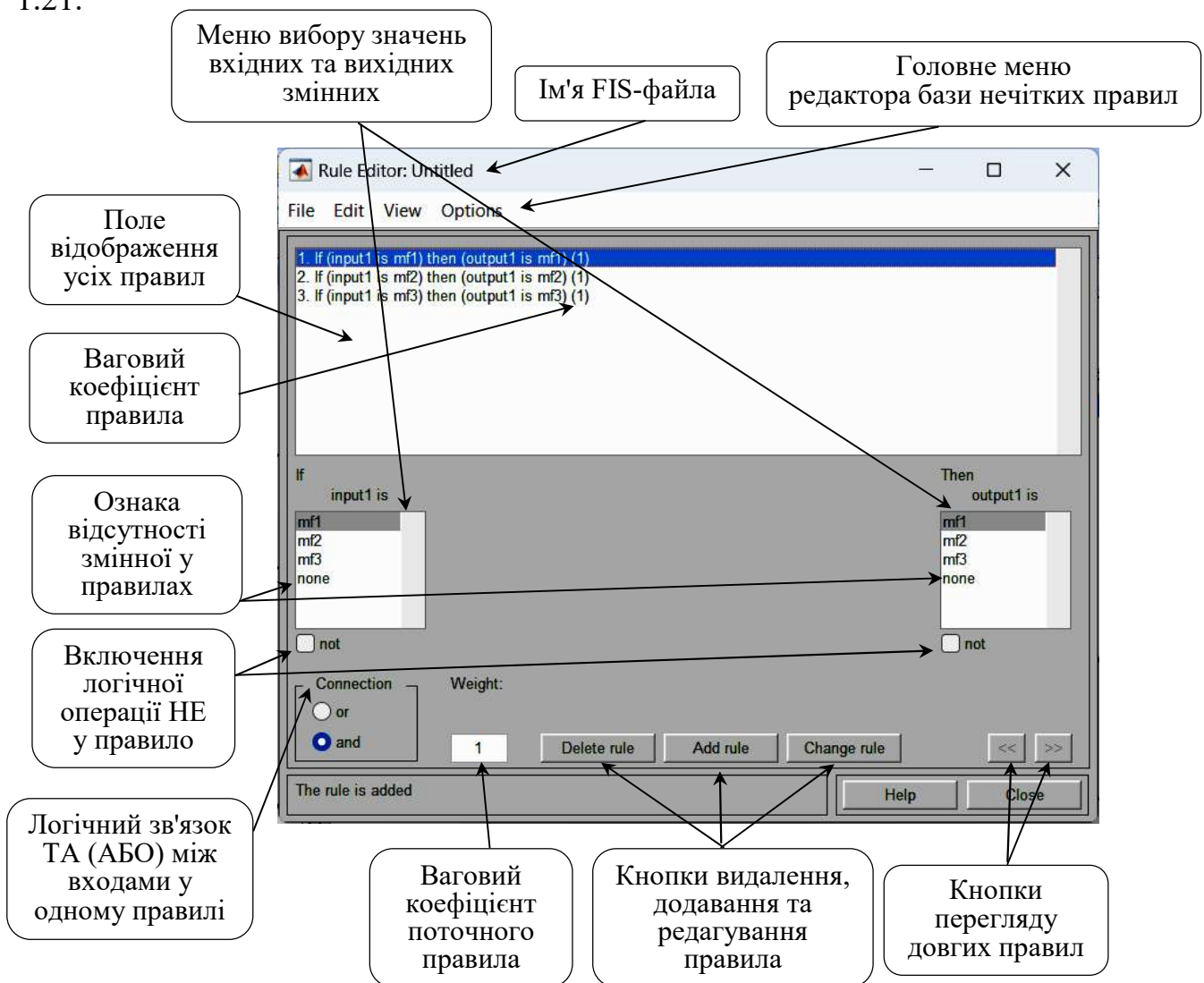


Рисунок 1.21 – Вигляд вікна редактора бази нечітких правил

У верхній частині редактора бази нечітких правил розташований рядок, що містить ім'я fis-файлу (**Rule Editor: Untitled**). Нижче знаходиться головне меню редактора, що містить чотири пункти (**File, Edit, View, Options**). Під ним відображається поле розташування усіх нечітких правил.

Нижче знаходяться вікна з полями вибору термів вхідних та вихідних змінних (**mf1, mf2, mf3, none**). На рис. 1.21 зображені два вікна **ЯКЩО (If)** та **ТО (Then)** для вхідної (**input1**) та вихідної (**output1**) змінних. У разі наявності кількох змінних додається відповідна кількість вікон, і правила будуть об'єднуватися за допомогою логічних операторів **ТА (and)** або **АБО (or)**, що розташовані у вікні встановлення логічних операцій (**Connection**). Під кожним вікном вхідних та вихідних змінних розташовані поля для встановлення опцій, коли необхідно увімкнути або вимкнути логічну операцію **НЕ (not)**.

У нижній частині вікна редактора бази нечітких правил у полі (**Weight**) вказують відповідні вагові коефіцієнти кожного нечіткого правила. Далі розташовані три кнопки (**Delete Rule, Add Rule, Change Rule**), що дозволяють додавати, видаляти та редагувати правила відповідно.

Для перегляду довгих правил, що не вміщуються у полі відображення, використовують дві кнопки переміщення по рядку правил (кнопка << – призначена для переміщення ліворуч, а кнопка >> – для переміщення праворуч).

Крім того, знизу розташовані кнопки (**Help**) та (**Close**), що призначені для виклику довідкової інформації та закриття редактора бази нечітких правил відповідно.

Для введення нового правила у базу знань необхідно: за допомогою миші обрати відповідну комбінацію лінгвістичних термів вхідних та вихідних змінних з вікон (**If**) та (**Then**); встановити, якщо необхідно, тип логічної зв'язки (**ТА** або **АБО**) з вікна (**Connection**) між змінними всередині правила; встановити наявність чи відсутність логічної операції **НЕ** для кожної лінгвістичної змінної; ввести значення вагового коефіцієнту правила з вікна (**Weight**) та натиснути (**Add Rule**).

На рис. 1.21 зображені правила, що встановлені відповідно до наступних логічних міркувань: **ЯКЩО вхід mf1 ТО вихід mf1; ЯКЩО вхід mf2 ТО вихід mf2; ЯКЩО вхід mf3 ТО вихід mf3**.

За замовчуванням у системі нечіткого логічного виведення встановлено наступні параметри: логічна зв'язка змінних усередині правила – **ТА**; логічна операція **НЕ** – відсутня; значення вагового коефіцієнта правила – **1**.

Можливі випадки, коли істинність правила не змінюється у разі довільного значення певної вхідної змінної, тобто ця змінна не впливає на результат нечіткого логічного виведення у цій галузі факторного простору. Тоді в якості лінгвістичного значення цієї змінної необхідно встановити (**none**).

Для видалення правила з бази знань необхідно клікнути лівою кнопкою миші на цьому правилі та натиснути кнопку (**Delete Rule**).

Для модифікації правила необхідно клікнути лівою кнопкою миші на цьому правилі, потім натиснути кнопку (**Change Rule**) та встановити необхідні параметри правила.

Необхідно зауважити, що додавання будь-яких чисельних значень у вікна для введення інформації потрібно закінчувати натисканням клавіші **Enter**.

Розглянемо головне меню редактора бази нечітких правил (**RULE-редактора**). Пункти **File** та **View** однакові для усіх GUI-модулів, що використовуються у системах нечіткого логічного виведення. Вони докладно описані під час розгляду FIS-редактора.

EDIT

Загальний вигляд пункту **Edit** головного меню редактора бази правил зображено на рис. 1.22. З його допомогою здійснюються операції редагування усіх правил, що відповідають системі нечіткого логічного виведення.

Undo	Ctrl+Z
FIS Properties...	Ctrl+1
Membership Functions...	Ctrl+2

Рисунок 1.22 – Пункт головного меню Edit

Команда **Undo (Ctrl + Z)** дозволяє користувачеві скасувати раніше скоєну дію. Якщо вона не активна, отже, раніше не було виконано жодної дії.

Команди **FIS Properties... (Ctrl+1)** та **Membership Function... (Ctrl +2)** відкривають FIS-редактор (рис. 1.3) та редактор функцій приналежності (рис. 1.16) відповідно.

OPTIONS

Загальний вигляд пункту **Options** головного меню редактора бази нечітких правил зображено на рис. 1.23. Цей пункт дозволяє встановити мову та формат правил системи нечіткого логічного виведення.

Після вибору команди **(Language)** з'явиться вікно зі списком мов: **English** (Англійська), **Deutsch** (Німецька), **Francais** (Французька), з якого необхідно обрати одну. За замовчуванням у редакторі прийнята англійська мова.

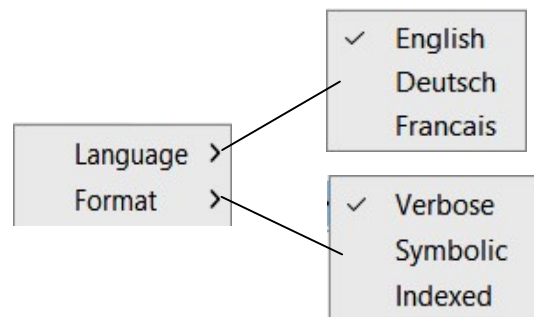


Рисунок 1.23 – Пункт головного меню Options з підпунктими

У разі вибору команди **(Format)** з'явиться список можливих форматів завдання правил бази нечітких знань: **(Verbose)** – лінгвістичний; **(Symbolic)** – логічний; **(Indexed)** – індексований.

На рис. 1.24 зображено лінгвістичний **(Verbose)** формат виведення бази нечітких правил, на рис. 1.25 зображено логічний **(Symbolic)** формат виведення бази нечітких правил, та рис. 1.26 зображено індексований **(Indexed)** формат виведення бази нечітких правил.

1. If (input1 is mf1) then (output1 is mf1) (1)
2. If (input1 is mf2) then (output1 is mf2) (1)
3. If (input1 is mf3) then (output1 is mf3) (1)

Рисунок 1.24 – Лінгвістичний формат бази нечітких правил

1. (input1==mf1) => (output1=mf1) (1)
2. (input1==mf2) => (output1=mf2) (1)
3. (input1==mf3) => (output1=mf3) (1)

Рисунок 1.25 – Логічний формат бази нечітких правил

1. 1 (1) : 1
2. 2 (1) : 1
3. 3 (1) : 1

Рисунок 1.26 – Індексований формат бази нечітких правил

1.5. Візуалізація процесу нечіткого логічного виведення (Rule Viewer)

Візуалізація процесу нечіткого логічного виведення здійснюється за допомогою GUI-модуля (**Rule Viewer**). Цей модуль дозволяє графічно відобразити увесь процес логічного виведення за кожним правилом, отримати результуючу нечітку множину та виконати процедуру дефаззифікації.

У вікні перегляду процесу нечіткого логічного виведення не можна редагувати правила та функції приналежності термів змінних. Це вікно використовується після синтезування системи нечіткого виведення на етапі її аналізу та оцінки. Його також доцільно застосовувати у тому випадку, коли необхідно візуально представити увесь процес нечіткого логічного виведення від початку до кінця та зробити певні висновки. При цьому є можливість оцінити значення вихідних змінних fuzzy-моделі та вплив кожного з правил на результат нечіткого виведення за допомогою зміни значень вхідних змінних.

Вікно **Rule Viewer** може бути відкрито з будь-якого GUI-модуля, що використовується у системах нечіткого логічного виведення. З FIS-редактора, редактора функцій приналежності та редактора бази нечітких правил його можна викликати командою **Rules (Ctrl +5)** з пункту **View** головного меню.

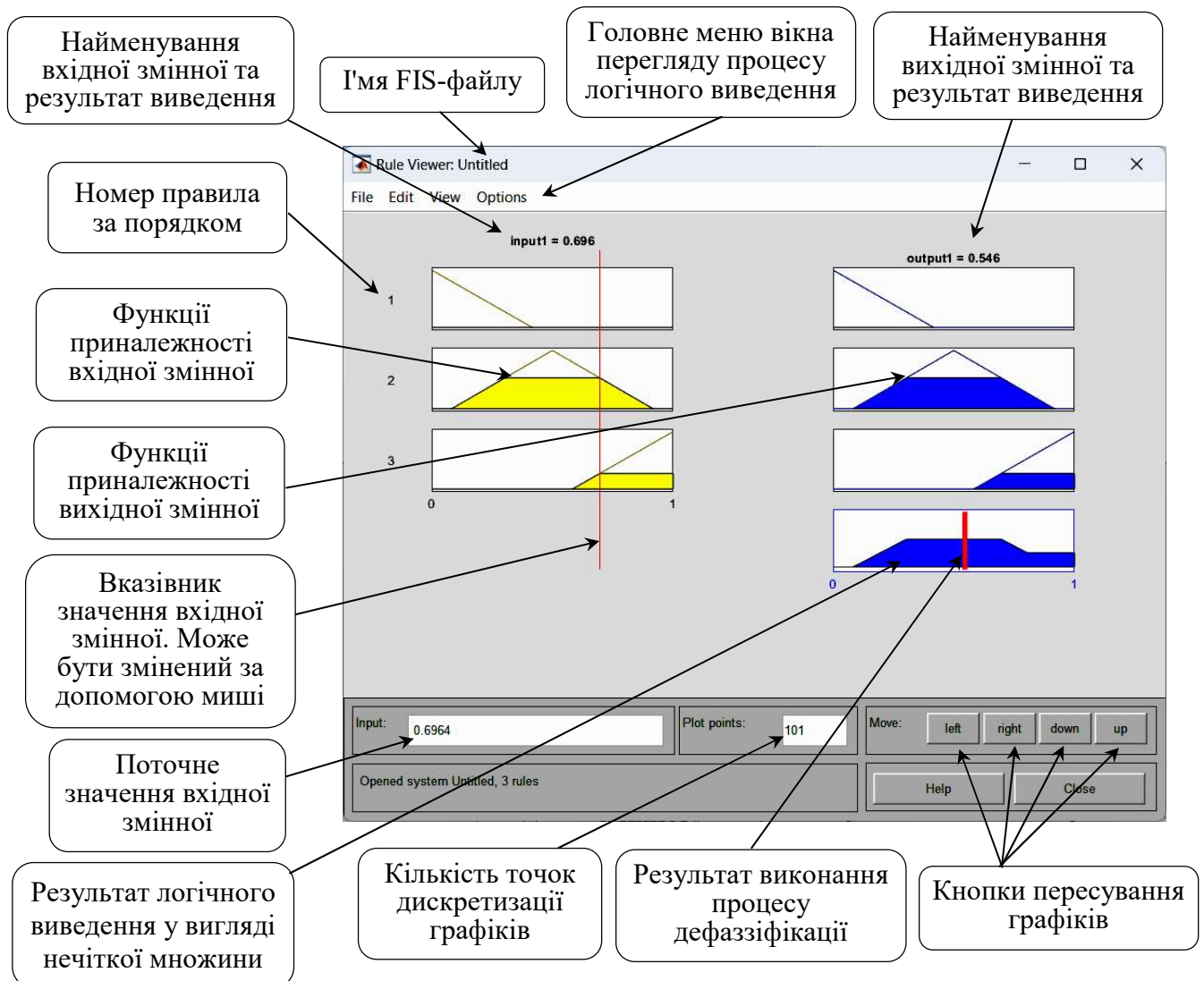


Рисунок 1.27 – Вигляд вікна перегляду процесу нечіткого виведення

Загальний вигляд редактора перегляду (візуалізації) процесу нечіткого логічного виведення із зазначенням функціонального призначення основних полів графічного вікна зображено на рис. 1.27.

У верхній частині вікна перегляду процесу нечіткого логічного виведення розташований рядок, що містить ім'я fis-файлу (**Rule Viewer: Untitled**). Нижче знаходиться головне меню, що містить чотири пункти (**File, Edit, View, Options**). Під ним відображено поле з графіками входних та вихідних функцій приналежності, а також результати дефаззифікації.

Кожне правило бази знань представляється у вигляді послідовності горизонтально розташованих прямокутників. При цьому для даної системи три прямокутники у лівій частині вікна відображають функції приналежності термів-посилки правила **ЯКЩО**, а група з трьох прямокутників у правій частині вікна відповідає функції приналежності терма-наслідку вихідної змінної (правило **ТО**). Четвертий прямокутник у правій частині вікна графічного інтерфейсу відображає процес дефаззифікації вихідної змінної після акумулювання усіх висновків правил нечіткого логічного виведення.

Жовте заливання графіків функцій приналежності входних змінних вказує наскільки значення входів відповідають термам даного правила. Для виведення правила у форматі **Rule Editor** необхідно зробити клік лівою кнопкою миші на номеру відповідного правила. У цьому випадку вказане правило буде виведене у нижній частині вікна.

Блакитне заливання графіка функції приналежності вихідної змінної є результатом логічного виведення у вигляді нечіткої множини за цим правилом. Результируюча нечітка множина, що відповідає логічному виведення за всіма правилами, показана у нижньому прямокутнику останнього стовпця графічного вікна. У цьому прямокутнику червона вертикальна лінія відповідає чіткому значенню логічного виведення, що отримано у результаті дефаззифікації.

Додавання значень входних змінних може здійснюватися двома способами: шляхом введення чисельних значень у полі (**Input**) та за допомогою миші шляхом переміщення лінії покажчика червоного кольору. В останньому випадку необхідно позиціонувати курсор миші на червоній вертикальній лінії, натиснути ліву клавішу і, не відпускаючи її, перемістити покажчик до потрібної позиції. Нове чисельне значення відповідної входної змінної буде перераховано автоматично та виведено у вікні (**Input**) (на рис. 1.27 значення **[0.668]**).

У полі (**Plot points**) зазначається кількість точок дискретизації для побудови графіків функцій приналежності. Задане значення за замовчуванням – **101**.

У нижній правій частині вікна розташовано поле (**Move**) з кнопками переміщення зображення ліворуч-праворуч (**left-right**) та вгору-вниз (**up-down**).

Крім того, знизу розташовані кнопки (**Help**) та (**Close**), що призначені для виклику довідкової інформації та закриття вікна візуалізації нечіткого логічного виведення відповідно.

Розглянемо головне меню вікна візуалізації процесу нечіткого логічного виведення (**Rule Viewer**). Пункти **File** та **View** однакові для усіх GUI-модулів, що використовуються у нечітких системах. Вони докладно описані під час розгляду FIS-редактора.

EDIT

Загальний вигляд пункту **Edit** головного меню вікна перегляду процесу нечіткого логічного виведення зображений на рис. 1.28. Команда **Undo (Ctrl + Z)** дозволяє користувачеві скасувати раніше скоєну дію. Якщо вона не активна, отже, раніше не було виконано жодної дії.

Undo	Ctrl+Z
FIS Properties...	Ctrl+1
Membership Functions...	Ctrl+2
Rules...	Ctrl+3

Рисунок 1.28 – Пункт головного меню Edit

Команди **FIS Properties... (Ctrl+1)**, **Membership Function... (Ctrl + 2)** та **Rules ... (Ctrl + 3)** відкривають FIS-редактор (рис. 1.3), редактор функцій приналежності (рис. 1.16) та редактор бази нечітких правил (рис.1.21) відповідно.

OPTIONS

Загальний вигляд пункту **Options** головного меню вікна перегляду процесу нечіткого логічного виведення зображений на рис. 1.29. Це меню містить лише одну команду (**Format**), що як і для редактора бази нечітких правил дозволяє встановити формат правил бази знань системи нечіткого логічного виведення (рис. 1.24 – рис. 1.26).

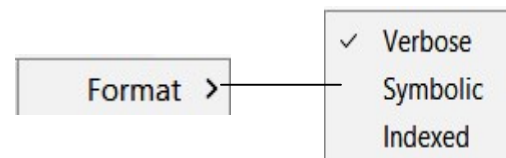


Рисунок 1.29 – Пункт головного меню Options з підпунктом

Список можливих форматів завдання правил бази нечітких знань наступний: (**Verbose**) – лінгвістичний; (**Symbolic**) – логічний; (**Indexed**) – індексований.

1.6. Візуалізація поверхні "входи-вихід" (Surface Viewer)

Візуалізація поверхні "входи-вихід" здійснюється за допомогою GUI-модуля (**Surface Viewer**). Цей модуль дозволяє вивести графічне зображення залежності значення будь-якої вихідної змінної від довільних (однієї або декількох) входних змінних. У першому випадку спостерігається двовірна залежність вихідних змінних від входного сигналу, а у другому тривимірна поверхня.

Вікно (**Surface Viewer**) може бути викликано з будь-якого GUI-модуля, що використовується у системах нечіткого логічного виведення. З FIS-редактора, редактора функцій приналежності, редактора бази нечітких правил та вікна перегляду процесу нечіткого логічного виведення можна викликати командою **Surface (Ctrl + 6)** з пункту **View** головного меню.

Загальний вигляд вікна візуалізації поверхні нечіткого логічного виведення із зазначенням функціонального призначення основних полів графічного вікна зображено на рис. 1.30.

У верхній частині вікна перегляду поверхні "входи-вихід" розташований рядок, що містить ім'я fis-файлу (**Rule Viewer: Untitled**). Нижче знаходиться головне меню, що містить чотири пункти (**File, Edit, View, Options**). Під ним відображається поле з поверхнею "входи-вихід". Зручний графічний інтерфейс дозволяє візуально оцінити увесь процес нечіткого логічного виведення.

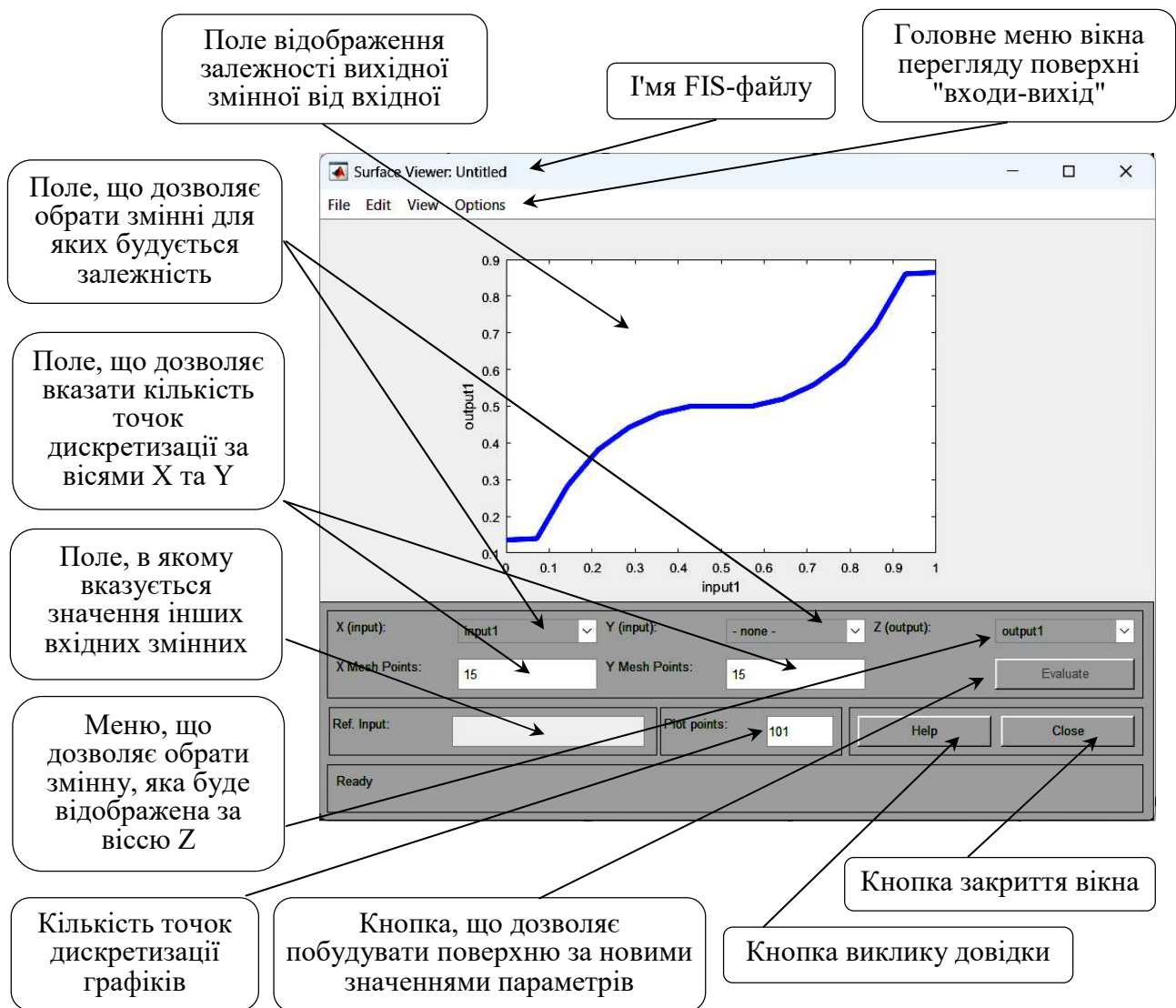


Рисунок 1.30 – Вигляд вікна перегляду поверхні "входи-вихід"

Нижче розташовані поля вибору координатних вісій - **X (input)**, **Y (input)**, **Z (output)**, три поля введення інформації – **X grids**, **Y grids**, **Ref. Input** та кнопка **Evaluate** для побудови поверхні за новими параметрами.

Крім того, знизу розташовані кнопки (**Help**) та (**Close**), що призначені для виклику довідкової інформації та закриття вікна перегляду поверхні "входи-вихід" відповідно.

Поля **X grids** та **Y grids** призначені для завдання кількості точок дискретизації за вісями **X** та **Y** у разі побудови поверхні "входи-вихід". За замовчуванням кількість точок кожної вісі дорівнює **15**. Для зміни цього значення необхідно встановити маркер у полі **X grids** (**Y grids**), ввести нове значення та натиснути клавішу **Enter**.

Поле **Ref. Input** призначено для завдання значень вхідних змінних, крім тих, що асоційовані з координатними вісями. За замовчуванням це значення середини інтервалів змінних. Для зміни цього значення необхідно встановити маркер у полі **Ref. Input** та ввести нове значення.

Меню координатних вісій **X (input)**, **Y (input)**, **Z (output)** дозволяють поставити у відповідність вісям координат вхідні та вихідні змінні. При цьому

вхідні змінні можуть відображатися тільки за вісями **X** та **Y**, а вихідні змінні тільки за віссю **Z**. У **Surface Viewer** є можливість побудови однофакторних залежностей "вхід-вихід". Для цього в меню другої координатної вісі (**X (input)** або **Y (input)**) необхідно обрати **none** (рис. 1.30).

Графічне відображення поверхні "входи-вихід" може мати як двовимірну залежність (показана на рис. 1.30), так і тривимірну, якщо кількість входів системи більше одиниці. У випадку, коли нечітка система містить два входи, маємо графік поверхні, що відображає залежність вихідної величини від двох вхідних. На рис. 1.31 зображені вікна всіх розглянутих раніше редакторів системи нечіткого логічного виведення для двох вхідних змінних.

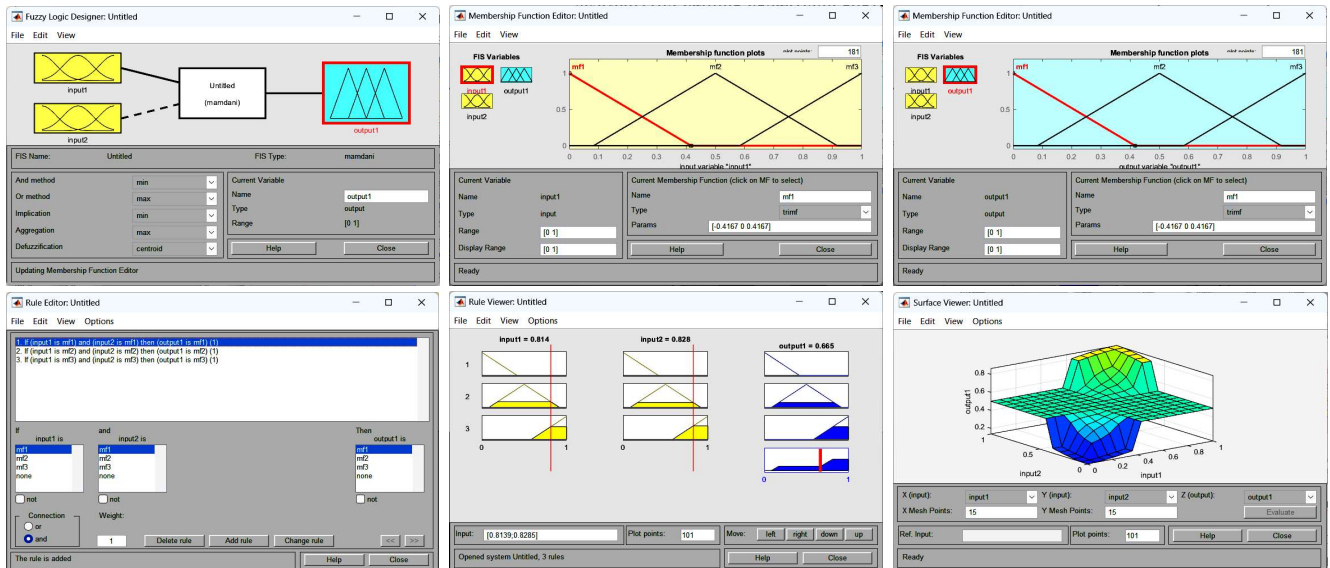


Рисунок 1.31 – Процес нечіткого логічного виведення для системи з двома входами

Surface Viewer дозволяє обертати поверхню "входи-вихід" за допомогою миші. Для цього необхідно позиціонувати курсор миші на поверхні "входи-вихід", натиснути ліву клавішу і, не відпускаючи її, повернути графічне зображення на потрібний кут.

Розглянемо головне меню вікна поверхні "входи-вихід" (**Surface Viewer**). Пункти **File** та **View** однакові для всіх GUI-модулів, що використовуються у системах нечіткого логічного виведення. Вони докладно описані під час розгляду FIS-редактора.

Пункт головного меню **Edit** такий самий, як і для вікна перегляду процесу нечіткого логічного виведення (рис. 1.28).

OPTIONS

Загальний вигляд пункту **Options** головного меню вікна перегляду поверхні "входи-вихід" нечіткого логічного виведення зображений на рис. 1.32. Це меню містить три команди: **Plot**, **Color Map**, **Always evaluate**.

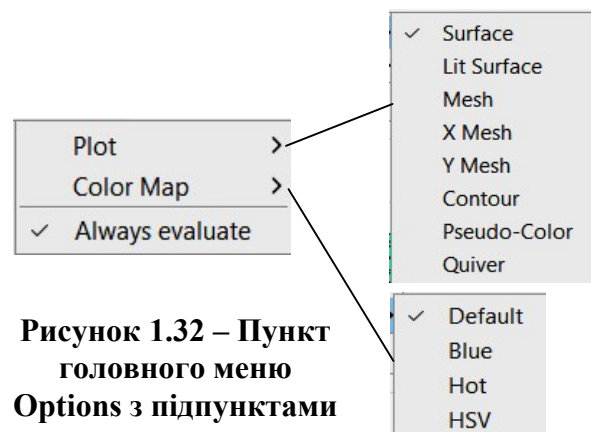


Рисунок 1.32 – Пункт головного меню Options з підпунктами

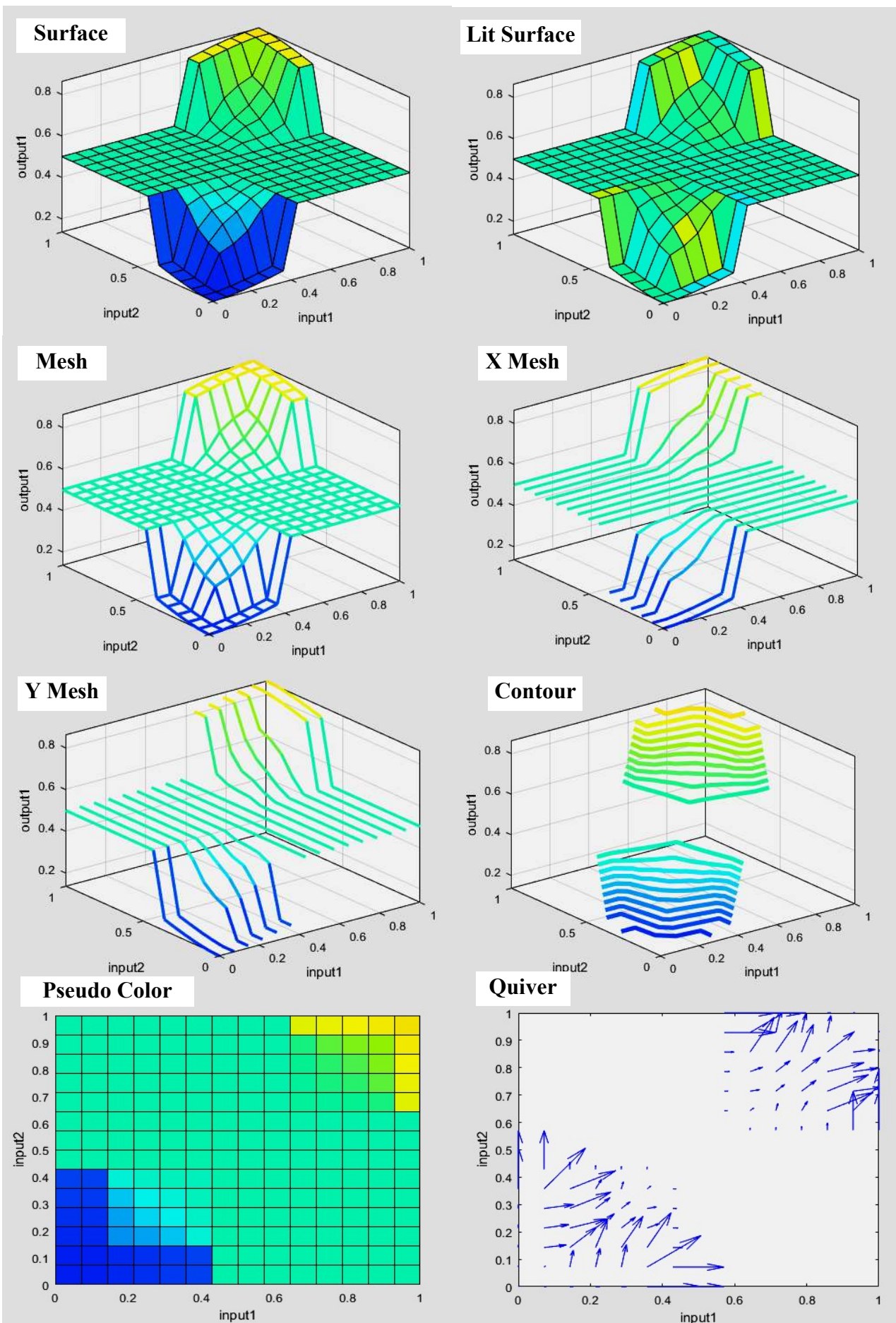


Рисунок 1.33 – Приклади форматів поверхні "входи-вихід"

Команда **Plot** дозволяє керувати форматом виведення поверхні "входи-вихід". Після вибору цієї команди з'являється меню, у якому потрібно обрати формат виведення поверхні. На рис. 1.33 зображено поверхні "входи-вихід" системи (рис. 1.31) для всіх форматів, що підтримуються.

Команда **Color Map** дозволяє керувати палітрою кольорів під час виведення поверхні "входи-вихід". У разі вибору цієї команди з'являється меню, в якому необхідно обрати одну з палітр:

Default – використовувати палітру, встановлену за замовчуванням;

Blue – холодна синьо-блакитна палітра;

Hot – тепла палітра, що складається з чорного, червоного, жовтого та білого кольорів;

HSV – палітра насичених кольорів: червоний, жовтий, зелений, ціан, блакитний, помаранчевий.

Команда **Always evaluate** дозволяє встановити (скасувати) режим автоматичного, тобто без натискання кнопки **Evaluate**, перемальовування поверхні "входи-вихід" у разі будь-якої зміни параметрів.

2. Реалізація одноканальної системи fuzzy-керування типу Mamdani

За допомогою FIS-редактора у пакеті прикладних програм Fuzzy Logic Toolbox можемо синтезувати одноканальну систему fuzzy-керування (fuzzy-контролер) типу **Mamdani** з подальшою інтеграцією у модуль **Simulink**.

У якості вхідної змінної приймемо сигнал **ПОХИБКИ** регулювання, а у якості вихідної – сигнал **КОЕФІЦІЄНТА** передачі.

Функції приналежності вхідного та вихідного сигналу визначимо трьома однаковими трикутними нечіткими множинами: **МАЛИЙ (Small-Ms)**, **СЕРЕДНІЙ (Middle-Mm)**, **ВЕЛИКИЙ (Big-Mb)**.

У якості нечітких правил побудови fuzzy-контролера будемо вважати наступні логічні міркування:

якщо ПОХИБКА МАЛА, то КОЕФІЦІЄНТ МАЛИЙ;
якщо ПОХИБКА СЕРЕДНЯ, то КОЕФІЦІЄНТ СЕРЕДНІЙ;
якщо ПОХИБКА ВЕЛИКА, то КОЕФІЦІЄНТ ВЕЛИКИЙ.

Дефазифікацію здійснимо "гравітаційним" методом (методом "центру ваги" нечітких множин).

Для завантаження вікна головного FIS-редактору необхідно виконати команду **fuzzy** у рядку вікна команд системи Matlab. У вікні (**Fuzzy Logic Designer**) нової системи типу **Mamdani**, що зображена на рис. 2.1 перейменуємо по черзі вхідну та вихідну змінні. Для змінної входу у поле **Current Variable** у вікні **Name** введемо ім'я **ПОХИБКА**, а для змінної виходу – ім'я **КОЕФІЦІЄНТ**. Підтвердження будь-якого введення здійснюємо натисканням клавіші **Enter**.

Перемикання між вхідною та вихідною змінною здійснюється одноразовим кліком лівої клавіші миші на відповідному блоці функціональної схеми системи нечіткого логічного виведення у верхній частині редактора. Після перейменування змінних під функціональними блоками з'являються нові назви.

Задамо ім'я нечіткої системи, для чого у пункті **File** головного меню FIS-редактора оберемо у підменю **Export** команду **To File...** та у діалоговому вікні збереження, що з'явилося, оберемо диск та папку, куди бажано зберегти модель та введемо ім'я системи, наприклад, **fuzzy_controller_m**. Синтезована система буде збережена на диск у файлі з розширенням **.fis**.

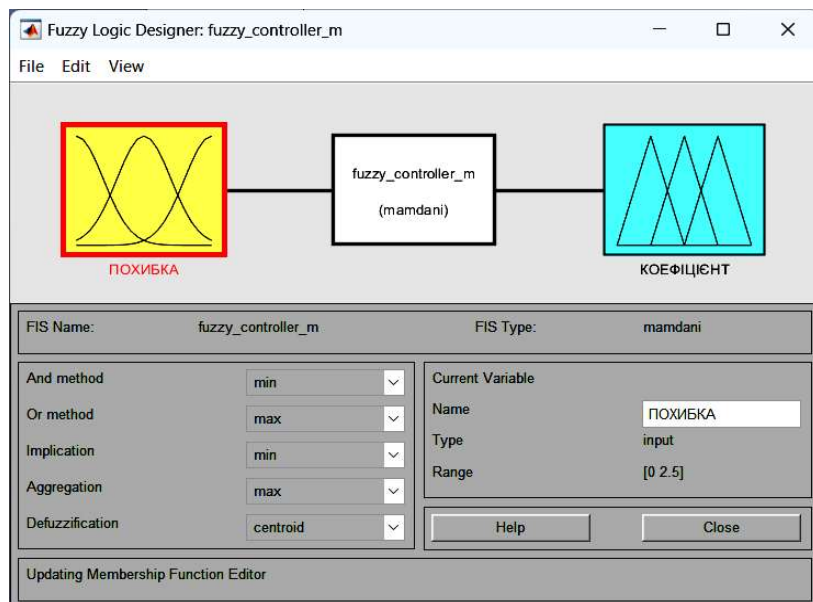


Рисунок 2.1 – Вікно FIS-редактора типу Mamdani

Перейдемо до редактора функцій приналежності (**Membership Function Editor**), для чого двічі клікнемо лівою кнопкою миші на відповідному функціональному блоці (для вхідної змінної на блоці **ПОХИБКА**, для вихідної – на блоці **КОЕФІЦІЄНТ**) або скористаємося пунктом головного меню FIS-редактора **Edit**, в якому виконаємо **команду Membership Functions...** або натиснемо комбінацію клавіш **Ctrl +2**.

У вікні редагування вхідної функції приналежності (рис. 2.2) задамо діапазон зміни змінної **ПОХИБКА**, для чого введемо у поле **Range** вектор діапазону **[0 2.5]**. У полі **Display Range** також введемо діапазон **[0 2.5]**, на який буде поділена вісь абсцис.

Задамо функції приналежності змінної **ПОХИБКА**. Для лінгвістичної оцінки цієї змінної будемо використовувати три терми з трикутними функціями приналежності.

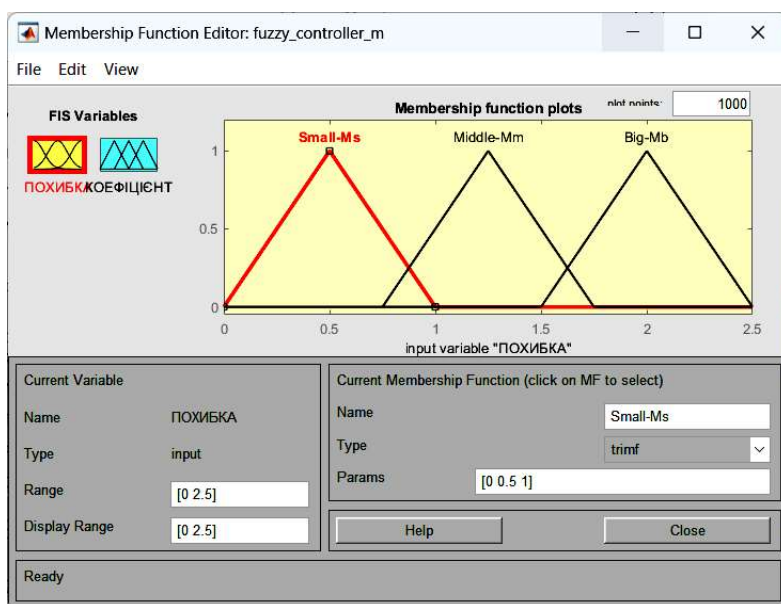


Рисунок 2.2 – Вікно редактора налаштувань вхідної функції приналежності

Для цього можна налаштувати наявні терми або спочатку видалити всі наявні командою **Remove All MF**, а потім додати необхідну кількість командою **Add MFs...** з пункту головного меню **Edit** редактора функцій приналежності. Після додавання відповідних термів з'явиться вікно налаштувань, в якому потрібно вказати кількість термів у полі **Number of MFs** та тип функції приналежності у полі **Type** (для нашого завдання оберемо три трикутні (**trimf**) функції приналежності).

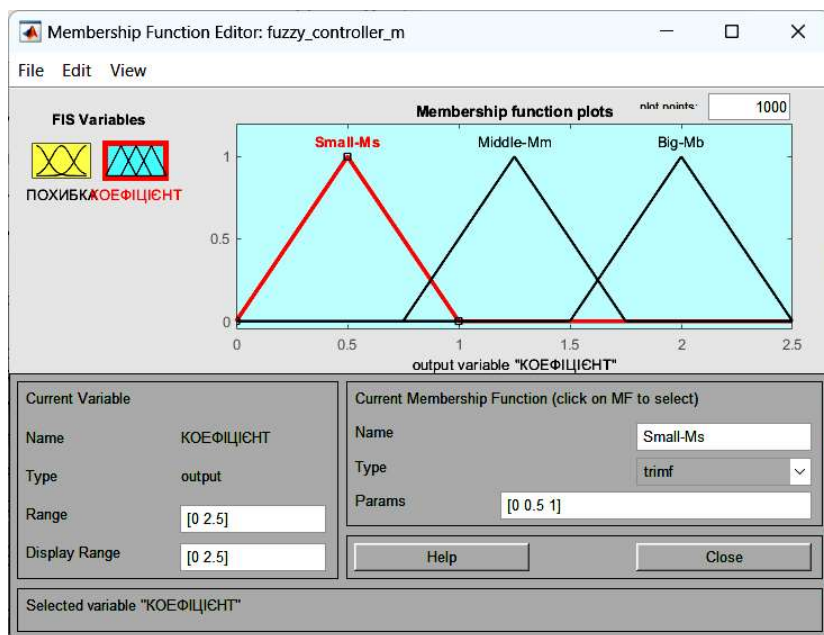


Рисунок 2.3 – Вікно редактора налаштувань вихідної функції приналежності

У полі **Type** показано тип функції приналежності (у нашому прикладі це **trimf**), який також можна змінити зі спливаючого списку стандартних функцій. Нижче, у полі **Params** задамо чисельні значення координат основи та вершини трикутника. Для функції приналежності **Small-Ms** вектор координат задамо **[0 0.5 1]**. Виконаємо цю ж процедуру для решти двох термів. Другий терм, що відповідає **СЕРЕДНІЙ ПОХИБЦІ**, назвемо **Middle-Mm**, для нього координати задамо таким вектором **[0.75 1.25 1.75]**. Третій терм, що відповідає **ВЕЛИКІЙ ПОХИБЦІ**, перейменуємо як **Big-Mb** з координатами **[1.5 2 2.5]**. Крім того, у полі **plot points** редактора змінимо кількість точок дискретизації (**1000**) для покращення візуального відображення графічного елементу.

Аналогічну процедуру проведемо і для функцій приналежності сигналу виходу нечіткої системи **КОЕФІЦІЄНТ**. Для переходу у вікно редактора вихідної функції приналежності (рис. 2.3), двічі клікнемо лівою кнопкою миші на відповідному блоці у лівій частині вікна (рис. 2.2) або на блоці у правій частині функціональної схеми (рис. 2.1).

Після закінчення редагування параметрів функцій приналежності перейдемо до редактора бази нечітких правил (**Rule Editor**), для цього скористаємося командою **Rules...** з пункту **Edit** головного меню редактора функцій приналежності або натиснемо комбінацію клавіш **Ctrl + 3**, або двічі клікнемо лівою кнопкою миші на функціональному блоці (**fuzzy_controller_m (mamdani)**) на схемі у FIS-редакторі (рис. 2.1).

У вікні редактора бази нечітких правил (рис. 2.4) на основі візуального спостереження за графіком, сформулюємо та введемо наступні три логічні правила:

якщо **ПОХИБКА Small-Ms**, то **КОЕФІЦІЄНТ Small-Ms**;
якщо **ПОХИБКА Middle-Mm**, то **КОЕФІЦІЄНТ Middle-Mm**;
якщо **ПОХИБКА Big-Mb**, то **КОЕФІЦІЄНТ Big-Mb**.

Після появи відповідних термів у вікні графічного відображення задамо їх найменування. Спочатку виділимо кліком лівої кнопки миші перший терм і у полі **Current Membership Function** (click on MF to select) введемо його ім'я (**Small-Ms**), що відповідає **МАЛІЙ ПОХИБЦІ** та після введення натиснемо клавішу **Enter**. Після цього назва терму відобразиться над відповідним графічним зображенням.

Для введення цих нечітких правил обираємо послідовно у лівому вікні правила-посилки **ЯКЩО (If)** та у правому вікні правила-наслідку **ТО (Then)** відповідні назви термів одноразовим кліком миші та для підтвердження кожного правила натискаємо кнопку **Add rule...** У разі помилково введеного правила його можна виділити та видалити, натиснувши на кнопку **Delete rule**.

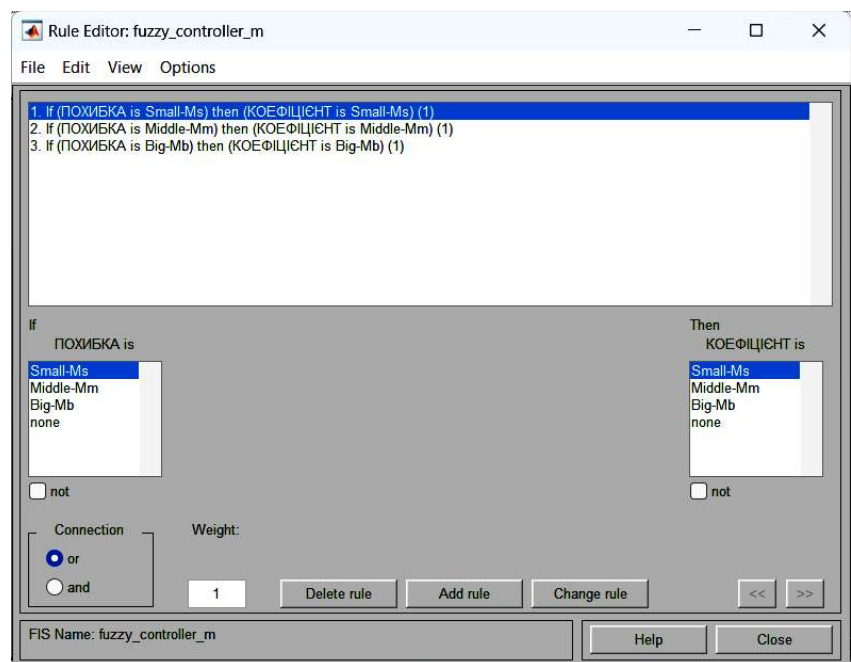


Рисунок 2.4 – Вікно редактора нечітких правил

Вагові коефіцієнти кожного правила, що показані у дужках та у полі **Weight**, залишимо рівними одиниці (1).

На цьому синтез fuzzy-системи закінчено, тому ще раз збережемо її на диску, виконавши команду **To File...** з підменю **Export** пункту **File** головного меню будь-якого редактору.

На завершення відобразимо увесь процес нечіткого логічного виведення, для чого відкриємо вікно візуалізації fuzzy-системи командою **Rules (Ctrl + 5)** з пункту **View** головного меню FIS-редактора, редактора функцій приналежності або редактора бази нечітких правил. Процес нечіткого логічного виведення зображено на рис. 2.5.

Кожне нечітке правило представляється у якості послідовностей горизонтально розташованих прямокутників. При цьому для даної системи три прямокутники у лівій частині вікна відображають трикутні функції приналежності термів-посилки правила **ЯКЩО**, а група з трьох прямокутників у правій частині вікна відповідає функції приналежності термів-наслідку вихідної змінної (правило **ТО**).

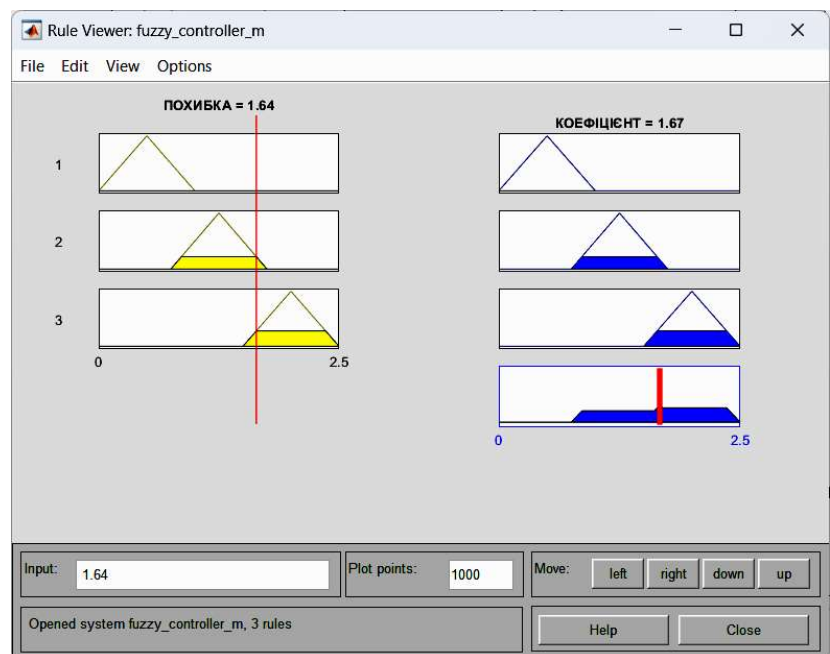


Рисунок 2.5 – Вікно перегляду ходу нечіткого логічного виводу

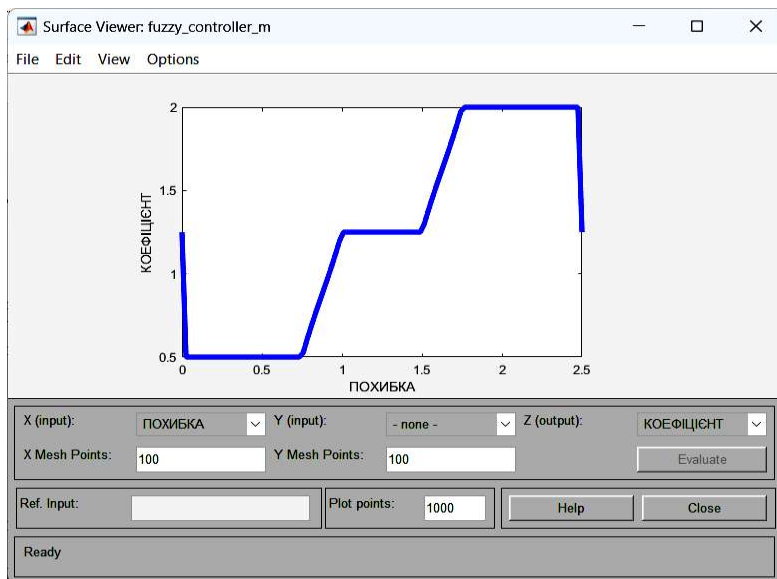


Рисунок 2.6 – Вікно перегляду поверхні
"входи-вихід"

На рис. 2.6 зображений графік залежності вихідної величини **КОЕФІЦІЄНТ** від вхідної **ПОХИБКА**. Для покращення відображення графіку введемо у полях **X grids** та **Y grids** значення (**100**). Отриманий графік відображає процес дефаззифікації даної системи гравітаційним методом (**centroid**), який на рівні з іншими параметрами процесу нечіткого логічного виведення, вказані за замовчуванням у вікні FIS-редактора (рис. 2.1). Цей графік відображає поведінку вихідної координати **КОЕФІЦІЄНТ**, що змінюється у разі лінійного зростання сигналу **ПОХИБКИ**.

3. Реалізація одноканальної системи fuzzy-керування типу Sugeno

За допомогою FIS-редактора у пакеті прикладних програм Fuzzy Logic Toolbox можна синтезувати одноканальну систему fuzzy-керування (fuzzy-контролер) типу **Sugeno** з подальшою інтеграцією у модуль **Simulink**.

У якості вхідної змінної приймемо сигнал **ПОХИБКИ** регулювання, а у якості вихідної – сигнал **КОЕФІЦІЄНТА** передачі.

Функції приналежності вхідного та вихідного сигналу визначимо трьома однаковими трикутними нечіткими множинами **МАЛИЙ (Small-Ms)**, **СЕРЕДНІЙ (Middle-Mm)**, **ВЕЛИКИЙ (Big-Mb)**.

Вихідний сигнал контролера представимо у вигляді трьох постійних чисельних величин, що є координатами центрів ваги відповідних термів вхідної функції приналежності з лінгвістичними іменами: **МАЛИЙ**, **СЕРЕДНІЙ**, **ВЕЛИКИЙ**.

У якості нечітких правил побудови fuzzy-контролера будемо вважати наступні логічні міркування:

якщо ПОХИБКА МАЛА, то КОЕФІЦІЄНТ МАЛИЙ;
якщо ПОХИБКА СЕРЕДНЯ, то КОЕФІЦІЄНТ СЕРЕДНІЙ;
якщо ПОХИБКА ВЕЛИКА, то КОЕФІЦІЄНТ ВЕЛИКИЙ.

Четвертий прямокутник у правій частині вікна графічного інтерфейсу відображає процес дефаззифікації вихідної змінної після акумулювання всіх висновків правил нечіткого логічного виведення. Переміщуючи вертикальну червону лінію-показчик можна бачити увесь процес фаззифікації та дефаззифікації візуально у інтерактивному режимі. У вікні **Input** відображається поточне значення вхідної змінної.

У полі **Plot points** вкажемо кількість точок дискретизації відображення графіків (**1000**).

Дефаззифікацію виконаємо методом "зваженого середнього значення" нечітких множин (**wtaver**).

Для завантаження вікна головного FIS-редактору виконаємо команду **fuzzy** у рядку вікна команд системи Matlab. У вікні (**Fuzzy Logic Designer**) нової системи типу **Mamdani**, у пункті головного меню **File** оберемо команду **New FIS** та вкажемо тип системи **Sugeno**. У вікні нової системи, що зображено на рис. 3.1, перейменуємо по черзі вхідну та вихідну змінні. Для змінної входу у полі **Current Variable** у вікні **Name** введемо ім'я **ПОХИБКА**, а для змінної виходу – ім'я **КОЕФІЦІЄНТ**.

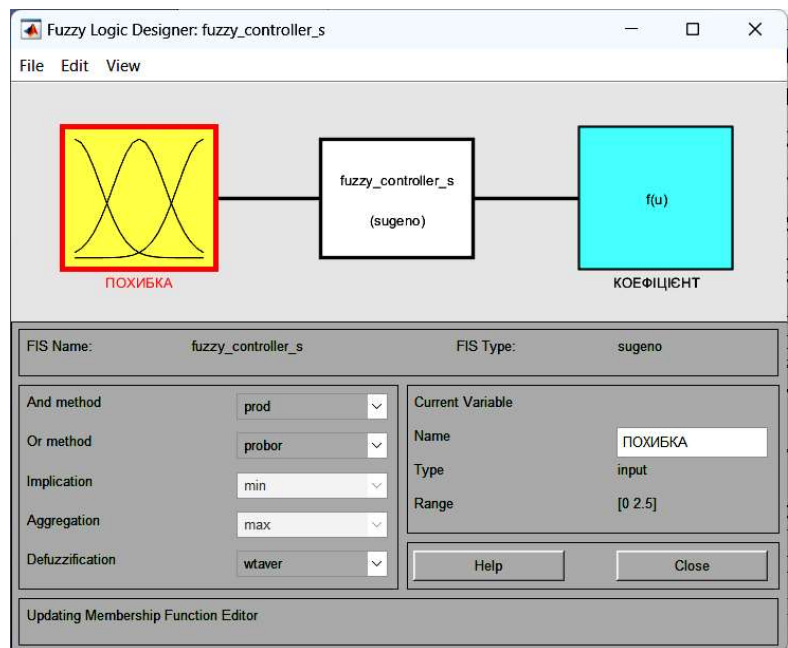


Рисунок 3.1 – Вікно FIS-редактора типу Sugeno

Підтвердження будь-якого введення здійснюємо натисканням клавіші **Enter**.

Перемикання між вхідною та вихідною змінною здійснюється кліком лівої клавіші миші на відповідному блоці функціональної схеми системи нечіткого логічного виведення у верхній частині редактора. Після перейменування змінних під функціональними блоками з'являться нові назви.

Задамо ім'я нечіткої системи, для чого у пункті **File** головного меню FIS-редактора оберемо у підменю **Export** команду **To File...** та у діалоговому вікні збереження, що з'явилося, оберемо диск і папку, куди бажано зберегти модель та введемо ім'я системи, наприклад **fuzzy_controller_s**. Синтезована система буде збережена на диск у файл з розширенням **.fis**.

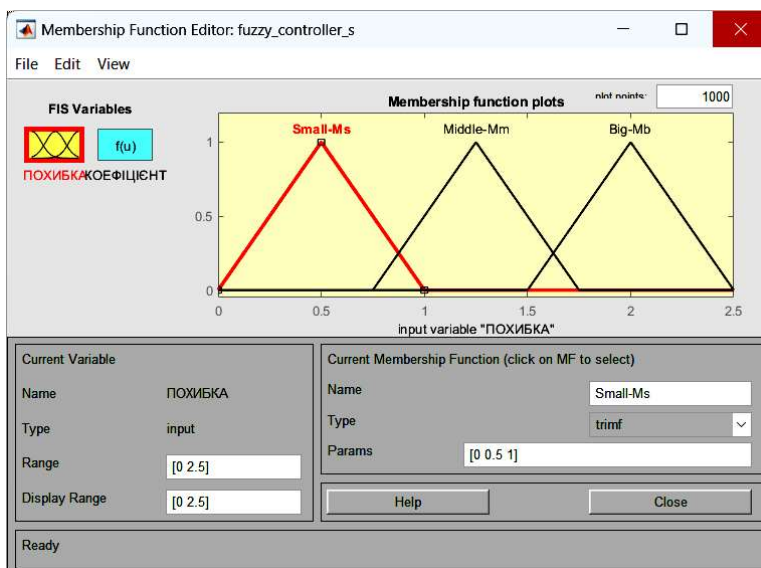


Рисунок 3.2 – Вікно редактора налаштувань вхідної функції приналежності

Перейдемо до редактора функцій приналежності (**Membership Function Editor**), для чого двічі клікнемо лівою кнопкою миші на відповідному функціональному блоці (для вхідної змінної на блоці **ПОХИБКА**, для вихідної – на блоці **КОЕФІЦІЄНТ**) або скористаємося пунктом головного меню FIS-редактора **Edit**, в якому виконаємо **команду Membership Functions...** або натисимо комбінацію клавіш **Ctrl + 2**.

У вікні редагування вхідної функції приналежності (рис. 3.2) задамо діапазон зміни змінної **ПОХИБКА**, для чого введемо у поле **Range** вектор діапазону **[0 2.5]**. У полі **Display Range** також введемо діапазон **[0 2.5]**, на який буде поділена вісь абсцис. Задамо функції приналежності змінної **ПОХИБКА**. Для лінгвістичної оцінки цієї змінної будемо використовувати три терми з трикутними функціями приналежності.

Для цього можна налаштувати наявні терми або спочатку видалити всі наявні командою **Remove All MF**, а потім додати необхідну кількість командою **Add MFs...** з пункту головного меню **Edit** редактора функцій приналежності. У разі додавання відповідних термів з'явиться вікно налаштувань, в якому потрібно вказати кількість термів у полі **Number of MFs** та тип функції приналежності у полі **Type** (для нашого завдання оберемо три трикутні (**trimf**) функції приналежності).

Після появи відповідних термів у вікні графічного відображення задамо їх найменування. Спочатку виділимо одноразовим кліком лівої кнопки миші перший терм і у полі **Current Membership Function (click on MF to select)** введемо його ім'я (**Small-Ms**), що відповідає **МАЛІЙ ПОХИБЦІ** та після введення натиснемо клавішу **Enter**. Після цього назва терму відобразиться над відповідним графічним зображенням.

У полі **Type** показано тип функції приналежності (у нашому прикладі це **trimf**), який також можна змінити зі спливаючого списку стандартних функцій. Нижче, у полі **Params** задамо чисельні значення координат основи та вершини трикутника. Для функції приналежності **Small-Ms** вектор координат задамо **[0 0.5 1]**. Виконаємо цю ж процедуру для решти двох термів. Другий терм, що відповідає **СЕРЕДНІЙ ПОХИБЦІ**, назовемо **Middle-Mm**, для нього координати задамо таким вектором **[0.75 1.25 1.75]**. Третій терм, що відповідає **ВЕЛИКІЙ ПОХИБЦІ**, перейменуємо як **Big-Mb** з координатами **[1.5 2 2.5]**. Крім того, у полі **plot points** редактора змінимо кількість точок дискретизації (**1000**) для покращення візуального відображення графіку.

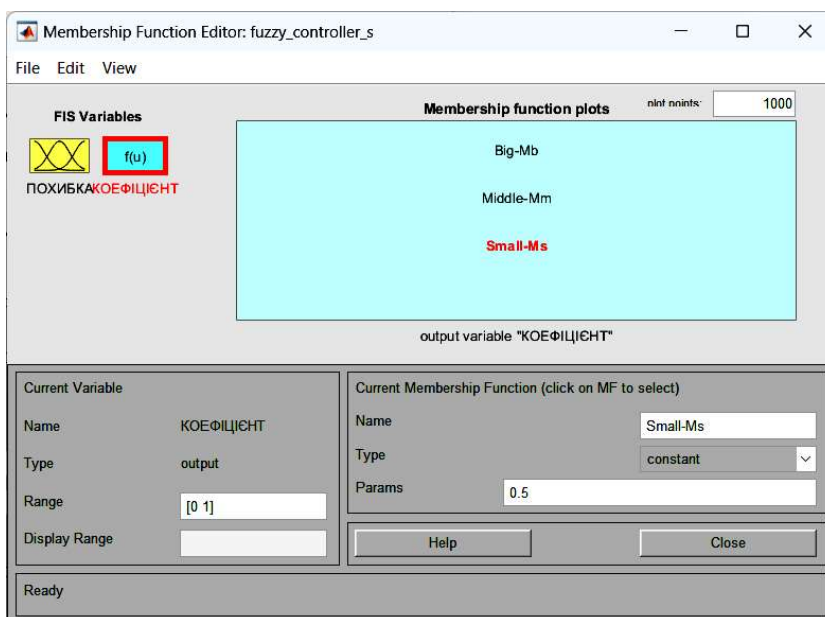


Рисунок 3.3 – Вікно редактора налаштувань вихідної функції приналежності

Аналогічну процедуру проведемо і для функцій приналежності сигналу виходу нечіткої системи **КОЕФІЦІЄНТ**. Для переходу у вікно редактора вихідної функції приналежності (рис. 3.3), двічі клікнемо лівою кнопкою миші на відповідному блоці у лівій частині вікна (рис. 3.2) або на блоці у правій частині функціональної схеми (рис. 3.1).

Для систем типу **Sugeno** у полі **Type** вкажемо тип функції приналежності (у нашому прикладі це **constant**), який можна змінити на лінійний (**linear**) зі спливаючого списку стандартних функцій.

Задамо діапазон зміни змінної **КОЕФІЦІЄНТ**, для чого надрукуємо у полі **Range** вектор діапазону **[0 2.5]**.

У полі **Params** введемо чисельні значення констант. Для функції приналежності, що відповідає **МАЛОМУ КОЕФІЦІЄНТУ**, яку назвемо **Small-Ms**, введемо значення константи **0.5**. Для функції приналежності, що відповідає **СЕРЕДНЬОМУ КОЕФІЦІЄНТУ**, яку назвемо **Middle-Mm**, введемо значення константи **1.25**. Для функції приналежності, що відповідає **ВЕЛИКОМУ КОЕФІЦІЄНТУ**, яку назвемо **Big-Mb**, введемо значення константи **2**.

Крім того, у полі **plot points** редактора змінимо кількість точок дискретизації (**1000**) для покращення візуального відображення графіку.

Після закінчення редагування параметрів функцій приналежності перейдемо до редактора бази нечітких правил (**Rule Editor**), для цього скористаємося командою **Rules...** з пункту **Edit** головного меню редактора функцій приналежності або натиснемо комбінацію клавіш **Ctrl+3**, або двічі клікнемо лівою кнопкою миші на функціональному блоці (**fuzzy_controller_s (sugeno)**) на схемі у FIS-редакторі (рис. 3.1).

У вікні редактора бази нечітких правил (рис. 3.4) на основі візуального спостереження за графіком, сформулюємо та введемо наступні три логічні правила:

якщо ПОХИБКА Small-Ms, то КОЕФІЦІЄНТ Small-Ms;
якщо ПОХИБКА Middle-Mm, то КОЕФІЦІЄНТ Middle-Mm;
якщо ПОХИБКА Big-Mb, то КОЕФІЦІЄНТ Big-Mb.

Для введення цих нечітких правил обираємо послідовно у лівому вікні правила-посилки **ЯКЩО (If)** та у правому вікні правила-наслідку **ТО (Then)** відповідні назви термів одноразовим кліком миші та для підтвердження кожного правила натискаємо кнопку **Add rule...** У разі помилково введеного правила його можна виділити та видалити, натиснувши на кнопку **Delete rule**.

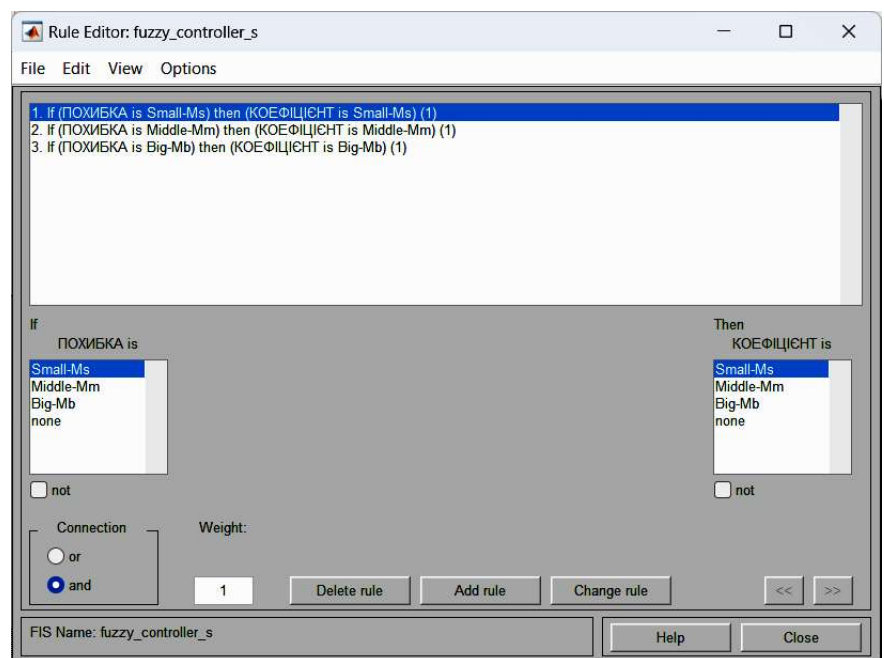


Рисунок 3.4 – Вікно редактора нечітких правил

Вагові коефіцієнти кожного правила, що показані у дужках та у полі **Weight**, залишимо рівними одиниці (**1**).

На цьому синтез fuzzy-системи закінчено, тому ще раз збережемо її на диску, виконавши команду **To File...** з підменю **Export** пункту **File** головного меню будь-якого редактору.

На завершення відобразимо увесь процес нечіткого логічного виведення, для чого відкриємо вікно візуалізації fuzzy-системи командою **Rules (Ctrl + 5)** з пункту **View** головного меню FIS-редактора, редактора функцій приналежності або редактора бази нечітких правил. Процес нечіткого логічного виведення зображено на рис. 3.5.

Кожне нечітке правило представляється у якості послідовності горизонтально розташованих прямокутників. При цьому для даної системи три прямокутники у лівій частині вікна відображають трикутні функції приналежності термів-посилки правила **ЯКЩО**, а група з трьох прямокутників у правій частині вікна відповідає функції приналежності термів-наслідку вихідної змінної (правило **ТО**).

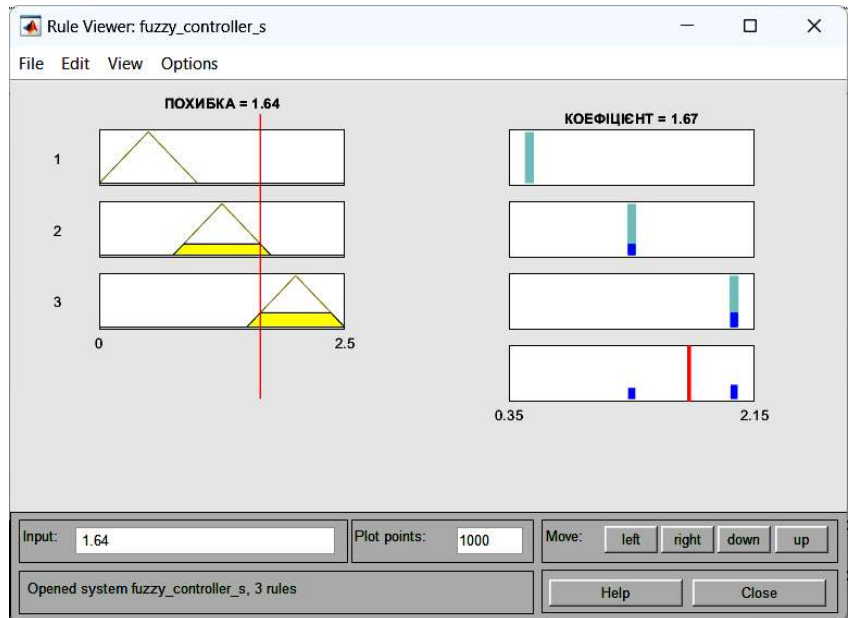


Рисунок 3.5 – Вікно перегляду ходу нечіткого логічного виведення

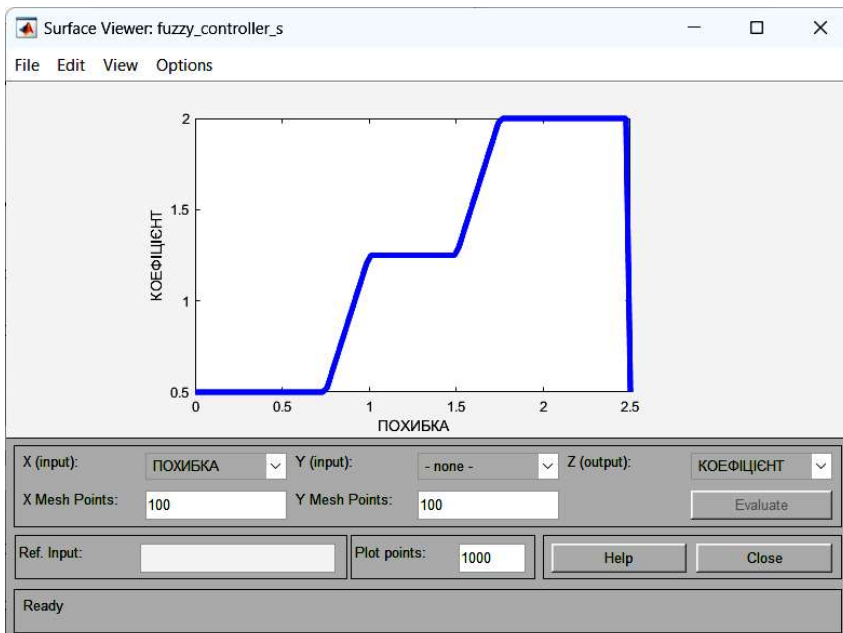


Рисунок 3.6 – Вікно перегляду поверхні "входи-вихід"

У полі **Plot points** вкажемо кількість точок дискретизації відображення графіків (**1000**).

На рис. 3.6 зображено графік залежності вихідної величини **КОЕФІЦІЄНТ** від вхідної **ПОХИБКА**. Для покращення відображення введемо у полях **X grids**

Четвертий прямокутник у правій частині вікна графічного інтерфейсу відображає процес дефазифікації вихідної змінної після акумулювання усіх висновків правил нечіткого логічного виведення. Переміщуючи вертикальну червону лінію-показчик можна бачити увесь процес фазифікації та дефазифікації візуально у інтерактивному режимі. У вікні **Input** відображається поточне значення вхідної змінної.

та **Y grids** значення (**100**). Отриманий графік відображає процес дефаззифікації даної системи методом зваженого середнього значення (**wtaver**), який, як і інші параметри процесу нечіткого логічного виведення, вказані за замовчуванням у вікні FIS-редактора (рис. 3.1). Цей графік відображає поведінку вихідної координати **КОЕФІЦІЄНТ**, що змінюється під час лінійного зростання сигналу **ПОХИБКИ**.

Необхідно відзначити, що основна відмінність між системами типу Mamdani та Sugeno полягає у різних способах завдання значень вихідної змінної у правилах, що утворюють основу нечітких знань. Якщо у системах типу Mamdani значення вихідної змінної задаються нечіткими термами, то у системах типу Sugeno – як лінійна комбінація вхідних змінних (або стала величина).

Більш детальний аналіз fuzzy-систем цих двох типів свідчить, що модель типу Sugeno точніша. Перевага моделей типу Mamdani полягає у тому, що правила бази знань є прозорими та інтуїтивно зрозумілими, тоді як для моделей типу Sugeno не завжди зрозуміло які лінійні залежності "входи-вихід" необхідно використовувати.

Як буде показано далі у пакеті Fuzzy Logic Toolbox є спеціальна команда, що дозволяє здійснювати швидкий перехід від систем типу Mamdani до систем типу Sugeno. При цьому, що дуже важливо, програма сама перераховує параметри системи у лінійну залежність або постійне значення.

4. Інтеграція fuzzy-системи у додаток Simulink

Інтеграція отриманих систем нечіткого логічного виведення (fuzzy-систем) у додаток **Simulink** виконується однаково як для систем типу Mamdani, так і для систем типу Sugeno.

Для інтеграції системи нечіткого логічного виведення у вигляді блоку fuzzy-контролера у додаток Simulink, потрібно обов'язково експортувати створену модель системи (незалежно від типу Mamdani чи Sugeno) у робочий простір середовища Matlab.

Цей процес здійснюється виконанням команди **To Workspace...** з підменю **Export** пункту головного меню **File** вікна будь-якого редактора. У діалоговому вікні необхідно ввести ідентифікатор (ім'я) fuzzy-системи (зазвичай він приймається таким, під яким ім'ям система була збережена на диску). Натискання кнопки **Ok** експортує систему нечіткого логічного виведення у робочий простір системи Matlab.

Для створення блок-схеми моделі нечіткої системи необхідно відкрити додаток **Simulink**, для чого потрібно натиснути відповідну кнопку у головному меню системи Matlab. З'явиться інтерфейсне вікно додатку у якому треба обрати блок створення нової моделі. З головного меню цього вікна відкривається браузер бібліотек (**Simulink Library Browser**). У бібліотеці необхідно обрати папку **Fuzzy Logic Toolbox**, звідки перенести функціональний блок fuzzy-контролера (**Fuzzy Logic Controller**) у вікно створення нової моделі (рис. 4.1).

Крім функціонального блоку fuzzy-контролера (**Fuzzy Logic Controller**) для візуального представлення результатів синтезу нечіткої системи у вікно створення Simulink-моделі з бібліотеки **Fuzzy Logic Toolbox** папки **Membership Functions** (функції приналежності) перенесемо блок трикутної функції приналежності (**Triangular MF**).

Відкриття додатку Simulink

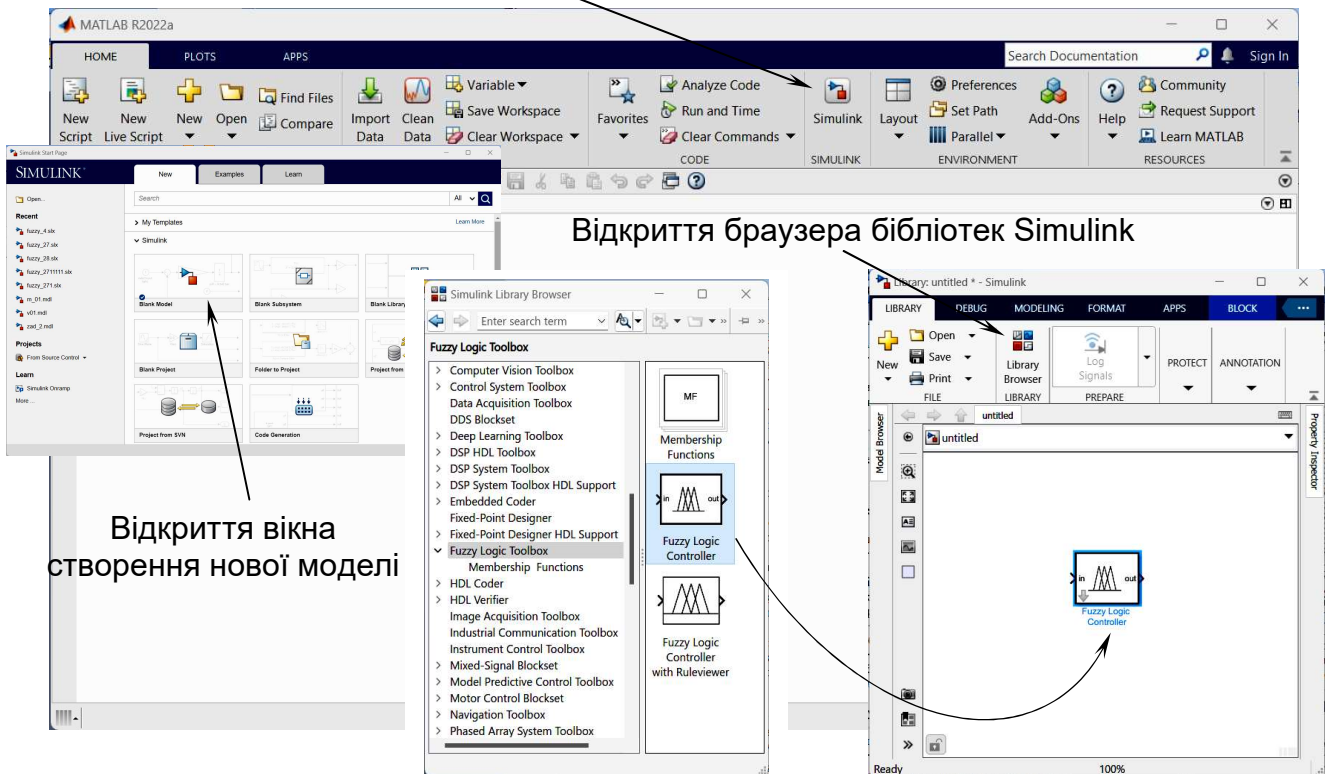


Рисунок 4.1 – Створення блоку fuzzy-контролера у додатку Simulink

Інші функціональні блоки перенесемо з бібліотеки Simulink. З папки джерел сигналів та впливів (**Sources**) перенесемо блок завдання лінійно-зростаючого вхідного сигналу (**Ramp**) для завдання зміни вхідного сигналу – ПОХИБКИ регулювання; з папки віртуальних реєстраторів (**Sinks**) перенесемо блок віртуального осцилографа (**Scope**) для відображення результатів моделювання; з папки компонентів, що підключаються (**Signal & Systems**) перенесемо блок мультиплексора (**Mux**) для відображення необхідних графічних результатів на одному екрані осцилографа. Усі перенесені блоки об'єднаємо функціональними зв'язками у єдину модель системи нечіткого логічного виведення (рис. 4.2).

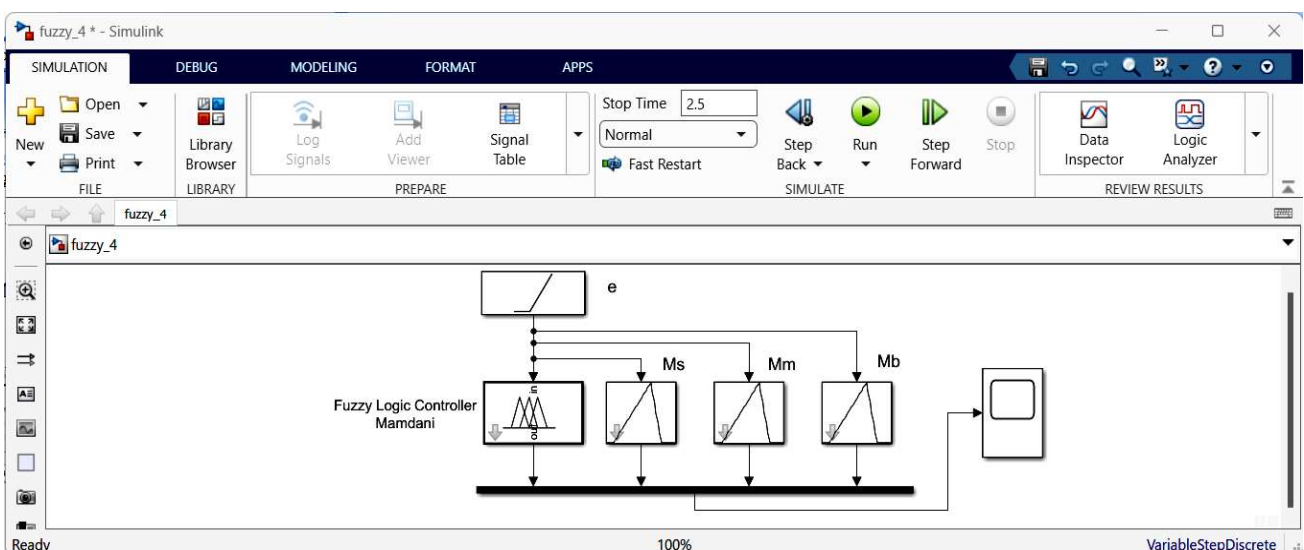


Рисунок 4.2 – Simulink-модель нечіткої системи типу Mamdani

Налаштуємо параметри кожного блоку нечіткої моделі, для чого двічі клікнемо лівою кнопкою миші на необхідному блоці. Для блоку трикутної функції приналежності, у вікні налаштування параметрів вкажемо координати вершини та основ трикутника (рис. 4.3). Для блоку мультиплексора вкажемо число входів – 4. У блоці сигналу завдання визначимо одиничний лінійно-зростаючий сигнал.

Для налаштування fuzzy-контролера у діалоговому вікні (рис. 4.4) необхідно вказати ідентифікатор (ім'я), під яким система з **Fuzzy Logic Toolbox** була експортована до робочої області Matlab (у нашому випадку це **fuzzy_controller_m**).

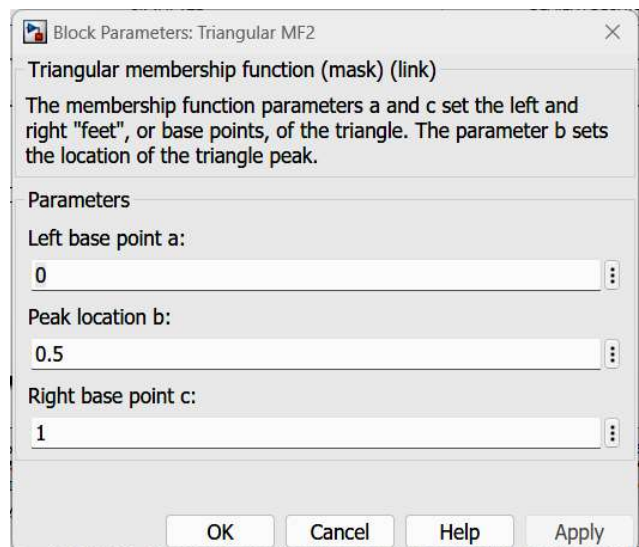


Рисунок 4.3 – Вікно налаштувань параметрів блоку трикутної функції приналежності

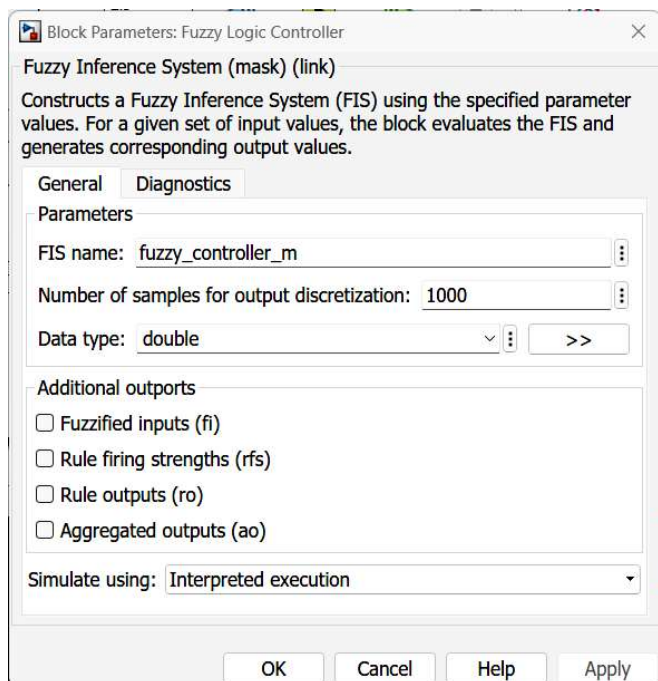


Рисунок 4.4 – Вікно налаштувань параметрів блоку fuzzy-контролера

Потрібно відзначити, що після кожного внесення зміни до fuzzy-системи (зміни параметрів функцій приналежності, нечітких правил тощо) потрібно знову експортувати модель до робочої області системи Matlab.

На завершальному етапі налаштуємо параметри процесу моделювання, для чого кліком правої кнопки миші у просторі моделі викличемо вікно налаштувань, де оберемо команду **Model Configuration Parameters**. У вкладці **Solver** вкажемо час початку моделювання (**Start time**) – 0 та час зупинки моделювання (**Stop time**) – 2.5 віртуальні секунди. Можна також вказати тип кроку та метод моделювання (за замовчуванням **Variable-step** та **auto**).

Після закінчення всіх налаштувань, збережемо модель на диску та запусимо процес моделювання (клікнувши мишею на кнопці **Run** головного меню вікна моделі), у результаті чого на екрані отримаємо графіки залежності функцій приналежності та координати центру ваги (дефаззифікації) від вхідного сигналу після відповідного форматування графіків у вікні **Scope** (рис. 4.5).

Результати моделювання свідчать про те, що графіки отримані у модулі **Simulink** ідентичні графікам з додатку **Fuzzy Logic Toolbox**, що характеризує правильність синтезу fuzzy-контролера та доцільність використання середовища Matlab під час моделювання систем нечіткого логічного виведення.

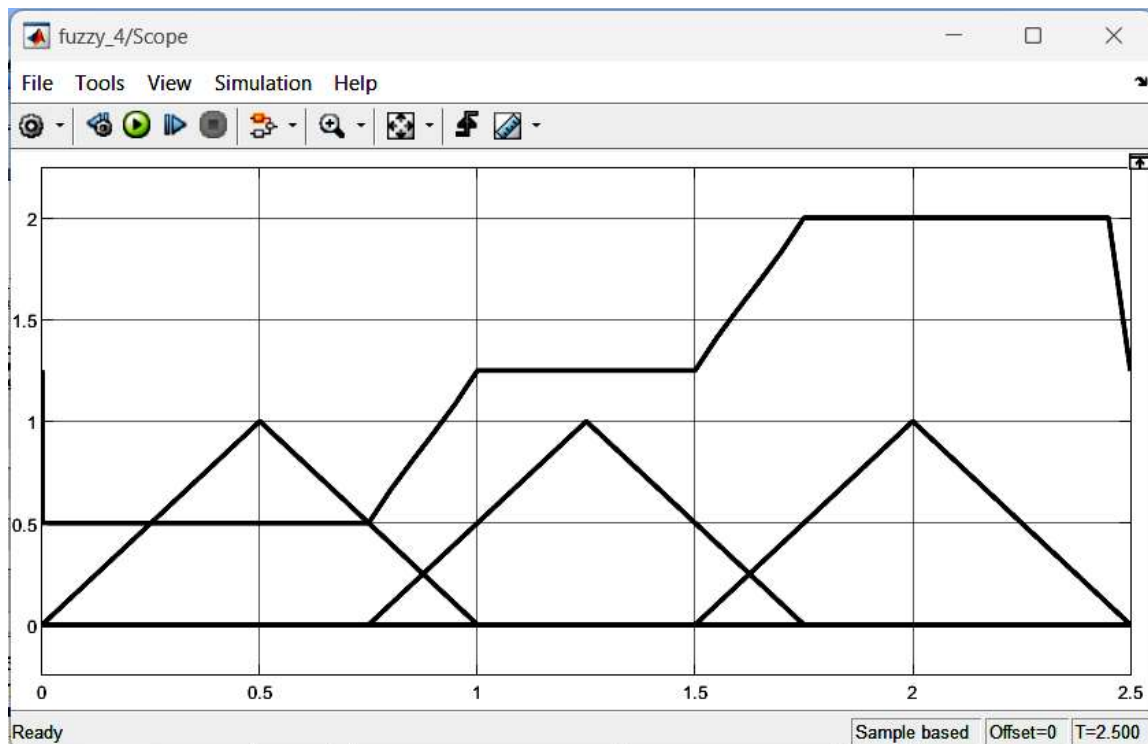


Рисунок 4.5 – Результати моделювання fuzzy-системи типу Mamdani

5. Робота з модулем Fuzzy Logic Toolbox у режимі командного рядку

Інтерактивний модуль **Fuzzy Logic Toolbox** дозволяє працювати не тільки у режимі графічних інтерфейсів користувача (GUI), але й має великий набір функцій, що виконуються безпосередньо з командного рядку системи Matlab. Більшість цих функцій є спеціальними *m*-файлами. Усі функції поділяються на групи:

- виклик програм графічного інтерфейсу;
- завдання функцій приналежності;
- редагування систем нечіткого логічного виведення;
- додаткові;
- сервісні;
- виклик діалогових вікон інтерфейсу;
- блоків Simulink;
- демонстрації можливостей пакету.

Розглянемо приклади виконання деяких команд, що найчастіше використовуються (спільно з основними функціями системи Matlab) для демонстрації можливостей пакету **Fuzzy Logic Toolbox**.

m-файли, що виконуються, рекомендується створювати та редагувати у редакторі-налагоджувачі. Наведені нижче приклади будемо порівнювати з fuzzy-системами, що були синтезовані раніше.

5.1. Побудова графіків функцій приналежності

У робочій області системи Matlab задамо параметри трьох функцій приналежності нечітких множин та побудуємо їх графіки.

У редакторі-налагоджувачі створимо *m*-файл для побудови графіків функції приналежності та після його виконання отримаємо графічне зображення (рис. 5.1). Під час першого запуску програми система вимагатиме присвоїти ім'я *m*-файлу у діалоговому вікні (рекомендується, щоб уникнути некоректної роботи задавати ім'я латинськими літерами).

У першому рядку *m*-файлу вказується діапазон зміни вхідного сигналу, позначеного символом *e* (від першої точки першого терму до останньої точки останнього терму). У рядках 2-4 задані три терми трикутних функцій приналежності. У лівій частині цих рядків вказані їх назви (**Ms, Mm, Mb**), а у правій частині позначення вхідної змінної та чисельні координати трикутників. Наприкінці усіх операторів стоїть крапка з комою, що перешкоджає виконанню окремого рядка програми без натискання клавіші **Enter**.

У п'ятому рядку програми командою **plot** виконується безпосереднє виведення на екран зображення графіків. Команда **grid on** створює зображення координатної сітки.

Кнопка запуску редактора команд Matlab

Кнопка запуску програми на виконання

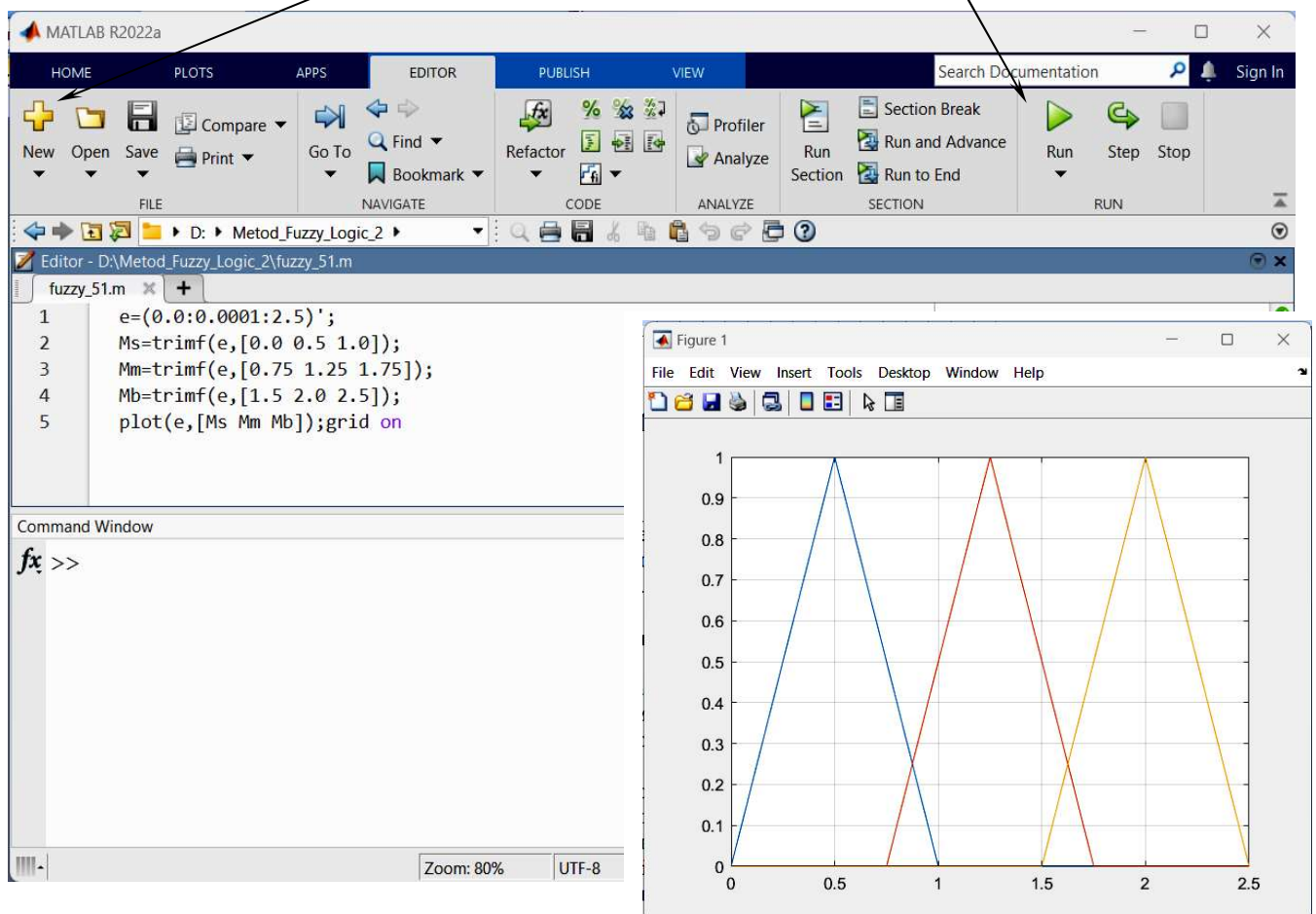


Рисунок 5.1 – Побудова графіків функцій приналежності

Необхідно уважно стежити за розділовими знаками між символами у рядках програми (крапка з комою, двокрапка, пробіл тощо). У разі некоректного введення команди система Matlab видасть повідомлення про помилку, яке висвітиться у вікні команд системи Matlab.

Зображений на рис. 5.1 графік не відформатований. У разі оформлення документів його необхідно відповідним чином відформатувати (збільшити товщину ліній, змінити колір, заливку області тощо).

5.2. Виконання процесу дефаззифікації

З допомогою команди **defuzz** виконаємо операцію дефаззифікації, тобто. за заданою функцією приналежності визначимо відповідне чітке число.

У редакторі-налагоджувачі створимо *m*-файл виконання процесу дефаззифікації та отримаємо її результати (рис. 5.2).

Структурно *m*-файл складається з трьох блоків, що обчислюють чисельні значення координат центру ваги для трьох функцій приналежності. У рядках 1,5,9 визначається інтервал зміни вхідного сигналу *e* (необхідно уважно стежити за синтаксисом написання цих рядків, який відрізняється від наведеного у прикладі 1). У рядках 2,6,10 задані параметри термів трикутних функцій приналежності, аналогічно прикладу 1. У рядках 3,7,11 вказані назви змінних, що є результатом дефаззифікації кожного нечіткого терму (**Xs**, **Xm**, **Xb**), а також найменування функцій приналежності (**Ms**, **Mm**, **Mb**) та метод дефаззифікації (**centroid**).

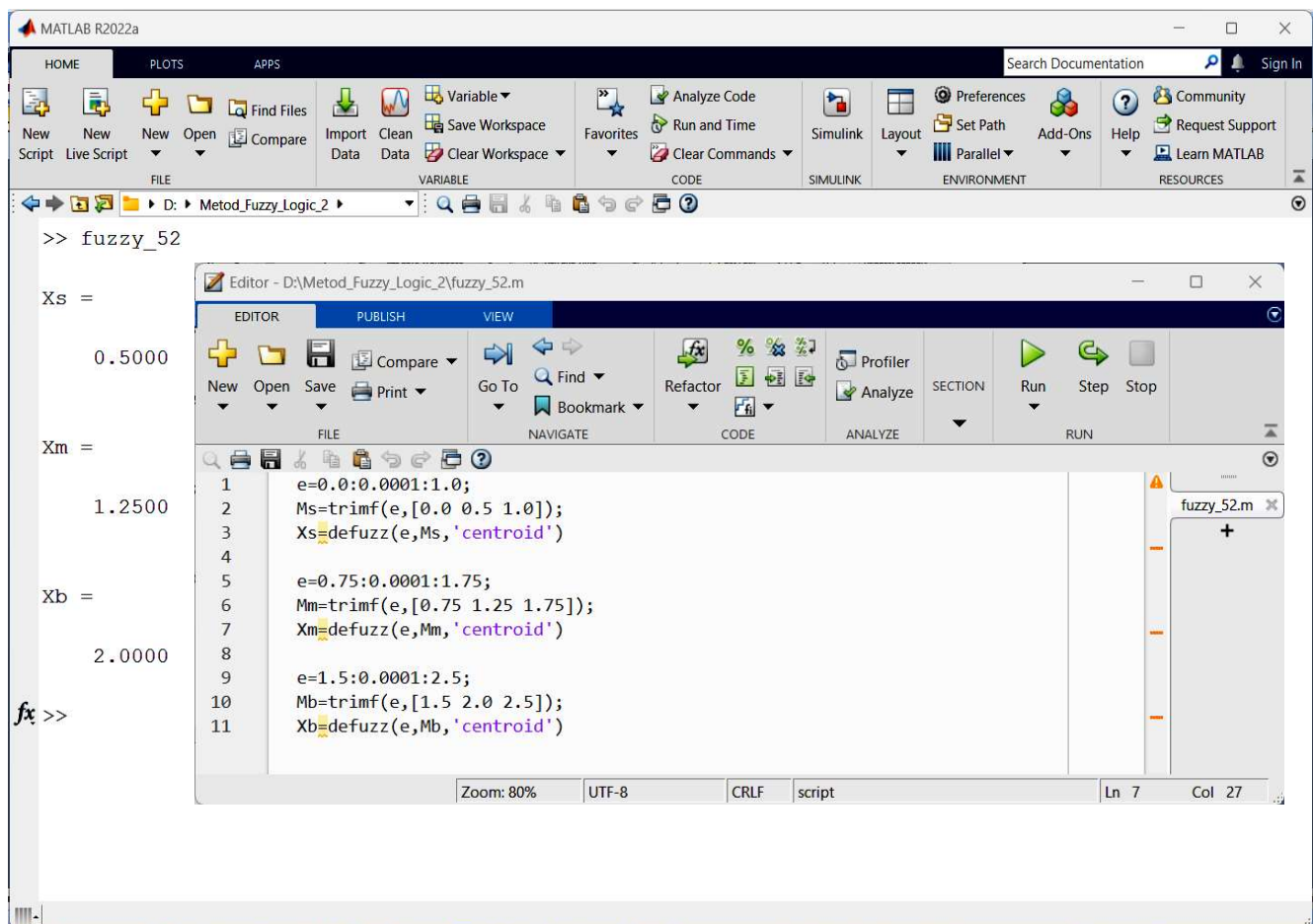


Рисунок 5.2 – *m*-файл та результати виконання процесу дефаззифікації

На задньому плані рис. 5.2 видно результати виконання *m*-файлу у вигляді трьох значень координат центрів ваги трьох трикутних функцій приналежності.

5.3. Відображення структури нечіткої моделі у вигляді системних команд

Після синтезу системи нечіткого логічного виведення за допомогою Fuzzy Logic Toolbox можна перевірити правильність завдання параметрів fuzzy-системи в цілому. Це досягається виконанням команди **showfis**, що дозволяє відобразити у вікні команд системи Matlab повну структуру створеної моделі нечіткого логічного виведення (fuzzy-системи) у вигляді послідовності рядків з відповідними операторами та чисельними й символьними позначеннями.

У якості прикладу покажемо виконання цієї команди для згенерованої раніше системи типу Mamdani (файл **fuzzy_controller_m.fis**, рис. 2.1 – рис. 2.6).

Спочатку необхідно відкрити файл **fuzzy_controller_m.fis**, експортувати його до робочого простору, а потім у редакторі-налагоджувачі або безпосередньо у вікні команд системи Matlab створити та виконати наступні рядки програми:

```
>> a=readfis('fuzzy_controller_m');  
>> showfis(a)
```

Після виконання цих функцій у вікні команд системи Matlab буде виведена наступна інформація, яка повністю відображає увесь процес нечіткого логічного виведення із зазначенням імен змінних, типів функцій приналежності, їх координат, методу дефазифікації тощо.

1.	Name	fuzzy_controller_m
2.	Type	mamdani
3.	Inputs/Outputs	[1 1]
4.	NumInputMFs	3
5.	NumOutputMFs	3
6.	NumRules	3
7.	AndMethod	min
8.	OrMethod	max
9.	ImpMethod	min
10.	AggMethod	max
11.	DefuzzMethod	centroid
12.	InLabels	ПОХИБКА
13.	OutLabels	КОЕФІЦІЄНТ
14.	InRange	[0 2.5]
15.	OutRange	[0 2.5]
16.	InMFLabels	Small-Ms
17.		Middle-Mm
18.		Big-Mb
19.	OutMFLabels	Small-Ms
20.		Middle-Mm
21.		Big-Mb
22.	InMFTypes	trimf
23.		trimf
24.		trimf
25.	OutMFTypes	trimf
26.		trimf
27.		trimf
28.	InMFParams	[0 0.5 1]
29.		[0.75 1.25 1.75]

```

30.                                [1.5 2 2.5]
31. OutMFParams                  [0 0.5 1]
32.                                [0.75 1.25 1.75]
33.                                [1.5 2 2.5]
34. Rule Antecedent              1
35.                                2
36.                                3
37. Rule Consequent              1
38.                                2
39.                                3
40. Rule Weight                   1
41.                                1
42.                                1
43. Rule Connection              2
44.                                2
45.                                2

```

5.4. Відображення структури нечіткої моделі у вигляді блок-схеми

Команда **plotfis** дозволяє вивести на екран монітора зображення структури системи нечіткого логічного виведення у вигляді візуальної блок-схеми.

У якості прикладу покажемо виконання цієї команди для згенерованої раніше системи типу Mamdani (файл **fuzzy_controller_m.fis**, рис. 2.1 – рис. 2.6).

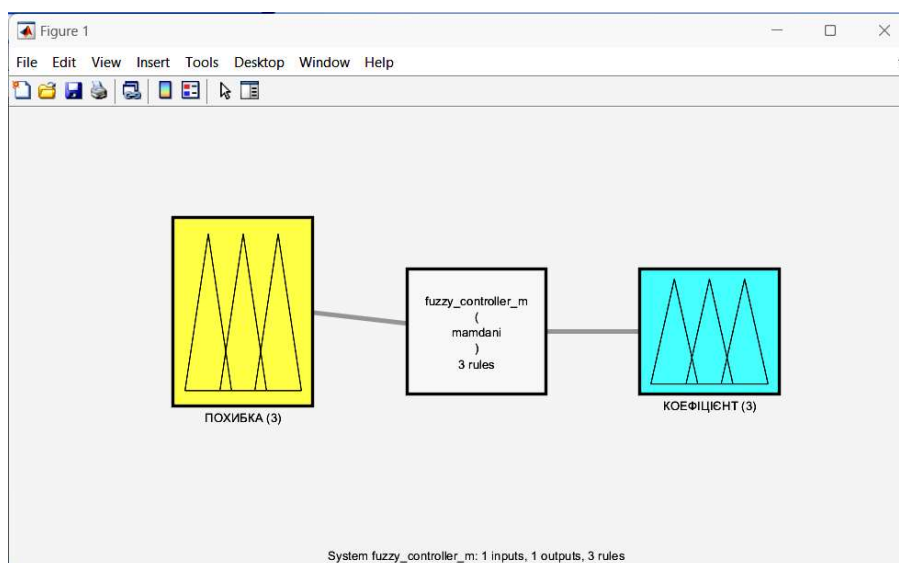
Спочатку необхідно відкрити файл **fuzzy_controller_m.fis**, експортувати його до робочого простору, а потім у редакторі-налагоджувачі або безпосередньо у вікні команд системи Matlab створити та виконати наступні рядки програми:

```

>> a=readfis('fuzzy_controller_m');
>> plotfis(a)

```

У результаті виконання на екрані з'явиться графічне зображення (рис. 5.3), що відображає функціональну блок-схему структури раніше створеної fuzzy-системи. Ця команда застосовується лише для візуального відображення структури fuzzy-системи у разі оформлення різного роду документів та звітів.



Потрібно відзначити, що отримана схема не відформатована. У разі оформлення документів її необхідно відповідним чином відформувати (збільшити товщину ліній, змінити колір, заливку області тощо).

Рисунок 5.3 – Зображення структури fuzzy-системи

5.5. Відображення графіків функцій приналежності

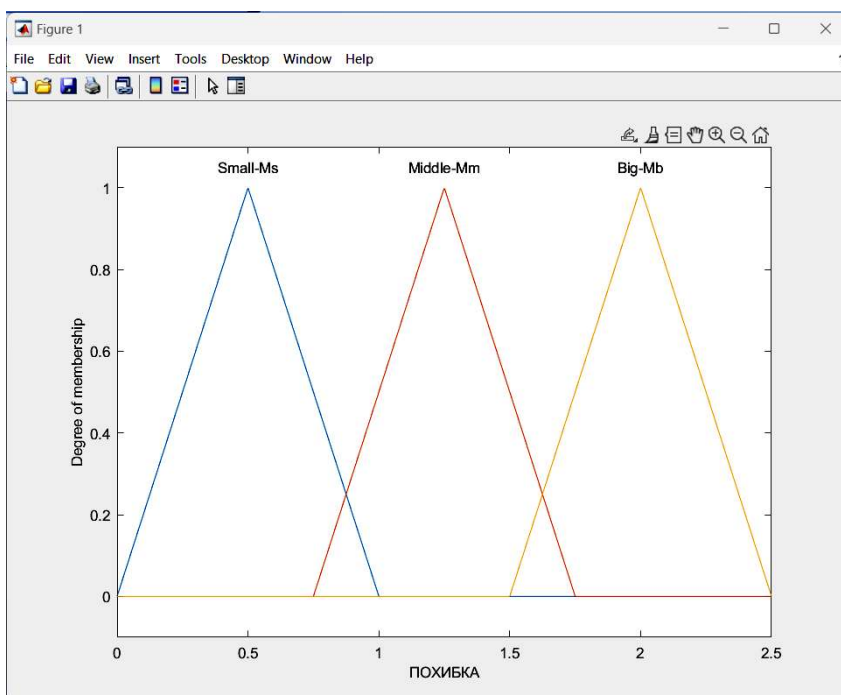
Команда **plotmf** дозволяє вивести на екран монітора зображення графіків функцій приналежності системи нечіткого логічного виведення.

У якості прикладу покажемо виконання цієї команди для згенерованої раніше системи типу Mamdani (файл **fuzzy_controller_m.fis**, рис. 2.1 – рис. 2.6).

Спочатку необхідно відкрити файл **fuzzy_controller_m.fis**, експортувати його до робочого простору, а потім у редакторі-налагоджувачі або безпосередньо у вікні команд системи Matlab створити та виконати наступні рядки програми:

```
>> a=readfis('fuzzy_controller_m');  
>> plotmf(a,'input',1,1000)
```

У результаті виконання на екрані з'явиться графічне зображення (рис. 5.4), що відображає входні функції приналежності раніше створеної fuzzy-системи.



У другому рядку в лапках вказують тип функції приналежності (вхід – **input**, вихід – **output**), порядковий номер змінної – **1**, кількість точок дискретизації – **1000**.

Потрібно відзначити, що отримані графіки не відформатовані. У разі оформлення документів їх необхідно відповідним чином відформатувати (збільшити товщину ліній, змінити колір, заливку області тощо).

Рисунок 5.4 – Графіки функцій приналежності

5.6. Відображення графіку поверхні "входи-вихід"

Команда **gensurf** дозволяє вивести на екран зображення поверхні "входи-вихід" системи нечіткого логічного виведення.

У якості прикладу покажемо виконання цієї команди для згенерованої раніше системи типу Mamdani (файл **fuzzy_controller_m.fis**, рис. 2.1 – рис. 2.6).

Спочатку необхідно відкрити файл **fuzzy_controller_m.fis**, експортувати його до робочого простору, а потім у редакторі-налагоджувачі або безпосередньо у вікні команд системи Matlab створити та виконати наступні рядки програми:

```
>> a=readfis('fuzzy_controller_m');  
>> gensurf(a,[1],1,100)
```

У результаті виконання на екрані з'явиться графічне зображення (рис. 5.5), що відображає графік поверхні "входи-вихід" раніше створеної fuzzy-системи.

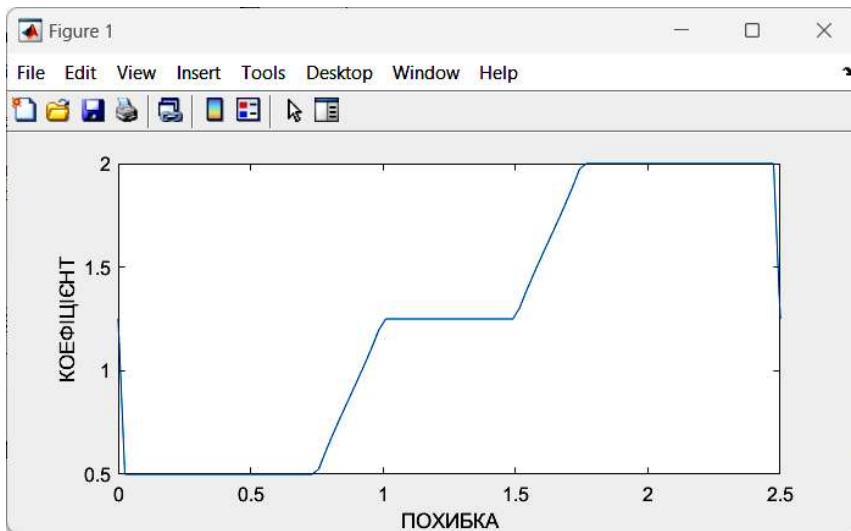


Рисунок 5.5 – Графік поверхні "входи-вихід"

Далі вказується порядковий номер вихідної змінної – **1**, якому буде поставлено у відповідність вісь аплікату поверхні.

На завершення можна вказати кількість точок дискретизації, що необхідні для покращення якості відображення графіку – **100**.

Потрібно відзначити, що отриманий графік не відформатований. У разі оформлення документів його необхідно відповідним чином відформатувати (збільшити товщину ліній, змінити колір, заливку області тощо).

5.7. Перетворення системи типу Mamdani у систему типу Sugeno

Команда **mam2sug** дозволяє перетворити систему нечіткого логічного виведення типу Mamdani у систему типу Sugeno. Fuzzy-система може мати декілька входів та виходів. Наслідком виконання цієї команди буде отримання системи типу Sugeno нульового порядку. Кількість правил у системі типу Sugeno залишається такою ж самою, як і у вихідній системі типу Mamdani. Без змін залишаються правила-посилки – частини **ЯКЩО**, а модифікуються лише правила-наслідки – частини **ТО**, для яких у якості вихідних змінних використовуються числові значення, що відповідають максимуму функцій приналежності вихідних змінних системи типу Mamdani.

У якості прикладу покажемо виконання цієї команди для згенерованої раніше системи типу Mamdani (файл **fuzzy_controller_m.fis**, рис. 2.1 – рис. 2.6).

Спочатку необхідно відкрити файл **fuzzy_controller_m.fis**, експортувати його до робочого простору, а потім у редакторі-налагоджувачі або безпосередньо у вікні команд системи Matlab створити та виконати наступні рядки програми:

```
>> a=readfis('fuzzy_controller_m');
>> b=mam2sug(a)
```

Ця послідовність команд перетворює раніше створену систему нечіткого логічного виведення типу Mamdani, яку у робочому просторі назвемо змінною *a*, у систему типу Sugeno, що позначена змінною *b*.

У другому рядку програми у квадратних дужках вказують вектор, який задає порядкові номери вхідних змінних системи, що будуть поставлені у відповідність вісям абсцис та ординат поверхні. У разі завдання лише однієї змінної буде побудовано двовірну залежність "вхід-вихід".

Після виконання цієї команди необхідно у системі Mamdani (файл **fuzzy_controller_m.fis**) у пункті головного меню **File** у підпункті **Import** обрати завантаження нової системи з робочої області Matlab **From Workspace....** У діалоговому вікні необхідно ввести назву нової системи – *b* та після цього відкриється вікно нової згенерованої системи типу Sugeno (рис. 5.6).

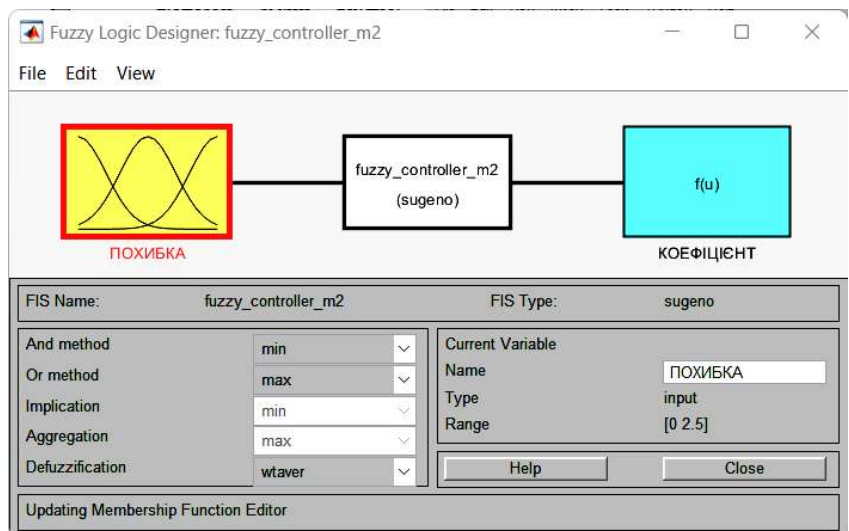


Рисунок 5.6 – Структура системи типу Sugeno, що перетворена з системи типу Mamdani

5.8. Заміна форми функцій приналежності еквівалентними

Команда **mf2mf** дозволяє замінити одну функцію приналежності еквівалентною (близькою) їй за формою іншою з деяким стандартним набором параметрів.

Таке перетворення функції виконується лише на рівні параметрів відповідних функцій приналежності. Функція заміни зберігає точки симетрії старої та нової функції приналежності. Однак результат виконання нової функції може призвести до втрати деякої інформації під час виконання зворотного перетворення.

Складемо *m*-файл:

```
>> e = 0:0.01:5;
>> a = [1 2 3];
>> b = mf2mf(a, 'gbellmf', 'trimf');
>> plot(e,gbellmf(e,a),e,trimf(e,b));grid on
```

У першому рядку програми встановлено діапазон зміни вхідної змінної. У другому рядку – ім'я – *a* та параметри вихідної функції приналежності, що підлягає перетворенню (дзвоноподібна). У третьому рядку – ім'я – *b* та вказівка щодо функції, у яку перетворювати вихідну (у трикутну). У четвертому рядку команда побудови графіків функцій приналежності вихідної та перетвореної. Результат виконання операції зображено на рис. 5.7.

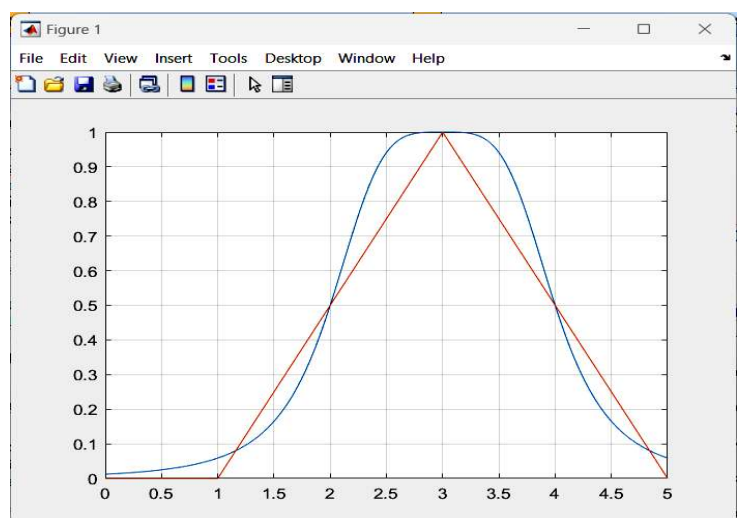


Рисунок 5.7 – Графіки вихідної та еквівалентної функції приналежності

5.9. Виконання основних нечітких арифметичних операцій

Розглянемо команду **fuzarith**, що дозволяє виконувати операції *нечіткої арифметики* (додавання, віднімання, множення та ділення) з нечіткими множинами.

Використовуючи арифметичні операції з нечіткими множинами, дана команда створює результуючу нечітку множину, як результат виконання бінарної операції.

У якості прикладу розглянемо чотири арифметичні операції над двома нечіткими множинами – трапецієподібною (*a*) та трикутною (*b*). Результат операції – нечітка множина (c_i , де $i = 1 \dots 4$).

Складемо чотири *m*-файла, що послідовно виконують арифметичні операції:

Нечітке додавання $c=a+b$

```
e=0:0.001:11;  
a=trapmf(e,[1 2 3 5]);  
b=trimf(e,[3 5 6]);  
c=fuzarith(e,a,b,'sum');  
plot(e,a,e,b,e,c);grid on
```

Нечітке віднімання $c=a-b$

```
e=-6:0.001:6;  
a=trapmf(e,[1 2 3 5]);  
b=trimf(e,[3 5 6]);  
c=fuzarith(e,a,b,'sub');  
plot(e,a,e,b,e,c);grid on
```

Нечітке множення $c=a*b$

```
e=1:0.001:30;  
a=trapmf(e,[1 2 3 5]);  
b=trimf(e,[3 5 6]);  
c=fuzarith(e,a,b,'prod');  
plot(e,a,e,b,e,c);grid on
```

Нечітке ділення $c=a/b$

```
e=0:0.001:6;  
a=trapmf(e,[1 2 3 5]);  
b=trimf(e,[3 5 6]);  
c=fuzarith(e,a,b,'div');  
plot(e,a,e,b,e,c);grid on
```

Результати виконання чотирьох нечітких арифметичних операцій з двома нечіткими множинами зображено на рис. 5.8.

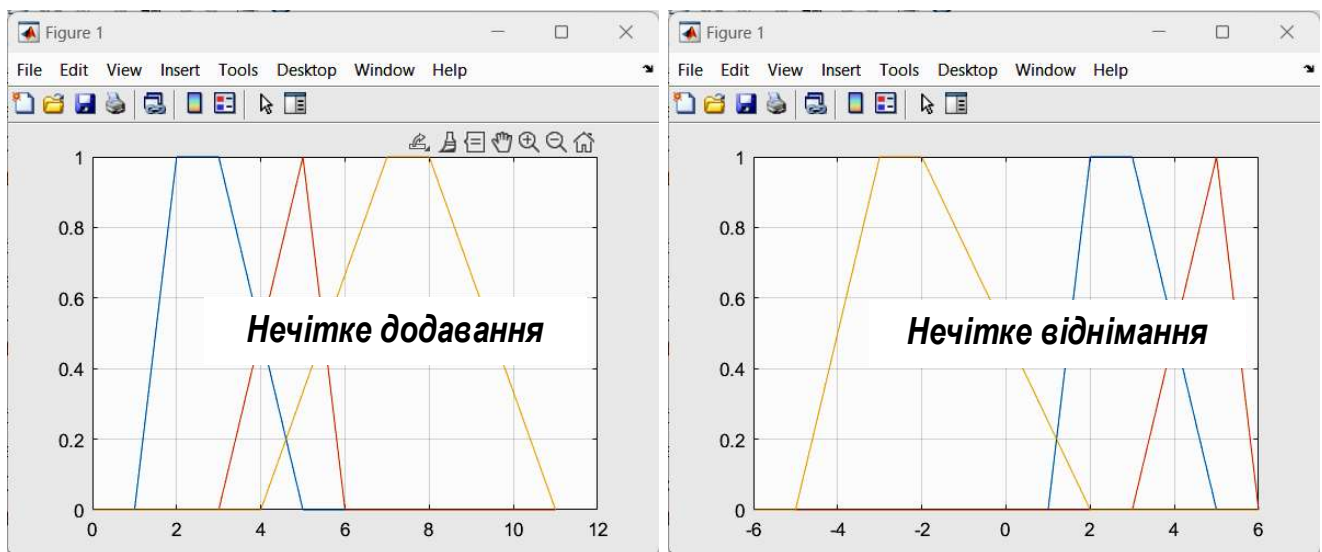


Рисунок 5.8 – Результати виконання арифметичних операцій з двома нечіткими множинами (початок)

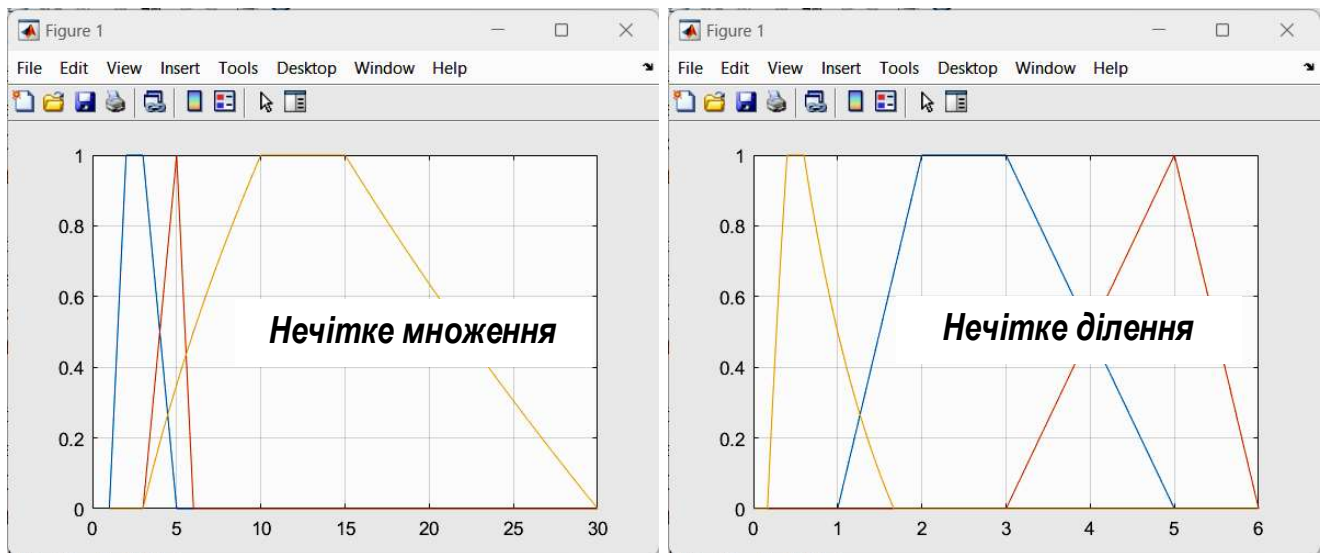


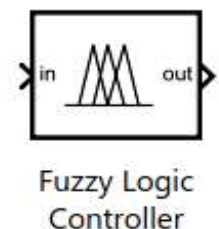
Рисунок 5.8 – Результати виконання арифметичних операцій з двома нечіткими множинами (закінчення)

6. Спеціальні Simulink-блоки побудови fuzzy-систем

Бібліотека нечіткої логіки (**Fuzzy Logic Toolbox**) браузера бібліотек **Simulink** містить 2 блоки, що реалізують fuzzy-контролери, 15 блоків побудови функцій приналежності та блоки виконання різних логічних операцій над нечіткими множинами. Розглянемо найпоширеніші з них.

Блок fuzzy-контролера (Fuzzy Logic Controller)

Цей блок дозволяє у додатку Simulink моделювати роботу системи нечіткого логічного виведення. У вікні налаштувань блоку необхідно вказати ім'я FIS-структури, що експортована у робочу область системи Matlab після синтезування її у модулі Fuzzy Logic Toolbox.



Блок fuzzy-контролера з вікном перегляду правил (Fuzzy Logic Controller with Ruleviewer)

Цей блок ідентичний першому, проте, дозволяє у процесі моделювання нечіткої системи переглядати логічні правила. Якщо нечітка система має кілька входів чи виходів, вони повинні бути з'єднані разом до розміщення у блоках контролера.



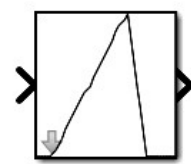
Блоки функцій приналежності (Membership Function)

Ця бібліотека блоків дозволяє будувати 15 типів функцій приналежності, кожна з яких представлена окремим функціональним блоком. Наведемо деякі з них.



Трикутна функція приналежності (Triangular MF)

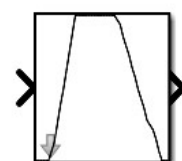
Цей блок використовується для побудови трикутної функції приналежності нечіткої множини. Вікно налаштування параметрів містить три аргументи: a, b, c – деякі числові параметри, що набувають довільних дійсних значень та впорядковані відношенням $a \leq b \leq c$. При цьому значення a та c визначають координати основи трикутника, а значення b – координату вершини трикутника.



Triangular MF

Трапецієподібна функція приналежності (Trapezoidal MF)

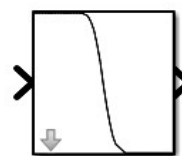
Цей блок використовується для побудови трапецієподібної функції приналежності нечіткої множини. Вікно налаштування параметрів містить чотири аргументи: a, b, c, d – деякі числові параметри, що набувають довільних дійсних значень та впорядковані відношенням $a \leq b \leq c \leq d$. При цьому значення a та d визначають координати нижньої основи трапеції, а значення b та c – координати верхньої основи трапеції.



Trapezoidal MF

Z-подібна функція приналежності (Z-Shaped MF)

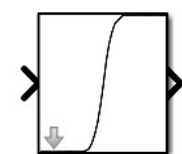
Цей блок використовується для побудови Z-подібної (сплайнової) функції приналежності нечіткої множини. Вікно налаштування параметрів містить два аргументи: a, b – деякі числові параметри, що набувають довільних дійсних значень та впорядковані відношенням $a < b$. При цьому значення a та b визначають діапазон зменшення функції.



Z-shaped MF

S-подібна функція приналежності (S-Shaped MF)

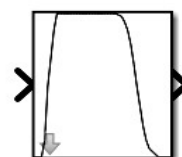
Цей блок використовується для побудови S-подібної (сплайнової) функції приналежності нечіткої множини. Вікно налаштування параметрів містить два аргументи: a, b – деякі числові параметри, що набувають довільних дійсних значень та впорядковані відношенням $a < b$. При цьому значення a та b визначають діапазон зростання функції.



S-shaped MF

Pi-подібна функція приналежності (Pi-Shaped MF)

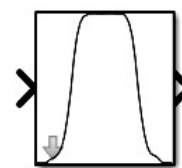
Даний блок використовується для побудови Pi-подібної функції приналежності нечіткої множини, що визначається як алгебраїчний добуток двох сплайнових функцій: Z-подібної та S-подібної. Вікно налаштування параметрів містить чотири аргументи: a, b, c, d . При цьому значення a та b характеризують S-подібну криву, а значення c та d характеризують Z-подібну криву.



Pi-shaped MF

Дзвоноподібна функція приналежності (Generalized Bell MF)

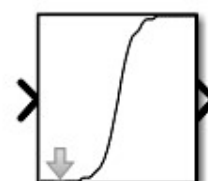
Даний блок використовується для побудови П-подібної функції приналежності нечіткої множини за типом дзвоноподібної кривої. Вікно параметрів містить три аргументи: a, b, c , що визначають параметри нелінійної функції. При цьому параметр b має бути обов'язково додатнім.



Generalized Bell MF

Сигмоїдальна функція приналежності (Sigmoidal MF)

Даний блок використовується для побудови сигмоїдальної функції приналежності нечіткої множини. Функція приналежності визначається за наступною аналітичною залежністю: $f(x) = 1/(1+\exp(-a(x-c)))$. Вікно налаштування параметрів містить дві величини a, c . Якщо значення параметру a додатне, крива зростає зліва направо, і навпаки, якщо значення параметру a від'ємне, крива зменшується зліва направо. Параметр c задає точку перегину кривої.



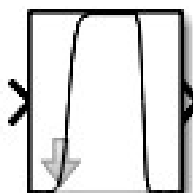
Sigmoidal MF

Добуток двох сигмоїдальних функцій приналежності (Prod. Sigmoidal MF)

Даний блок реалізує функцію приналежності на основі добутку двох сигмоїдальних кривих, які визначаються за наступними рівняннями:

$$f_k(x) = 1/(1+\exp(-a_k(x-c_k))), \quad k=1,2.$$

Параметри a_1 та a_2 встановлюють нахили лівої та правої частини кривої. Параметри c_1 та c_2 встановлюють точки перегину для лівої та правої частини кривої. Параметри a_1 та a_2 повинні бути додатним та від'ємним відповідно.



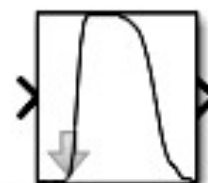
Prod. Sigmoidal MF

Різниця двох сигмоїдальних функцій приналежності (Diff. Sigmoidal MF)

Даний блок реалізує функцію приналежності на основі різниці двох сигмоїдальних кривих, які визначаються за наступними рівняннями:

$$f_k(x) = 1/(1+\exp(-a_k(x-c_k))), \quad k=1,2.$$

Параметри a_1 та a_2 встановлюють нахили лівої та правої частини кривої. Параметри c_1 та c_2 встановлюють точки перегину для лівої та правої частини кривої. Параметри a_1 та a_2 повинні бути додатними.



Diff. Sigmoidal MF

СПИСОК ВИКОРИСТАНИХ ЛІТЕРАТУРНИХ ДЖЕРЕЛ

Базова література:

1. Математичний апарат штучного інтелекту в електроенергетичних системах: підручник / В.В. Кирик. – Київ : КПІ ім. Ігоря Сікорського, Вид-во «Політехніка», 2019. – 224 с.
2. Лозинський А.О., Мороз В.І., Паранчук Я.С. Розв'язання задач електромеханіки в середовищах пакетів MathCAD і MATLAB: Навчальний посібник. – Львів: Видавництво Державного університету «Львівська політехніка», 2000. – 166 с.
3. Ротштейн О.П. Інтелектуальні технології ідентифікації: нечіткі множини, генетичні алгоритми, нейронні мережі / О.П. Ротштейн. – Вінниця: УНІВЕРСУМ-Вінниця, 1999. – 320 с.
4. Щерба А.А. Системи з нечіткою логікою регулювання електроенергетичних режимів / А.А. Щерба, В.В. Кирик. – К.: Лазурит – Поліграф, 2011. – 329 с.
5. Використання пакета MATLAB–Simulink для моделювання динамічних систем та пристроїв: Метод. вказівки до виконання лабораторних, розрахунково-графічних робіт, курсового та дипломного проектування для студ. спец. 7.092203 – «Електромеханічні системи автоматизації та електропривод» і 7.092204 – «Електромеханічне обладнання енергоємних виробництв» / Укладачі: О.В. Чермалих, О.В. Данілін, В.В. Кузнецов. – К.: ІВЦ «Політехніка», 2004. – 72 с.

Допоміжна література:

6. Зайченко Ю.П. Основи проектування інтелектуальних систем: навч. посіб. / Ю.П. Зайченко. – К.: Вид. дім «Слово», 2004. – 352 с.
7. Системи функції-управління / Архангельський В.І., Богаєнко І.Н., Грабовський Г.Г., Рюмшин М.А. – Київ: Техніка, 1997. – 208 с.
8. Zadeh L.A. Fuzzy sets. – Information and Control, vol. 8, 1965, pp. 338 – 353.
9. Mamdani E.H. Application of fuzzy logic to approximate reasoning using linguistic synthesis. – IEEE Transactions on Computers, vol. 26, no. 12, 1977, pp. 1182 – 1191.
10. Sugeno M. Fuzzy measures and fuzzy integrals: a survey. (in M.M. Gupta, G.N. Saridis, and B.R. Gaines, editors) Fuzzy Automata and Decision Processes, pp. 89 – 102, North-Holland, New-York, 1977.