

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені В. Н. КАРАЗІНА

**ТЕХНОЛОГІЇ НЕЙРОННИХ МЕРЕЖ І НЕЧІТКОГО
МОДЕЛЮВАННЯ В СИСТЕМАХ УПРАВЛІННЯ**

Методичні вказівки
до проведення практичних занять для здобувачів вищої освіти
другого (магістерського) рівня за спеціальністю 174 «Автоматизація,
комп'ютерно-інтегровані технології та робототехніка»

У двох частинах

Частина 2

Електронний ресурс

Харків – 2025

Рецензенти:

Р. М. Тріщ – доктор технічних наук, професор, завідувач кафедри мехатроніки та електротехніки Національного аерокосмічного університету ім. М. Є. Жуковського «Харківський авіаційний інститут»;

О. О. Литвин – доктор фізико-математичних наук, професор, завідувач кафедри ХТЛПід Навчально-наукового інституту «Українська інженерно-педагогічна академія».

*Затверджено до розміщення в мережі Інтернет рішенням Науково-методичної ради
Харківського національного університету імені В. Н. Каразіна
(протокол № 11 від 25 червня 2025 року)*

Т 38 **Технології** нейронних мереж і нечіткого моделювання в системах управління : методичні вказівки до проведення практичних занять для здобувачів вищої освіти другого (магістерського) рівня за спеціальністю 174 «Автоматизація, комп'ютерно-інтегровані технології та робототехніка». У 2-х ч. Ч. 2 [Електронний ресурс] / уклад. Г. І. Канюк, Т. Ю. Василець. – Харків : ХНУ імені В. Н. Каразіна, 2025. – (PDF 96 с.)

Методичні вказівки з проведення практичних занять із дисципліни «Технології нейронних мереж і нечіткого моделювання в системах управління» спрямовані на відпрацювання отриманих знань і накопичення досвіду з розрахунку інтелектуальних систем управління. Метою практичних занять другої частини курсу є набуття студентами практичних навичок синтезу нечітких систем з використанням пакету прикладних програм MATLAB.

Для здобувачів вищої освіти освітнього другого (магістерського) рівня, що навчаються за освітньо-професійною програмою «Автоматизація та комп'ютерно-інтегровані технології» за спеціальністю 174 «Автоматизація, комп'ютерно-інтегровані технології та робототехніка».

УДК 658.5.011.56:658.589(075.5)

© Харківський національний університет імені В. Н. Каразіна, 2025

© Канюк Г. І., Василець Т. Ю., укл., 2025

ЗМІСТ

Практичне заняття 1. ВИВЧЕННЯ ФУНКЦІЙ ПАКЕТУ FUZZY LOGIC TOOLBOX СИСТЕМИ MATLAB.....	5
1.1 Мета заняття	5
1.2 Призначення і можливості пакету Fuzzy Logic Toolbox	5
1.3 Графічний інтерфейс Fuzzy Logic Toolbox.....	5
1.4 Редактор систем нечіткого виведення FIS Editor	6
1.5 Редактор функцій приналежності Membership Function Editor	10
1.6 Редактор правил системи нечіткого висновку Rule Editor	13
1.7 Програма перегляду правил системи нечіткого висновку Rule Viewer	15
1.8 Програма перегляду поверхні системи нечіткого висновку Surface Viewer .	17
1.9 Контрольні питання по темі заняття	19
Практичне заняття 2. ПОБУДОВА НЕЧІТКИХ МОДЕЛЕЙ БАГАТОМАСОВИХ ЕЛЕКТРОМЕХАНІЧНИХ СИСТЕМ ЗА ДОПОМОГОЮ ПАКЕТУ FUZZY LOGIC TOOLBOX СИСТЕМИ MATLAB	20
2.1 Мета заняття	20
2.2 Методика синтезу нечіткої моделі двомасової електромеханічної системи .	20
2.3 Завдання системи нечіткого висновку з використанням вікна FIS Editor	23
2.4 Завдання функцій приналежності змінних системи.....	23
2.5 Завдання правил нечіткого висновку	28
2.6 Робота з вікнами перегляду правил і поверхні відгуку.....	29
2.7 Моделювання синтезованої нечіткої моделі	33
2.8 Завдання для самостійного виконання.....	34
2.9 Контрольні питання по темі заняття	34
Практичне заняття 3. ПОБУДОВА МОДЕЛЕЙ ДИНАМІЧНИХ ОБ'ЄКТІВ З ВИКОРИСТАННЯМ ГІБРИДНИХ МЕРЕЖ.....	35
3.1 Мета заняття	35
3.2 Реалізація ANFIS в середовищі MATLAB	35
3.3 Вивчення пунктів меню і опції редактора	37
3.4 Генерація навчальних, тестових і перевірочних даних нечіткої моделі гібридної мережі для вирішення завдання ідентифікації двомасової електромеханічної системи	38
3.5 Генерація системи нечіткого виведення.....	44
3.6 Навчання гібридної мережі	47
3.7 Дослідження побудованої моделі гібридної мережі.....	51
3.8 Моделювання нечіткої моделі гібридної мережі	52
3.9 Завдання для самостійного виконання.....	54
3.10 Контрольні питання по темі заняття	54

Практичне заняття 4. СИНТЕЗ НЕЧІТКИХ РЕГУЛЯТОРІВЗ ВИКОРИСТАННЯМ СИСТЕМИ MATLAB.....	55
4.1 Мета заняття	55
4.2 Розробка моделі системи управління з нечітким регулятором	55
4.3 Формування бази правил системи нечіткого висновку	58
4.4 Розробка системи нечіткого висновку	60
4.5 Моделювання двомасової електромеханічної системи з нечітким регулятором	70
4.6 Завдання для самостійного виконання.....	74
4.7 Контрольні питання по темі заняття	74
СПИСОК ЛІТЕРАТУРИ.....	75
ДОДАТОК А. ВАРІАНТИ ЗАВДАНЬ ДЛЯ САМОСТІЙНОЇ РОБОТИ.....	77

Практичне заняття 1

ВИВЧЕННЯ ФУНКЦІЙ ПАКЕТУ FUZZY LOGIC TOOLBOX СИСТЕМИ MATLAB

1.1 Мета заняття

Ознайомлення з основними функціями пакету **Fuzzy Logic Toolbox** у середовищі MATLAB для створення, налаштування та моделювання нечітких логічних систем. Вивчення інструментів проєктування нечітких систем управління та застосування функцій для побудови та аналізу нечітких моделей.

1.2 Призначення і можливості пакету Fuzzy Logic Toolbox

Пакет нечіткої логіки Fuzzy Logic Toolbox – це пакет прикладних програм, що відносяться до теорії розмитих або нечітких множин і дозволяє конструювати так звані нечіткі експертні і управляючі системи.

Основні можливості пакету:

- побудова систем нечіткого висновку (експертних систем, регуляторів, аппроксиматоров залежностей);
- побудова адаптивних нечітких систем (гібридних нейронних мереж);
- інтерактивне динамічне моделювання систем з нечіткою логікою в середовищі пакету блокового моделювання Simulink.

Пакет забезпечує роботу:

- у режимі командного рядка;
- у вікнах графічного інтерфейсу;
- у середовищі пакету розширення Simulink.

Пакет Fuzzy Logic Toolbox може працювати в операційному середовищі системи MATLAB. Спільно з Fuzzy Logic Toolbox можуть використовуватися і інші пакети розширення, зокрема, пакет Simulink, що значно підвищує наочність рішень.

1.3 Графічний інтерфейс Fuzzy Logic Toolbox

Більшість додатків системи MATLAB мають найбільш наочну реалізацію при використанні засобів графічного інтерфейсу користувача GUI. До складу програмних засобів Fuzzy Logic Toolbox входять наступні основні програми, що дозволяють працювати в режимі GUI:

- редактор нечіткої системи виведення **Fuzzy Inference System Editor (FIS Editor** або **FIS-редактор**) разом з допоміжними програмами – редактором функцій приналежності (**Membership Function Editor**), редактором правил (**Rule Editor**),

переглядачем правил (**Rule Viewer**) і переглядачем поверхні нечіткого висновку (**Surface Viewer**);

- редактор гібридних систем (**ANFIS Editor, ANFIS-редактор**);
- програма знаходження центрів кластерів (програма **Clustering** – кластеризація).

Набір даних програм надає користувачеві максимальні зручності для створення, редагування і використання різних систем нечіткого висновку.

1.4 Редактор систем нечіткого виведення **FIS Editor**

Редактор систем нечіткого виведення **FIS** (або просто редактор **FIS**) є основним засобом, який використовується для створення або редагування систем нечіткого висновку в графічному режимі. Редактор **FIS** може бути відкритий за допомогою введення функції `fuzzy` або `fuzzy('fismat')` у вікні команд. Ця функція надає користувачеві можливість задавати і редагувати на високому рівні властивості системи нечіткого висновку, такі як число вхідних і вихідних змінних, тип системи нечіткого висновку, використовуваний метод дефазифікації і т. д.

Якщо функція `fuzzy` викликається без аргументів, то редактор **FIS** викликається для новостворюваної системи нечіткого висновку з ім'ям `Untitled` за умовчанням (рис.1.1). При цьому за умовчанням також задається цілий ряд параметрів, таких як тип системи нечіткого висновку (Мамдані), нечіткі логічні операції, методи імплікації, агрегації і дефазифікації і деякі інші. Користувач може погодитися з цими значеннями або змінити їх.

Якщо функція `fuzzy` викликається з аргументом у формі `fuzzy('fismat')`, де `fismat` – ім'я зовнішнього файлу з розширенням `fis` з вже розробленою системою нечіткого висновку, то редактор викликається з вже завантаженою системою **FIS** з ім'ям `fismat` (рис. 1.2).

Можливий також виклик редактора **FIS** за допомогою цієї ж функції у форматі `fuzzy(fismat)`, де `fismat` – ім'я структури **FIS** в робочій області **MATLAB**. В цьому випадку відповідна структура нечіткого висновку повинна бути заздалегідь створена (наприклад, засобами командного режиму) або завантажена в робочу область за допомогою функції `readfis('fismat')`, де `fismat` – ім'я зовнішнього `fis`-файлу з розробленою раніше системою нечіткого висновку. Зокрема, виклик редактора функцією `fuzzy('Kran_Pidjom')` завантажує в нього нечітку систему висновку, що використовується для управління двомасовою електромеханічною системою механізму підйому мостового крана.

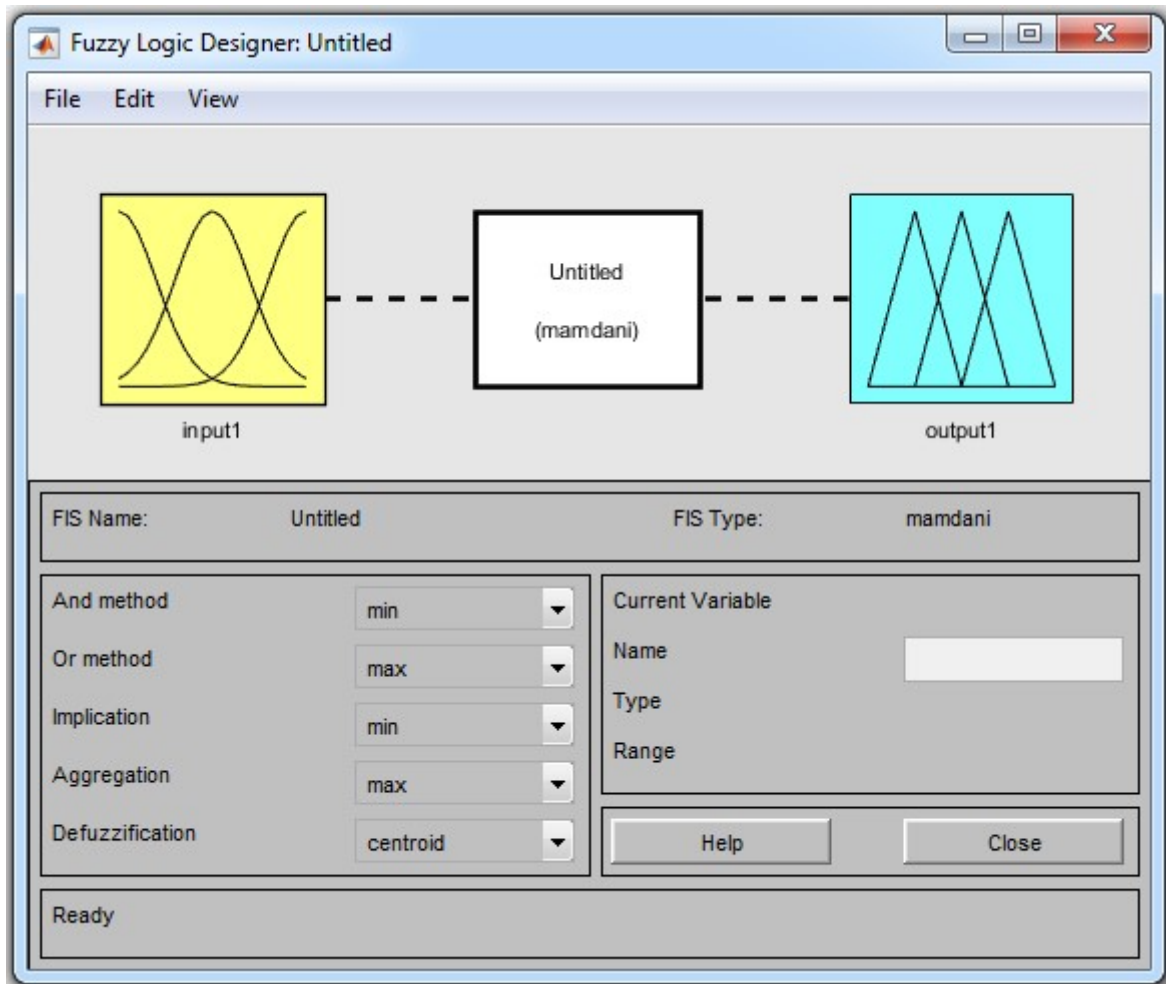


Рис. 1.1. Графічний інтерфейс редактора FIS, що викликається функцією fuzzy

Редактор FIS має графічний інтерфейс і дозволяє викликати всі інші редактори і програми перегляду систем нечіткого висновку. Графічний інтерфейс цього редактора має максимальну зручність і гнучкість, необхідну для інтерактивної роботи з окремими компонентами системи нечіткого висновку.

У верхній частині робочого інтерфейсу редактора FIS зображується діаграма, що представляє у візуальній формі входи і виходи системи нечіткого висновку, в центрі яких знаходиться так званий процесор нечітких правил. Клацання на прямокутнику із зображенням входу або виходу виділяє відповідну змінну і робить її поточною. Прямокутник поточної змінної при цьому виділяється червоним кольором.

Подвійне клацання на прямокутнику із зображенням вхідної або вихідної змінної викликає редактор функцій приналежності із завантаженою в нього відповідною змінною. Подвійне клацання на зображенні процесора нечітких правил викликає редактор правил для відповідної системи нечіткого висновку. Якщо деяка змінна існує в системі нечіткого висновку, але не використовується в правилах висновку, то зв'язок її з процесором нечітких правил зображується не суцільною, а пунктирною лінією.

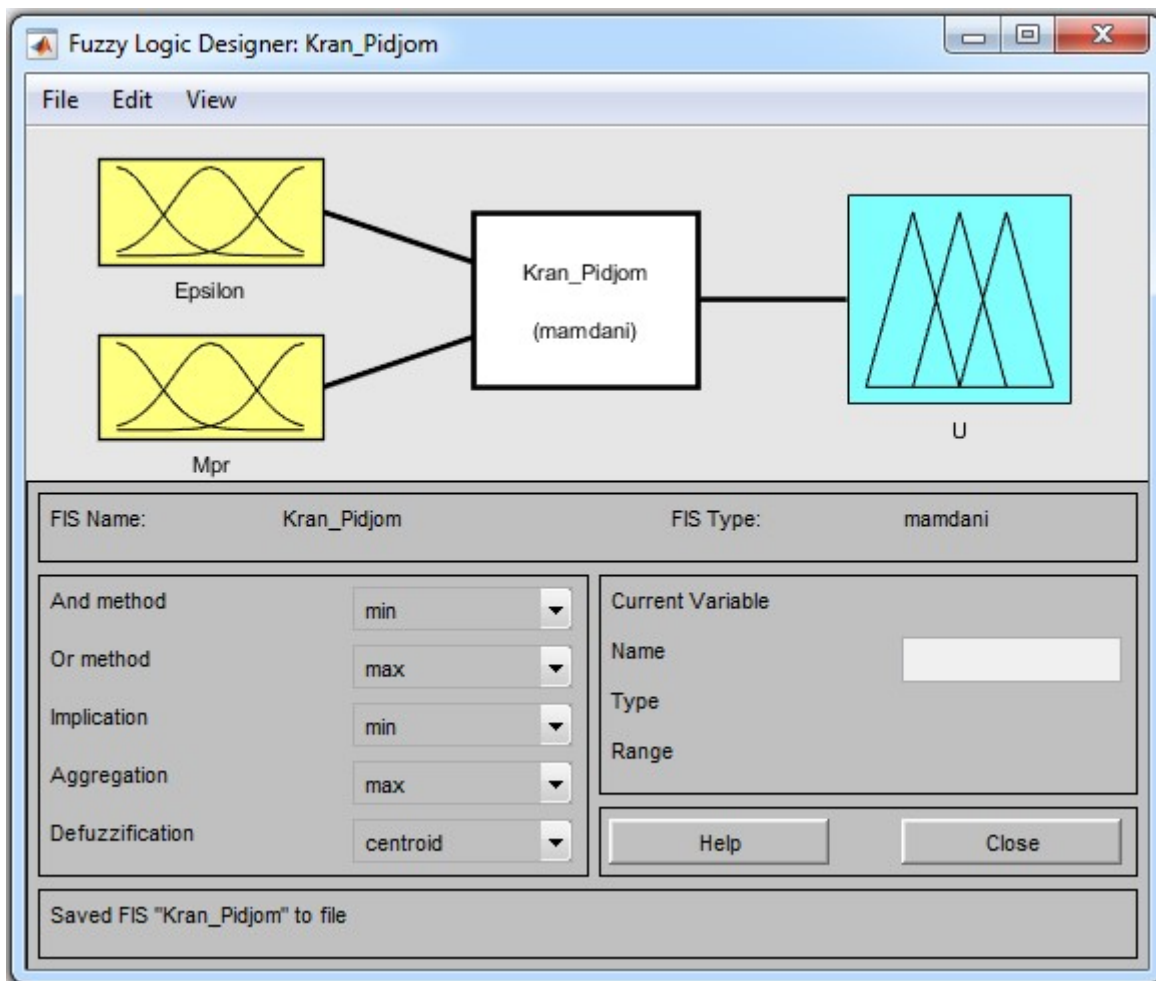


Рис. 1.2. Графічний інтерфейс редактора FIS, що викликається функцією fuzzy('Kran_Pidjom)

Редактор FIS має головне меню, яке дозволяє користувачеві викликати інші графічні засоби роботи з системою нечіткого виведення FIS, завантажувати і зберігати структуру FIS в зовнішніх файлах і т.д. Розглянемо призначення пунктів меню редактора FIS.

- Пункт меню **File** (Файл) редактора FIS містить наступні операції:
 - **New FIS...** – дозволяє вибрати тип нової системи нечіткого висновку, що задається: **Mamdani** – типа Мамдані або **Sugeno** – типа Сугено. При цьому система нечіткого висновку, що задається, не має ні вхідних, ні вихідних змінних, а її ім'я задається за умовчанням як Untitled;
 - **Import** – дозволяє завантажити в редактор FIS існуючу систему нечіткого висновку одним з наступних способів: **From Workspace...** – з робочого простору програми MATLAB або **From Disk...** – із зовнішнього файлу. У останньому випадку викликається стандартне діалогове вікно відкриття зовнішнього файлу з диска;
 - **Export** – дозволяє зберегти редаговану систему нечіткого висновку одним з наступних способів: **To Workspace...** – в робочому просторі програми

MATLAB або **To Disk...** – в зовнішньому файлі. У останньому випадку викликається стандартне діалогове вікно збереження файлу на диску;

- **Print** – дозволяє роздрукувати на принтері редаговану систему нечіткого висновку. В цьому випадку викликається стандартне діалогове вікно налаштування властивостей друку на підключеному до даного комп'ютера принтері;
- **Close** – закриває редактор **FIS**, при цьому викликається діалогове вікно з пропозиціями зберегти або відмовитися від збереження редагованої системи нечіткого висновку.

▪ Пункт меню **Edit** (Редагування) містить наступні операції:

- **Undo** – відмінняє виконання останньої дії;
- **Add Variable...** – дозволяє додати в редаговану систему нечіткого висновку змінну одного з наступних типів: **Input** – вхідну змінну або **Output** – вихідну змінну;
- **Remove Selected Variable** – видаляє вибрану змінну з редагованої системи нечіткого висновку;
- **Membership Functions...** – викликає редактор функцій приналежності;
- **Rules** – викликає редактор правил нечіткого висновку.

▪ Пункт меню **View** (Вигляд) містить наступні операції:

- **Rules** – викликає програму перегляду правил нечіткого висновку;
- **Surface** – викликає програму перегляду поверхні нечіткого висновку.

У лівій нижній частині робочого інтерфейсу редактора **FIS** є 5 спливаючих меню:

- **And method** (Метод логічної кон'юнкції) – дозволяє задати один з наступних методів для виконання логічної кон'юнкції в умовах нечітких правил:
 - **min** – метод мінімального значення;
 - **prod** – метод алгебраїчного добутку;
 - **Custom** – метод, визначений користувачем.
- **Or method** (Метод логічної диз'юнкції) – дозволяє задати один з наступних методів для виконання логічної диз'юнкції в умовах нечітких правил:
 - **max** – метод максимального значення;
 - **probor** – метод алгебраїчної суми;
 - **Custom** – метод, визначений користувачем.
- **Implication method** (Метод виведення висновку) – дозволяє задати один з наступних методів для виконання (активізації) логічного висновку в кожному з нечітких правил:
 - **min** – метод мінімального значення;
 - **prod** – метод алгебраїчного добутку;
 - **Custom** – метод, визначений користувачем. Це меню не використовується для систем нечіткого виведення типу Сугено.

- **Aggregation method** (Метод агрегації) – дозволяє задати один з наступних методів для агрегації значень функції приналежності кожній з вихідних змінних у висновках нечітких правил:
 - **max** – метод максимального значення;
 - **sum** – метод граничної суми;
 - **probor** – метод алгебраїчної суми.
 - **Custom** – метод, визначений користувачем.

Це меню не використовується для систем нечіткого виведення типа Сугено.

- **Denazification method** (Метод дефаззифікації) – дозволяє задати один з наступних методів для виконання дефаззифікації вихідних змінних в системі нечіткого виведення типа Мамдані:
 - **centroid** – метод центру тяжіння для дискретної безлічі значень функції приналежності;
 - **bisector** – метод центру площі;
 - **mom** (middle of maximum) – метод середнього максимуму, визначуваний як середнє арифметичне лівого і правого модальних значень;
 - **som** (smallest of maximum) – метод найменшого (лівого) модального значення;
 - **lom** (largest of maximum) – метод найбільшого (правого) модального значення;
 - **Custom** – метод, визначений самим користувачем.

Для систем нечіткого виведення типа Сугено можна вибрати один з наступних методів дефаззифікації:

- **wtaver** (weighted average) – метод зваженого середнього;
- **wtsum** (weighted sum) – метод зваженої суми.

Методи, визначені користувачем, повинні бути задані або в існуючих m-файлах, або у формі розроблених користувачем окремих m-файлів, розміщених в папці C:\MATLAB\toolbox\fuzzy або в папці C:\MATLAB\work (якщо система MATLAB встановлена за умовчанням на диску C).

У правому нижньому кутку знаходяться кнопка виклику вбудованої довідкової системи MATLAB (**Help**) і кнопка закриття редактора FIS (**Close**).

1.5 Редактор функцій приналежності Membership Function Editor

Редактор функцій приналежності, як випливає з його назви, призначений для завдання і редагування функцій приналежності окремих термів системи нечіткого висновку в графічному режимі. Редактор функцій приналежності може бути відкритий за допомогою введення функції **mfedit**, а також **mfedit('a')** або **mfedit(a)**, у вікні команд або за допомогою головного меню редактора FIS (командою меню **EditMembership Functions...** або натисненням клавіш Ctrl+2).

Ця функція, записана у форматі **mfedit**, просто викликає редактор функцій приналежності без завантаження якої б то не було системи нечіткого висновку. Фу-

нкція у форматі `mfedit('a')` викликає редактор функцій приналежності, який дозволяє користувачеві в графічному режимі аналізувати і модифікувати всі функції приналежності деякої структури FIS, збереженої в зовнішньому файлі з ім'ям `a.fis`. Функція у форматі `mfedit(a)` працює із змінною робочого простору MATLAB, відповідній структурі **FIS** з ім'ям `a`. Для кожної функції приналежності можна змінити її ім'я, тип і параметри. Редактор надає користувачеві не тільки можливість вибрати будь-яку з 11 вбудованих функцій приналежності, але і задати власну функцію приналежності.

Результат виклику редактора функцій приналежності за допомогою функції `mfedit('Kran')` зображений на рис. 1.3.

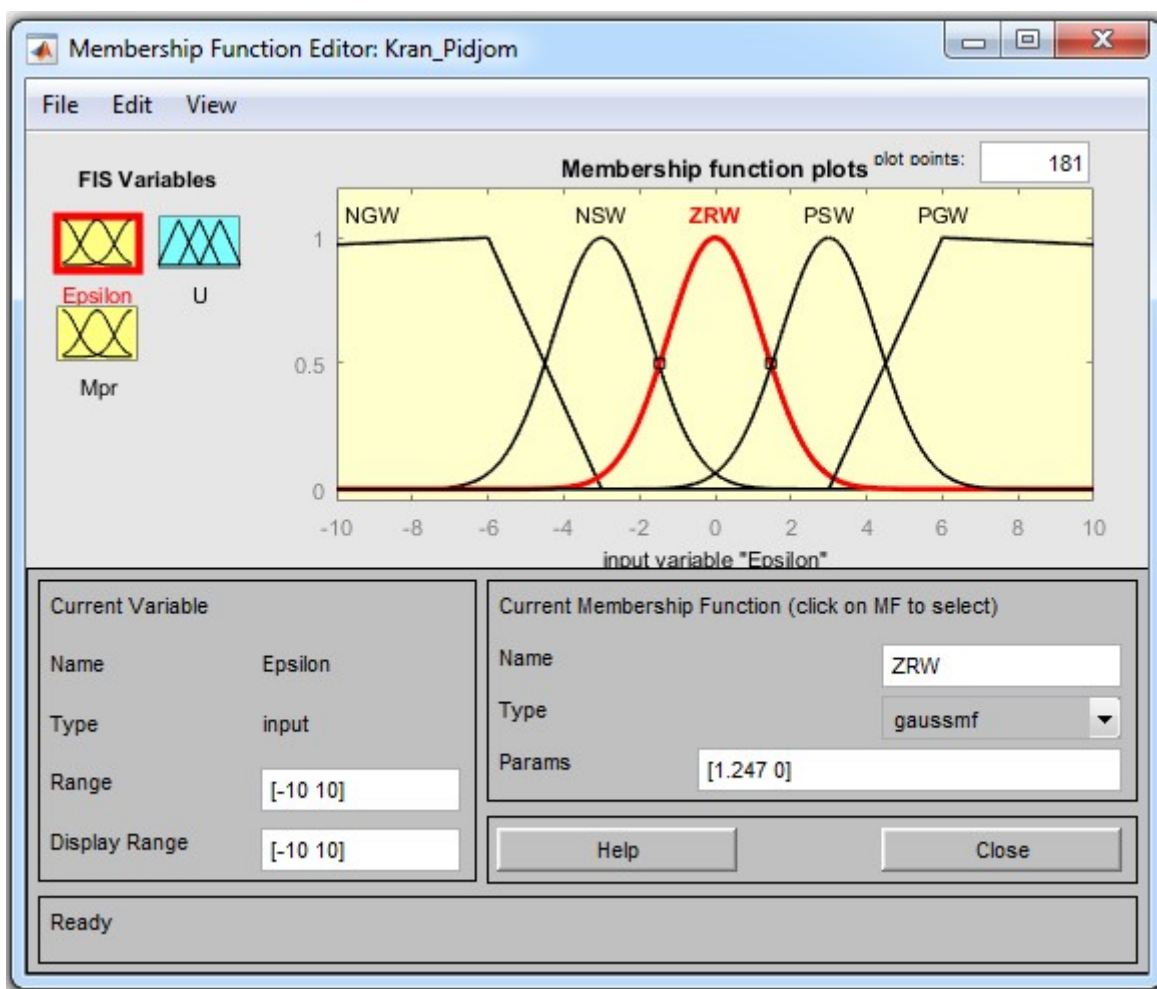


Рис. 1.3. Редактор функцій приналежності, що викликається функцією `mfedit('Kran_Pidjom')`

Для відображення графіків функцій приналежності слід вибрати необхідну змінну в лівій частині графічного інтерфейсу редактора під заголовком **FIS Variables** (Змінні FIS). Щоб вибрати потрібну функцію приналежності, слід клацнути на ній або її мітці в основному вікні з графіками функцій приналежності.

Редактор функцій приналежності має головне меню програми, яке дозволяє користувачеві викликати інші графічні засоби роботи з системою нечіткого виведення **FIS**, завантажувати і зберігати структуру **FIS** в зовнішніх файлах і т.д.

- Пункт меню **File** (Файл) редактора функцій приналежності містить такі ж операції, що і відповідний пункт меню редактора **FIS**.
- Пункт меню **Edit** (Редагування) містить наступні операції:
 - **Undo** – відмінняє виконання останньої дії;
 - **Add MF...** – дозволяє додати вбудовану функцію приналежності термів для вибраної змінної;
 - **Add Custom MF...** – дозволяє додати призначену для користувача функцію приналежності для окремої змінної;
 - **Remove Current MF** – дозволяє видалити окрему функцію приналежності;
 - **Remove All MFs** – дозволяє видалити всі функції приналежності для окремої змінної;
 - **FIS Properties...** – викликає редактор **FIS**;
 - **Rules...** – викликає редактор правил нечіткого висновку.
- Пункт меню **View** (Вигляд) містить наступні операції:
 - **Rules** – викликає програму перегляду правил нечіткого висновку;
 - **Surface** – викликає програму перегляду поверхні нечіткого висновку.

Список типів функцій приналежності, що розкривається, дозволяє вибрати одну з 11 вбудованих функцій приналежності. Використовуючи відповідні поля введення, можна змінити імена термів вибраної змінної в полі введення **Name**, модифікувати параметри вбудованих функцій приналежності в полі введення **Params**.

Оскільки даний редактор не дозволяє задати функцію приналежності, визначену користувачем, у разі подібної необхідності слід скористатися відповідними функціями командного режиму. Проте, вбудованих типів функцій приналежності виявляється цілком достатньо для більшості практичних застосувань.

Змінити вид функції приналежності можна також за допомогою миші. Для цього слід виділити змінну функцію приналежності на графіці (вона буде зображена червоним кольором) і, не відпускаючи натиснуту ліву кнопку миші, переміщати маркер в потрібну сторону. При цьому змінюватимуться графік відповідної функції приналежності і її параметри. Цією можливістю слід користуватися з обережністю, оскільки виконані зміни функції приналежності вже не вдасться відмінити. У правому нижньому кутку знаходяться кнопка виклику вбудованої довідкової системи **MATLAB (Help)** і кнопка закриття редактора функцій приналежності (**Close**).

1.6 Редактор правил системи нечіткого висновку Rule Editor

Редактор правил системи нечіткого висновку, як випливає з його назви, призначений для завдання і редагування окремих правил системи нечіткого висновку в графічному режимі. Редактор правил може бути відкритий за допомогою введення функції ruleedit('a') або ruleedit(a) у вікні команд або за допомогою головного меню редактора FIS (командою меню **Edit\Rules...** або натисненням клавіш Ctrl+3).

Ця функція, записана у форматі ruleedit('a'), викликає редактор правил, який дозволяє користувачеві в графічному режимі аналізувати і модифікувати правила продукції системи нечіткого виведення FIS, збереженої в зовнішньому файлі з ім'ям a.fis. Ця функція дозволяє також виконувати граматичний аналіз правил, які використовуються в деякій системі нечіткого виведення FIS.

Щоб використовувати даний редактор для створення правил, необхідно заздалегідь визначити всі вхідні і вихідні змінні, для чого можна скористатися редактором системи нечіткого виведення FIS і редактором функцій приналежності. При цьому задати правила можна за допомогою вибору відповідних значень термів вхідних і вихідних змінних.

Результат виклику функції ruleedit('Kran_Pidjom') зображений на рис. 1.4.

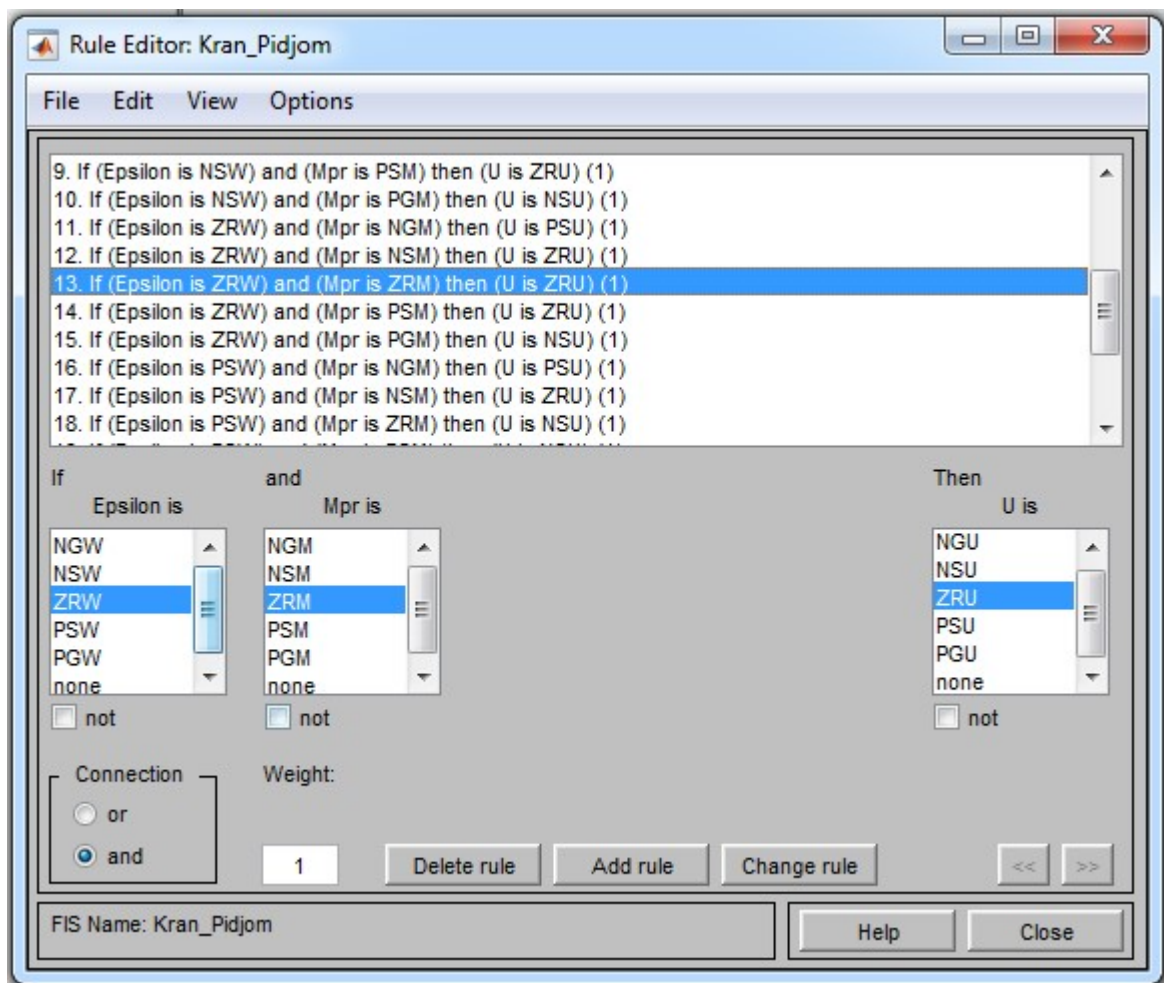


Рис. 1.4. Редактор правил, що викликається функцією ruleedit('Kran_Pidjom')

Функція у форматі ruleedit(a) викликає редактор правил для змінної робочого простору MATLAB, відповідній структурі FIS з ім'ям a.

Редактор правил має головне меню, яке дозволяє користувачеві викликати інші графічні засоби роботи з системою нечіткого виведення FIS, завантажувати і зберігати структуру FIS в зовнішніх файлах і т.д.

- Пункт меню **File** (Файл) редактора правил містить такі ж операції, що і відповідний пункт меню редактора FIS.
- Пункт меню **Edit** (Редагування) містить наступні операції:
 - **Undo** – відміняє виконання останньої дії;
 - **FIS Properties...** – викликає редактор FIS;
 - **Membership Functions...** – викликає редактор функцій приналежності.
- Пункт меню **View** (Вигляд) містить наступні операції:
 - **Rules** – викликає програму перегляду правил;
 - **Surface** – викликає програму перегляду поверхні висновку.
- Пункт меню **Options** (Сервіс) містить наступні операції:
 - **Language** – дозволяє вибрати мову для запису правил у формі тексту: English (англійський), Deutsch (німецький) або Francais (французький);
 - **Format** – дозволяє вибрати формат запису правил системи нечіткого висновку: Verbose (у формі тексту), Symbolic (у символічній формі) або Indexed (у цифровій формі).

При записі правил у формі тексту для створення закінчених пропозицій використовуються службові слова "if", "then", "is", "AND", "OR" і т.д. При записі правил в символічній формі ці службові слова замінюються символами відповідних операцій. Наприклад, правило "if (A is A) and (B is B) then (C is C)" перетвориться до вигляду: "(A == A) & (B == B) => (C == C)". Правила нечіткого висновку, записані в цифровій формі, відповідають формату їх уявлення в структурі FIS, який розглядається в спеціальній літературі.

Поля введення в середній частині графічного інтерфейсу редактора правил дозволяють задати нове правило в системі нечіткого висновку. Для цього необхідно виділити ім'я терма відповідної змінної, яка повинна бути заздалегідь визначена за допомогою редактора функцій приналежності. Якщо деякий терм не входить в правило, то для нього слід вибрати значення "none". Якщо в умові правила використовується логічне заперечення деякого терма, то для цього терма слід зазначити відповідний прапорець з міткою "not" ("виставити галочку").

Редактор правил дозволяє також задати логічні зв'язки для підумов правила (перемикач **Connection**) і вагу правила (поле введення **Weight**). Кнопки в нижній частині графічного інтерфейсу редактора правил, як впливає з їх назв, служать для видалення виділеного у вікні правила (**Delete rule**), додавання створеного правила в систему (**Add rule**) і внесення змін у виділене у вікні правило (**Change rule**). У правому нижньому кутку знаходяться кнопки виклику вбудованої довідкової системи MATLAB (**Help**) і кнопка закриття редактора правил (**Close**).

1.7 Програма перегляду правил системи нечіткого висновку Rule Viewer

Головне призначення програми перегляду правил полягає в можливості візуалізувати результати нечіткого висновку і отримати значення вихідних змінних залежно від початкових значень вхідних змінних. Графічний інтерфейс програми перегляду правил може бути відкритий за допомогою введення функції `ruleview('a')` або `ruleview(a)` у вікні команд або за допомогою головного меню редактора **FIS**, редактора функцій приналежності або редактора правил (командою меню **View\Rules** або натисненням клавіш `Ctrl+5`).

Функція, записана у форматі `ruleview('a')`, викликає програму перегляду правил, яка зображає діаграму нечіткого висновку для структури **FIS**, збереженої в зовнішньому файлі з ім'ям `a.fis`. Функція у форматі `ruleview(a)` викликає програму перегляду правил для змінній робочого простору **MATLAB**, відповідній структурі **FIS** з ім'ям `a`.

Програма перегляду правил не дозволяє редагувати правила і функції приналежності термів змінних і використовується після розробки системи нечіткого висновку на етапі її аналізу і оцінки. Функцію також доцільно використовувати у тому випадку, коли необхідно візуально представити весь процес нечіткого висновку від початку до кінця. При цьому користувач має можливість оцінити значення вихідних змінних нечіткої моделі і вплив кожного з правил на результат нечіткого висновку за допомогою зміни значень вхідних змінних.

Графічний інтерфейс програми перегляду правил зображений на рис. 1.5.

Програма перегляду правил має головне меню, яке дозволяє користувачеві викликати інші графічні засоби роботи з системою нечіткого виведення **FIS**, завантажувати і зберігати структуру **FIS** в зовнішніх файлах і т.д.

- Пункт меню **File** (Файл) редактора правил містить такі ж операції, що і відповідний пункт меню редактора **FIS**.
- Пункт меню **Edit** (Редагування) містить наступні операції:
 - **Rules** – викликає програму перегляду правил;
 - **Undo** – відмінняє виконання останньої дії;
 - **FIS Properties...** – викликає редактор **FIS**;
 - **Membership Functions...** – викликає редактор функцій приналежності;
 - **Rules...** – викликає програму редагування правил.
- Пункт меню **View** (Вигляд) містить наступні операції:
 - **Surface** – викликає програму перегляду поверхні висновку.
- Пункт меню **Options** (Сервіс) містить наступні операції:
 - **Formst** – дозволяє вибрати формат запису правил системи нечіткого висновку; **Verbose** (у формі тексту), **Symbolic** (у символічній формі) або **Sndexed** (в цифровій формі).

У центральній частині графічного інтерфейсу програми поглядання правил розташовані прямокутники, що відповідають окремим вхідним змінним (функції при-

належності жовтого кольору) і вихідні змінні (функції приналежності синього кольору) правил нечіткого висновку.

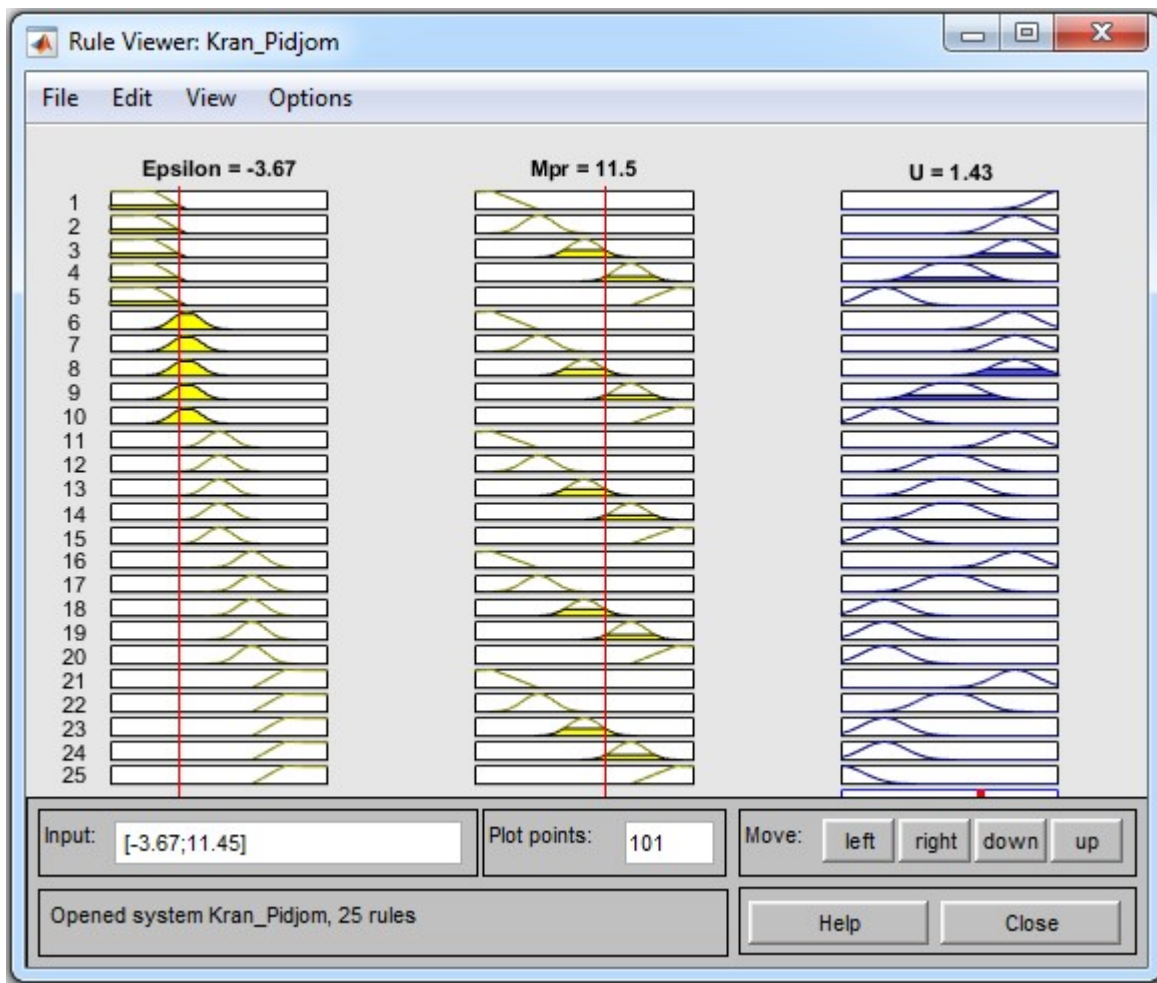


Рис. 1.5. Програма перегляду правил, викликана функцією ruleview('Kran_Pidjom')

В правій нижній частині графічного інтерфейсу розташований прямокутник, що зображає дефазифікацію вихідної змінної після акумулювання всіх правил нечіткого висновку. Отримане в результаті дефазифікації значення вихідної змінної вказується у верхній частині стовпця з іменем цієї змінної.

Прямокутники вхідних змінних перетинає вертикальна пряма червоного кольору, положення якої відповідає конкретному значенню вхідної змінної відповідного стовпця. Задати конкретні значення вхідних змінних можна або за допомогою їх запису в полі введення **Input**, або за допомогою миші, переміщаючи вертикальні прямі в потрібному напрямі. У останньому випадку можна клацнути на тій або іншій вертикальній прямій і, утримуючи натиснутою ліву кнопку миші, перемістити пряму управо або вліво, або просто клацнути в необхідній точці усередині прямокутника відповідної вхідної змінної.

Отримані після зміни значення вхідних змінних безпосередньо відображаються у верхній частині прямокутників після імені вхідних змінних і в полі введення з міткою **Input**. Більш того, система MATLAB реагує на кожну зміну значення окремої вхідної змінної виконанням процедури нечіткого висновку, отриманням і відображенням відповідних результуючих значень вихідних змінних.

1.8 Програма перегляду поверхні системи нечіткого висновку Surface Viewer

Програма перегляду поверхні системи нечіткого висновку дозволяє проглядати поверхню системи нечіткого висновку і візуалізувати графіки залежності вихідних змінних від окремих вхідних змінних. Графічний інтерфейс програми перегляду правил може бути відкритий за допомогою введення функції `surfview('a')` або `surfview(a)` у вікні команд або за допомогою головного меню редактора **FIS**, редактора функцій приналежності або редактора правил (командою меню **View\Surface** або натисненням клавіш `Ctrl+6`).

Функція, записана у форматі `surfview('a')`, викликає програму перегляду поверхні, яка зображає поверхню нечіткого висновку для структури **FIS**, збереженої в зовнішньому файлі з ім'ям `a.fis`, для будь-якої однієї або двох з її вхідних змінних. Функція у форматі `surfview(a)` викликає програму перегляду поверхні висновку для змінної робочого простору **MATLAB**, що відповідає структурі **FIS** з ім'ям `a`. Графічний інтерфейс програми перегляду поверхні зображений на рис. 1.6.

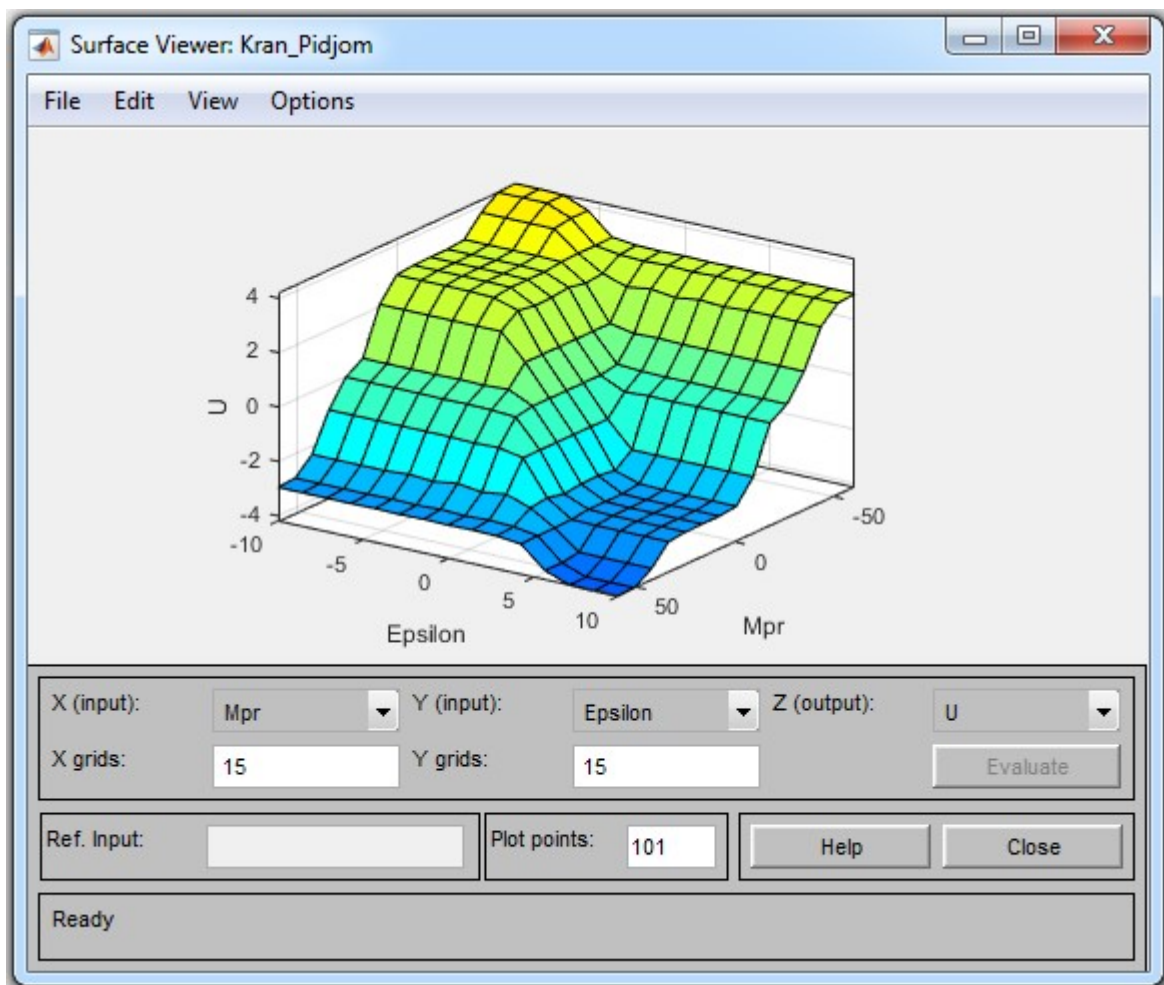


Рис. 1.6. Програма перегляду поверхні висновку, що викликається функцією `surfview('Kran_Pidjom')`

Програма перегляду поверхні висновку має головне меню, яке дозволяє користувачеві викликати інші графічні засоби роботи з системою нечіткого виведення **FIS**, завантажувати і зберігати структуру **FIS** в зовнішніх файлах і т.д.

- Пункт меню **File** (Файл) редактора правил містить такі ж операції, що і відповідний пункт меню редактора **FIS**.
- Пункт меню **Edit** (Редагування) містить наступні операції:
 - **Undo** – відмінює виконання останньої дії;
 - **FIS Properties...** – викликає редактор **FIS**;
 - **Membership Functions...** – викликає редактор функцій приналежності;
 - **Rules...** – викликає програму редагування правил
- Пункт меню **View** (Вигляд) містить наступні операції:
 - **Rules** – викликає програму перегляду правил.
- Пункт меню **Options** (Сервіс) містить наступні операції:
 - **Plot** – дозволяє вибрати один з 8 стилів зображення графіка поверхні висновку;
 - **Color Map** – дозволяє вибрати одну з 4 кольорних схем зображення графіка поверхні висновку;
 - **Always evaluate** – позначка галочкою цього пункту вкладеного меню приводить до автоматичного формування нової поверхні висновку всякий раз, коли вносяться зміни в систему нечіткого висновку, що впливають на форму графіка поверхні висновку (такі як зміна кількості точок сітки графіка). Це значення прийняте за умовчанням. Щоб його відмінити, необхідно зняти галочку у цього пункту вкладеного меню, клацнувши на цій позиції меню.

Програма перегляду поверхні висновку не дозволяє вносити зміни в систему нечіткого висновку і відповідну їй структуру **FIS**. Використовуючи головне меню програми, користувач може вибрати вхідні змінні і відповідні їм горизонтальні осі системи координат (X і Y), а також вихідну змінну, якій відповідає вертикальна вісь системи координат (Z).

Клацнувши і утримуючи ліву кнопку миші на осях графіка поверхні, за допомогою подальшого переміщення курсора миші в тому або іншому напрямі можна змінити кут перегляду поверхні висновку. Якщо розглядається система нечіткого висновку з більш ніж двома вхідними змінними, то для вхідних змінних, що не візуалізуються, слід задати деякі постійні значення (константи).

1.9 Контрольні питання по темі заняття

1. Яке призначення пакету прикладних програм **Fuzzy Logic Toolbox** системи **MATLAB**?
2. Перечисліть основні програми, що входять до складу **Fuzzy Logic Toolbox**, які дозволяють працювати в режимі **GUI**?
3. Поясніть призначення і можливості редактора систем нечіткого виведення **FIS Editor**.
4. Поясніть призначення і можливості редактора функцій приналежності **Membership Function Editor**.
5. Поясніть призначення і можливості редактора правил системи нечіткого висновку **Rule Editor**.
6. Поясніть призначення і можливості програми перегляду правил системи нечіткого висновку **Rule Viewer**.
7. Поясніть призначення і можливості програми перегляду поверхні відгуку системи нечіткого висновку **Surface Viewe**.

Практичне заняття 2

ПОБУДОВА НЕЧІТКИХ МОДЕЛЕЙ БАГАТОМАСОВИХ ЕЛЕКТРОМЕХАНІЧНИХ СИСТЕМ ЗА ДОПОМОГОЮ ПАКЕТУ FUZZY LOGIC TOOLBOX СИСТЕМИ MATLAB

2.1 Мета заняття

Набуття практичних навичок по застосовувати пакету **Fuzzy Logic Toolbox** системи **MATLAB** для вирішення конкретних інженерних задач - побудови нечітких моделі багатомасових електромеханічних систем, що передбачає налаштування бази правил, визначення функцій належності та оцінки результатів. Заняття дозволить практично застосувати системи нечіткого висновку для моделювання складних технічних систем.

2.2 Методика синтезу нечіткої моделі двомасової електромеханічної системи

Розглянемо можливість вирішення задачі ідентифікації за допомогою нечіткої апроксимуючої системи. Побудуємо нечітку систему, що відображає залежність між вхідною і вихідною змінною двомасової електромеханічної системи управління механізмом підйому мостового крана.

У якості привідного двигуна застосовано двигун постійного струму незалежного збудження. Якірна обмотка живиться від тиристорного підсилювача потужності. Систем управління електроприводом побудована за принципом підлеглого регулювання і містить два контури: контур регулювання струму і контур регулювання швидкості із зворотним зв'язком по ЕДС двигуна. Замкнений контур струму настроєний на модульний оптимум, а контур ЕДС – на симетричний оптимум.

Двомасової системи підлеглого регулювання швидкості з внутрішнім контуром струму і зовнішнім контуром ЕРС описується системою диференціальних рівнянь (2.1). При запису системи прийняті наступні позначення: ω_d , ω_m – кутові швидкості двигуна і механізму; M_d – обертаючий момент двигуна; $M_{ст}$ – момент статичного навантаження; $M_{пр}$ – момент пружності; $P = \frac{dM_d}{dt}$ – ривок; $U_{зе}$, $E_{зе}$ – напруга і ЕДС завдання; $U_{звЕ}$ – напруга зворотного зв'язку по ЕДС; $U_{іе}$ – вихідна напруга інтегратора III регулятора ЕДС; $T_{\muт}$ – мала некомпенсуєма постійна часу контуру струму; T_a – електромеханічна постійна часу електродвигуна; k_d – коефіцієнт посилення двигуна: $k_d = \frac{1}{k\Phi_n}$; $k\Phi_n$ – добуток конструктивного коефіцієнта і магнітного потоку

двигуна; k_T – коефіцієнт посилення зворотного зв'язку по току; k_H – коефіцієнт посилення зворотного зв'язку по напрузі; β – коефіцієнт внутрішнього в'язкого тертя; c_{12} – коефіцієнт жорсткості передачі; J_{Σ} – сумарний момент інерції двигуна і жорстко пов'язаних з ним елементів системи; J_M – момент інерції механізму; $k_{пе}$, $k_{іе}$ – коефіцієнти підсилення пропорційної і інтегральної частин регулятора ЕДС.

$$\left\{ \begin{array}{l} \frac{d\omega_M}{dt} = \frac{1}{J_M} M_{пр} + \frac{\beta}{J_M} \omega_{д} - \frac{\beta}{J_M} \omega_M - \frac{1}{J_M} M_{ст}; \frac{dM_{пр}}{dt} = c_{12} \omega_{д} - c_{12} \omega_M; \\ \frac{d\omega_{д}}{dt} = \frac{1}{J_{\Sigma}} M_{д} - \frac{1}{J_{\Sigma}} M_{пр} - \frac{\beta}{J_{\Sigma}} \omega_{д} - \frac{\beta}{J_{\Sigma}} \omega_M; \\ \frac{dM_{д}}{dt} = \frac{1}{k_{д}} P; \\ \frac{dP}{dt} = \frac{k_{пе}}{2T_{\mu T}^2 k_T} E_{3е} - \frac{k_{пе}}{2T_{\mu T}^2 k_T} U_{33е} - \frac{k_{д}}{2T_{\mu T}^2} M_{д} - \frac{1}{T_{\mu T}} P + \frac{1}{T_{\mu T}^2 k_T} U_{іе}; \\ \frac{dE_{3е}}{dt} = \frac{1}{T_a} U_{3е} - \frac{1}{T_a} E_{3е}; \\ \frac{dU_{33е}}{dt} = \frac{k_H}{T_a k_{д}} \omega_{д} - \frac{1}{T_a} U_{33е}; \\ \frac{dU_{іе}}{dt} = k_{іе} E_{3е} - k_{іе} U_{33е}. \end{array} \right. \quad (2.1)$$

Значення параметрів $k_{пе}$, $k_{іе}$ – можуть бути отримані з виразу передатної функції регулятора ЕДС:

$$W_{пе}(p) = \frac{T_M k_T}{2T_{\mu E} k_H R_{\Sigma}} \cdot \frac{4T_{\mu E} p + 1}{4T_{\mu E} p} = \frac{T_M k_T}{2T_{\mu E} k_H R_{\Sigma}} + \frac{T_M k_T}{8T_{\mu E}^2 k_H R_{\Sigma}} \frac{1}{p} = k_{пе} + \frac{k_{іе}}{p},$$

$$k_{пе} = \frac{T_M k_T}{2T_{\mu E} k_H R_{\Sigma}}, \quad k_{іе} = \frac{T_M k_T}{8T_{\mu E}^2 k_H R_{\Sigma}}.$$

T_M – електромеханічна постійна часу системи ТП-Д; при записі системи (2.1) замість T_M записано вираз $T_M = J_{\Sigma} \frac{R_{\Sigma}}{(k\Phi_H)^2}$, де $J_{\Sigma} = J_{\Sigma д} + J_M$ - сумарний момент інерції електроприводу: $J_{\Sigma} = J_{\Sigma д} + J_M$.

Алгоритмічна схема двомасової системи зображена на рис.2.1

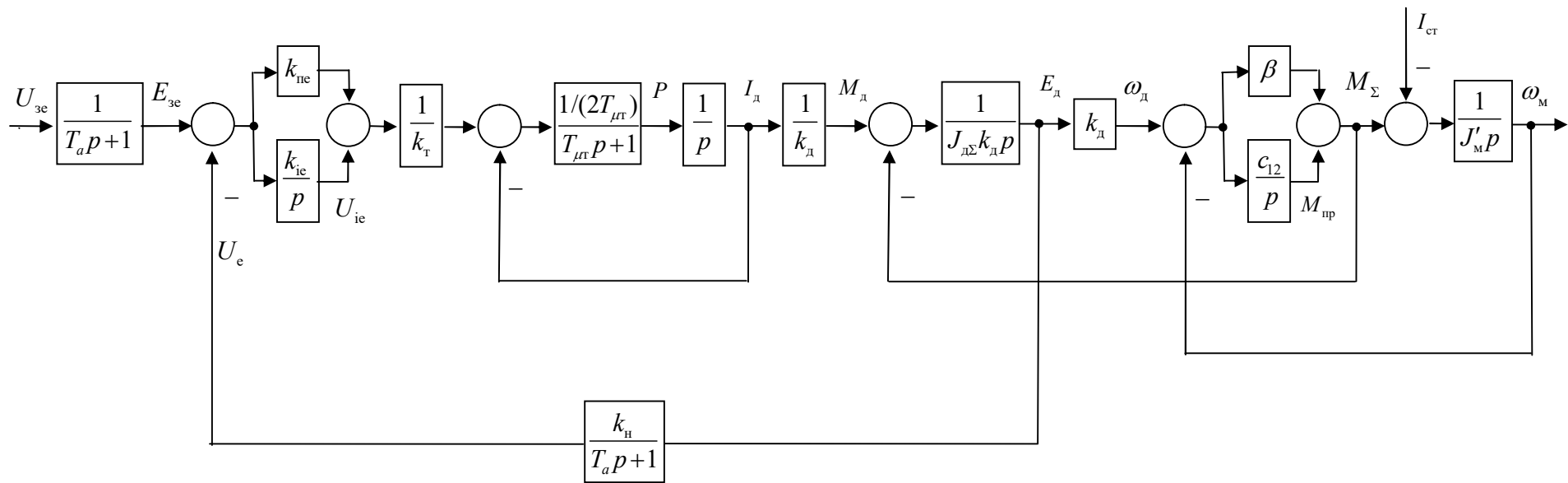


Рис. 2.1. Алгоритмічна схема двомасової системи

У даному прикладі вхідну послідовність формуємо на основі поточного значення вхідного сигналу системи $U_{zE}(k)$ і вхідного сигналу, затриманого на один шаг дискретності $U_{zE}(k-1)$, а також двох затриманих на один і два кроки вихідних сигналів, тобто $\omega_m(k-1)$ і $\omega_m(k-2)$ відповідно. Вихідним сигналом є швидкість механізму $\omega_m(k)$. Ці дані для ряду характерних точок візьмемо з вище приведеного прикладу побудови нейромережевої моделі двомасової системи. Точки повинні охоплювати весь діапазон зміни вхідного і вихідного сигналів системи.

2.3 Завдання системи нечіткого висновку з використанням вікна FIS Editor

Командою (функцією) **Fuzzy** з режиму командного рядка запускаємо основну інтерфейсну програму пакету **Fuzzy Logic** – редактор нечіткої системи висновку (**FIS**-редактор). Вид вікна, що відкривається при цьому, приведений вище на рис. 2.1. В меню **File** виберемо команду **New FIS\Sugeno** (нова система типу **Sugeno**), при цьому в блоці, що відображається білим квадратом в верхній частині вікна редактора, з'явиться надпис **Untitled2 (sugeno)**.

В системі повинні бути чотири входи, тому через пункт меню **Edit\Add Variable...\Input** додаємо в систему ще три входи (у вікні редактора з'являються другий, третій і четвертий блоки з іменами **input2**, **input3**, **input4**). Роблячи, далі, одноразове клацання на блоку **input 1**, міняємо його ім'я на «**UzE(k)**» (поточне значення вхідного сигналу системи $U_{zE}(k)$), завершуючи введення нового імені натисненням клавіші **Enter**. Аналогічним чином встановлюємо ім'я «**UzE(k-1)**» ($U_{zE}(k-1)$) блоку **input2**, «**Wm(k-1)**» ($\omega_m(k-1)$) блоку **input3**, «**Wm(k-2)**» ($\omega_m(k-2)$) блоку **input4** і «**Wm(k)**» (вихідний сигнал системи $\omega_m(k)$) – вихідному блоку (справа вгорі) **output 1**.

Надамо відразу ж і ім'я **FuzzyModel** всій системі і запишемо її на жорсткому диску у файлі з тим же ім'ям, виконавши це через пункт меню **File\Export\To Disk...** (зберегти в робочому просторі), а також експортуємо систему у робочий простір **MATLAB** через пункт меню **File\Export\To Workspace...** (зберегти в робочому просторі). Вид вікна редактора після вказаних дій приведений на рис.2.2.

2.4 Завдання функцій приналежності змінних системи

Задамо тепер функції приналежності змінних. Вкажемо, що програму – редактор функцій приналежності можна відкрити трьома способами:

- через пункт меню **Edit\Membership Functions**;
- подвійним клацанням на значку, що відображає відповідну змінну;
- натисненням клавіш **Ctrl+2**.

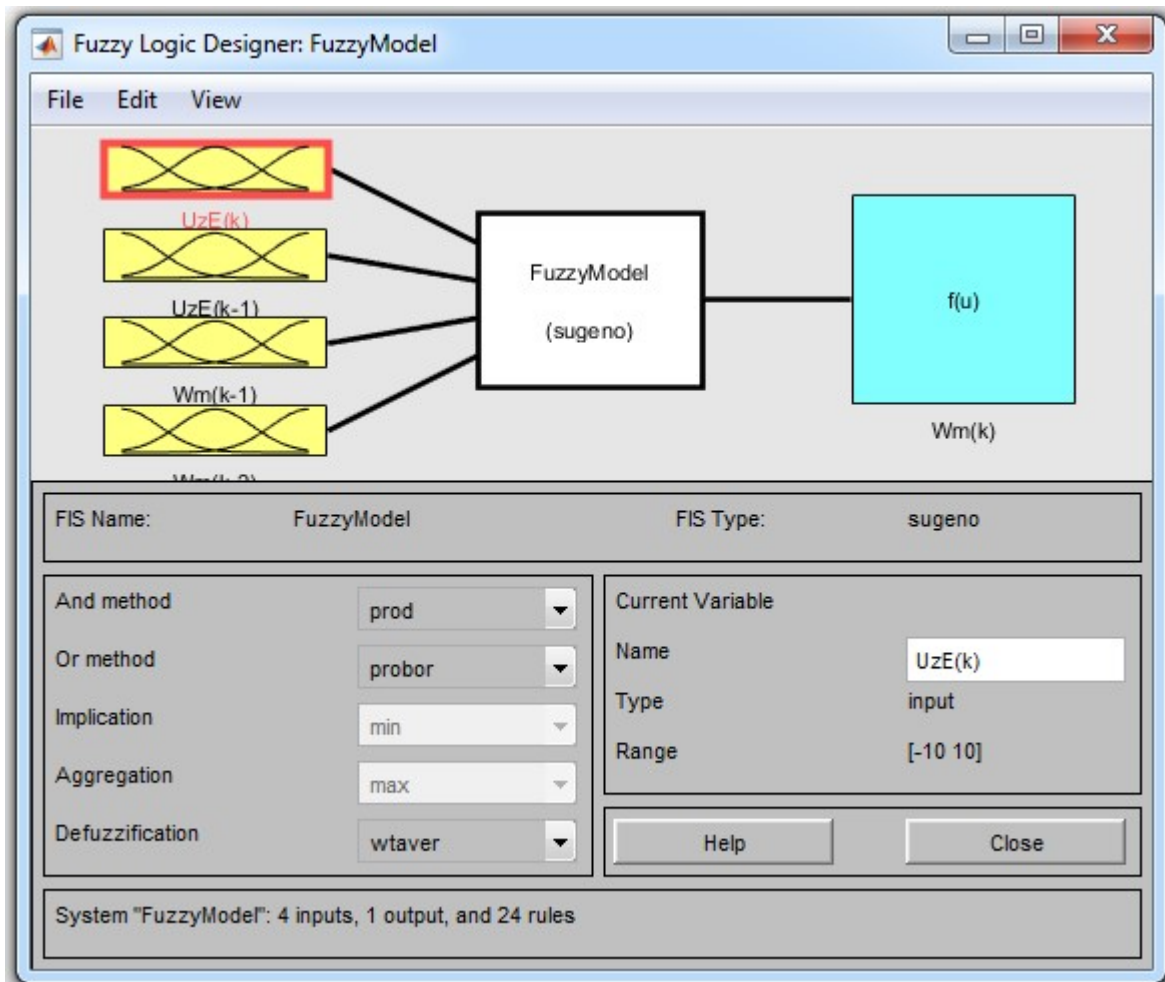


Рис. 2.2. Вид вікна **FIS Editor** після завдання структури системи

Будь-яким з приведених способів перейдемо до даної програми. Відкриється вікно редактора функцій приналежності – **Membership Functions Editor** (рис. 2.3). Завдання і редагування функцій приналежності почнемо із змінній «UZE(k)». Відкриємо меню **Edit** даного редактора і виберемо спочатку команду **Remove All MFs** (Видалити всі функції приналежності) – для видалення установок за умовчанням, а потім команду **Add MFs** (Додати функції приналежності). При цьому з'явиться вікно рис. 2.4, що дозволяє задати тип (**MF type**) і кількість (**Number of MFs**) функцій приналежності. Виберемо гаусові функції приналежності (**gaussmf**), а їх кількість задамо 9 – за кількість характерних точок вхідної послідовності. (кількість функцій, їх тип і параметри задалегідь визначити не можливо, тому спочатку вони приймаються на основі досвіду дослідника, а потім уточнюються в процесі моделювання; в прикладі, що розглядається, було добавлено ще 3 точки). Підтвердимо завдання інформації натисканням кнопки **ОК**, після чого відбудеться повернення до редактора функцій приналежності.

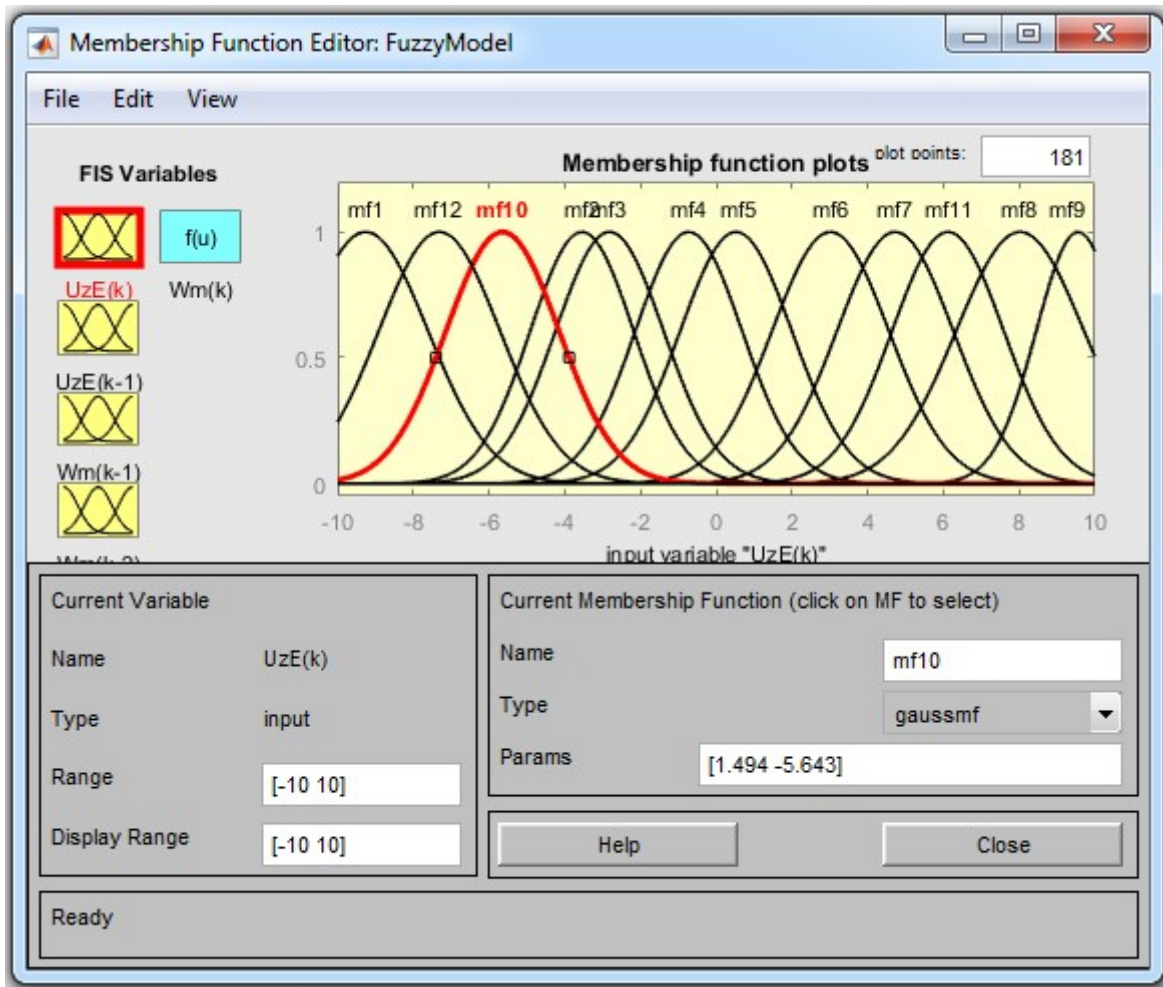


Рис. 2.3. Вікно редактора функцій приналежності (змінної «UzE(k)» ($U_{zE}(k)$))

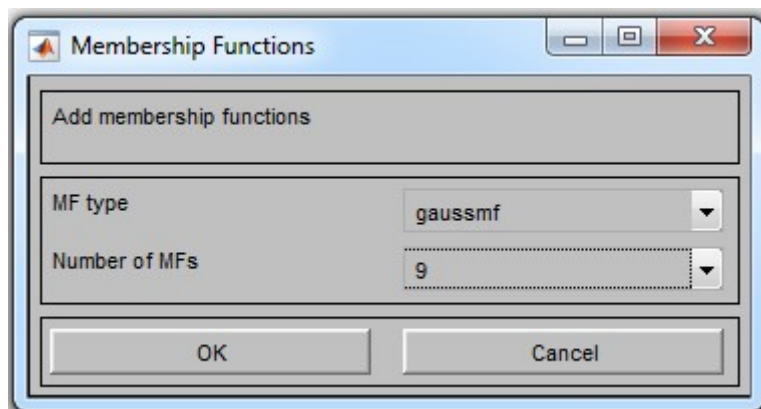


Рис. 2.4. Діалогове вікно завдання типу і кількості функцій приналежності

В полі **Range** (Діапазон) встановимо діапазон змінної $UzE(k)$ від -10 до $+10$ (як було зазначено в главі 6, діапазон зміни вхідного сигналу визначається при розрахунку системи підлеглого регулювання). Клацнемо потім лівою кнопкою миші десь в полі редактора (або натиснемо клавішу Enter), при цьому відбудеться відповідна зміна діапазону в полі **Display Range** (Діапазон відображення).

Виконаємо редагування функцій приналежності, що зображені в верхній частині вікна редактора функцій приналежності. Відмітимо, що для успішного рішення поставленої задачі необхідно, щоб ординатами максимумів цих функцій співпадали з заданими значеннями змінної $UzE(k)$, тобто функції необхідно «передвинути» вздовж осі абсцис. Для цього підводимо покажчик до потрібної кривої і клацаємо лівою кнопкою миші. Крива вибирається, забарвлюючись в червоний колір, після чого за допомогою миші її можна посунути в потрібну сторону (точнішу установку можна провести, змінюючи числові значення в полі **Params** (Параметри) – в даному випадку кожній функції приналежності відповідають два параметри, при цьому перший визначає розмах кривої, а другий – положення її центру). Крім цього, у полі **Name** можна змінити ім'я для вибраної кривої (завершуючи введення кожного імені натисненням клавіші Enter). В даному прикладі змінювати імена нема сенсу.

Аналогічно завдамо і відредагуємо функції приналежності змінних « $UzE(k-1)$ », « $Wm(k-2)$ », « $Wm(k-1)$ ». Діапазон змінних « $Wm(k-2)$ », « $Wm(k-1)$ » встановимо від -150 до $+150$ (рис. 2.5).

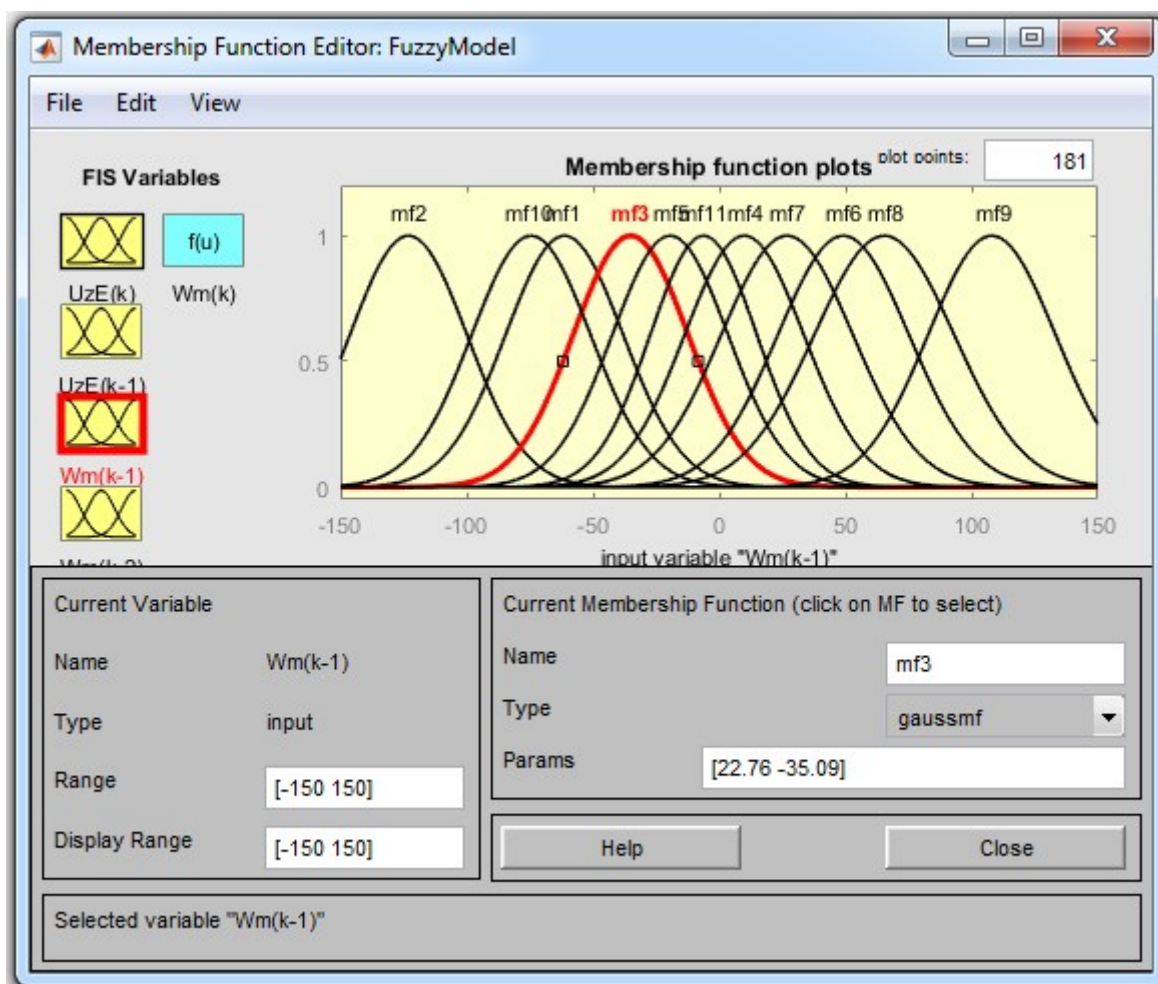


Рис. 2.5. Вікно редактора функцій приналежності (змінної $Wm(k-1)$ ($\omega_m(k-2)$))

Примітка. Кількість, тип, розмах і положення заданих функцій вибирається шляхом багатократного моделювання і порівняння модельних значень із значеннями, отриманими шляхом моделювання двомасової електромеханічної системи в Simulink.

Для вихідної змінної $W_m(k)$ ($\omega_m(k)$) також в меню **Edit** послідовно вибираємо команди **Remove All Mfs**, **Add MFs**. Діалогове вікно, що з'являється, виду, показаному на рис.2.6, дозволяє завдати тепер в якості функцій приналежності тільки лінійні (**linear**) або постійні (**constant**) – в залежності від того, який алгоритм Сугено (1-го чи 0-го порядку) вибирається. В задачі, що розглядається, потрібно вибрати постійні функції приналежності. Підтвердимо введення даних натисканням кнопки **OK**, після чого відбудеться повернення у вікно редактора функцій приналежності.

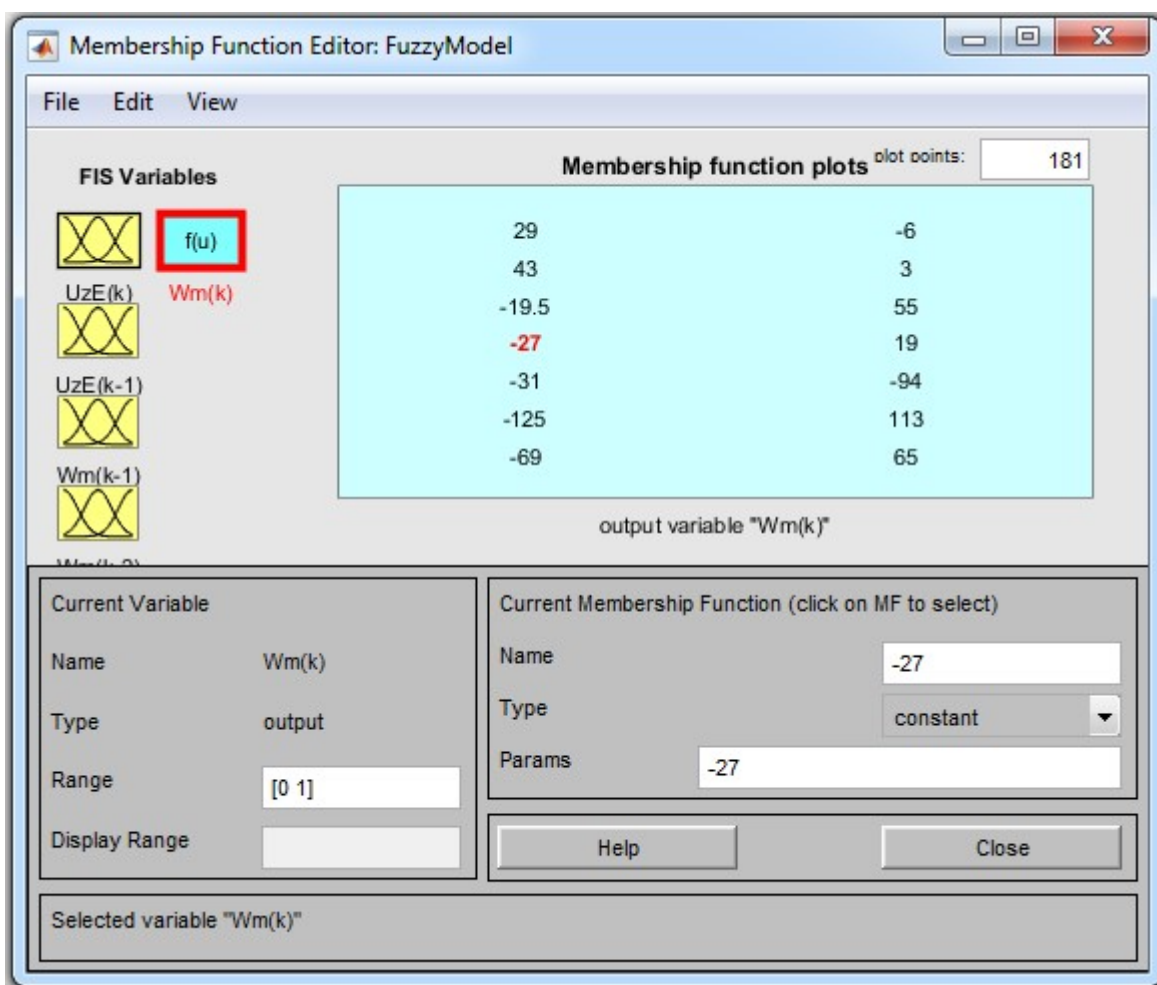


Рис. 2.6. Параметри функцій приналежності змінної $W_m(k)$ ($\omega_m(k)$)

Слід зазначити, що діапазон зміни (**Range**), що встановлений за умовчанням, – $[0, 1]$ – змінювати в даному випадку не потрібно. Змінимо імена функцій приналежності (їх графіки при використанні алгоритму Сугено для вихідних змінних не проводяться), наприклад, задавши їх як відповідні числові значення $W_m(k)$, тобто 29, 43, -19, 5 і т.д.; одночасно ці ж числові значення введемо у поле **Params** (рис.2.6).

Потім закриємо вікно натисканням кнопки **Close** і повернемося у вікно **FIS**-редактора.

2.5 Завдання правил нечіткого висновку

Два рази клацнемо на середньому (білому) блоці (те ж саме можна виконати, вибравши пункт меню **Edit\Rules**), при цьому розкриється вікно ще однієї програми – редактора правил **Rule Editor** (рис. 2.7). Введемо ряд правил. При вводі кожного правила необхідно позначити відповідність між кожною функцією приналежності вхідних аргументів і числовим значенням вихідної змінної. Наведемо приклад формування одного з правил.

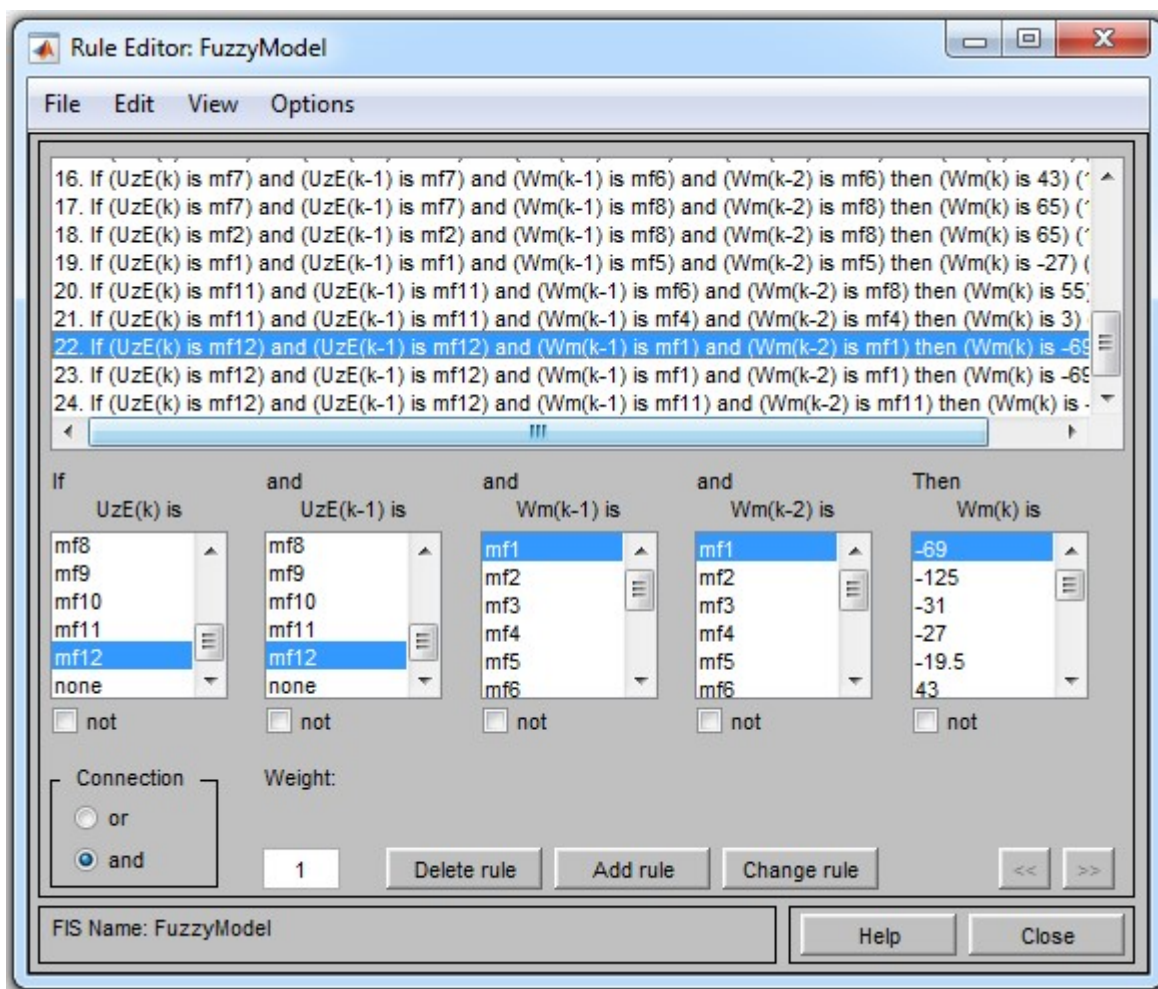


Рис. 2.7. Вікно редактора правил

З сформованої вхідної і вихідної послідовності (див. розділ 6.2), візьмемо наступні значення: $U_{zE}(k) = -7,17$, $U_{zE}(k-1) = -7,17$, $\omega_M(k-1) = -61,33$, $\omega_M(k-2) = -65,59$, $\omega_M(k) = -69$. Цим значенням відповідають наступні функції приналежності: $U_{zE}(k) \rightarrow mf12$, $U_{zE}(k-1) \rightarrow mf12$, $W_m(k-1) \rightarrow mf1$, $W_m(k-2) \rightarrow mf1$, $W_m(k) \rightarrow -69$. Тому виберемо в лівому нижньому полі з заголовком «UzE(k) is» варіант mf12, аналогічно в наступних полях: «UzE(k-1) is» – mf12, «Wm(k-1) is» –

mf1, «Wm(k-2) is» – mf1, «Wm(k) is» – -69 і натиснемо на кнопку **Add rule** (Добавити правило). Введене правило з'явиться у вікні правил і буде представляти собою наступний запис (у вікні рис.2.7 це правило має номер 22):

if UzE(k) is mf12 and UzE(k-1) is mf12 and Wm(k-1) is mf1 and Wm(k-2) is mf1 then Wm(k) is -69.

Аналогічно сформовано 24 правила (кількість правил уточнюється в процесі моделювання).

Кнопки **Change rule** і **Delete rule** дозволяють змінити і видалити правило відповідно.

2.6 Робота з вікнами перегляду правил і поверхні відгуку

Закриємо вікно редактора правил і повернемося у вікно **FIS** редактора. Побудову системи закінчено і можна почати експерименти по її дослідженню (попередньо збережемо на диску створену систему).

Розкриємо меню **View**. Виберемо команду **Rules**, при цьому відкриється вікно (рис. 2.8) ще однієї програми – перегляду правил (**Rule Viewer**).

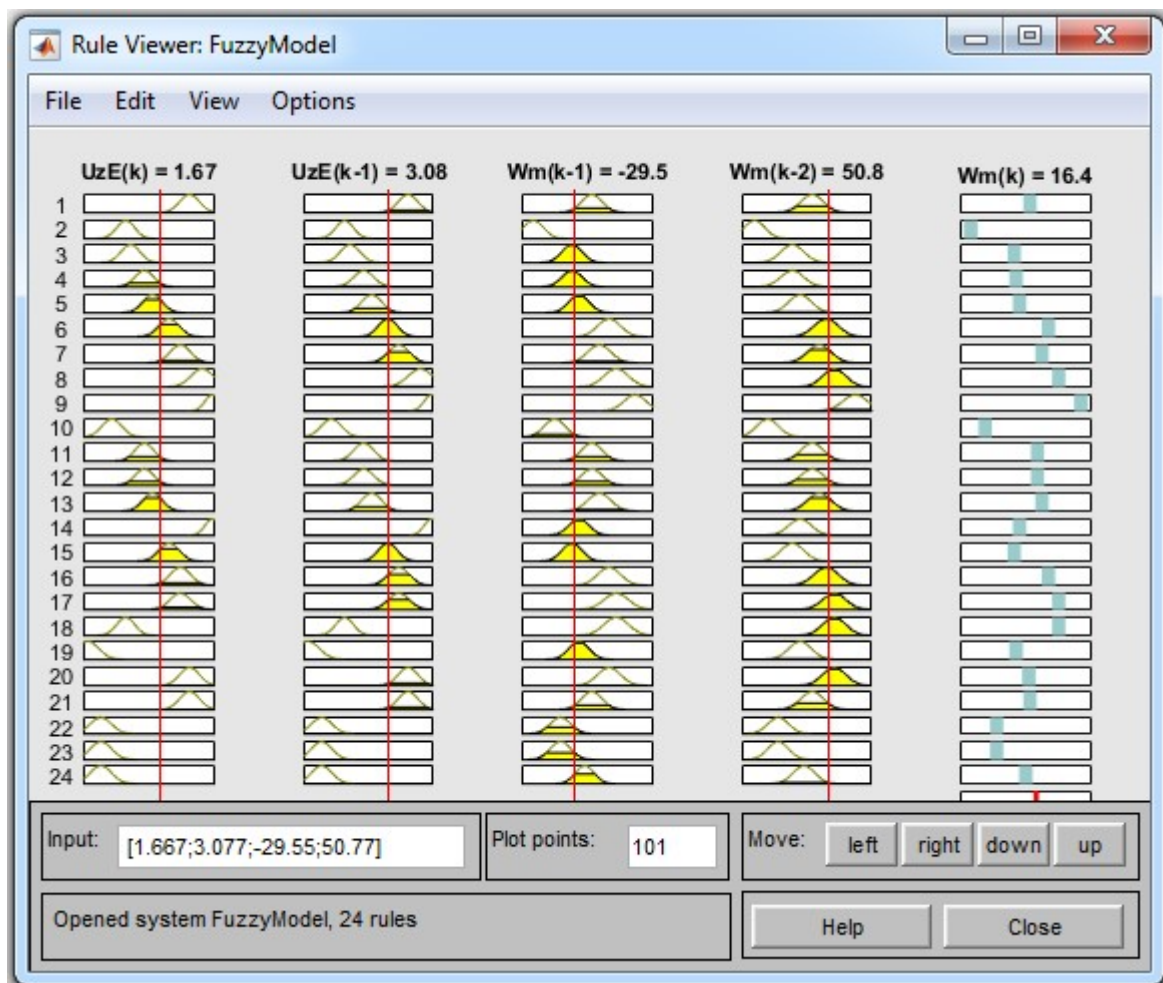


Рис. 2.8. Вікно перегляду правил

У лівій частині вікна в графічній формі представлені функції приналежності аргументів $U_{zE}(k)$ ($UzE(k)$), $U_{zE}(k-1)$ ($UzE(k-1)$), $\omega_m(k-1)$ ($Wm(k-1)$), $\omega_m(k-2)$ ($Wm(k-2)$), а в правій – функції приналежності змінної виходу $\omega_m(k)$ ($Wm(k)$) з поясненням механізму ухвалення рішення.

Червона вертикальна риска, що перетинає графіки в правій частині вікна, яку можна переміщати за допомогою миші, дозволяє змінювати значення змінних входу (це ж можна робити, задаючи числові значення в полі **Input** (Вхід) і натискаючи клавішу Enter), при цьому відповідно змінюються значення вихідної змінної у правій верхній частині вікна. Таким чином, за допомогою побудованої моделі і вікна перегляду правил можна вирішувати задачу інтерполяції, тобто задачу, рішення якої і потрібно було знайти. Зміна значень вхідної послідовностей шляхом переміщення червоної вертикальної лінії дуже наочно демонструє, як система визначає значення виходу.

Закриємо вікно перегляду правил і вибором команди меню **View\Surface** перейдемо до вікна перегляду поверхні відгуку (виходу) (рис.2.9). У вікні, що відкрилося, міняючи імена змінних в полях введення (X (input) і Y (input)) можна задати і проглядати залежність між будь-якими двома вхідними змінними і вихідною змінною. За допомогою мишки графік можна повертати на всі боки.

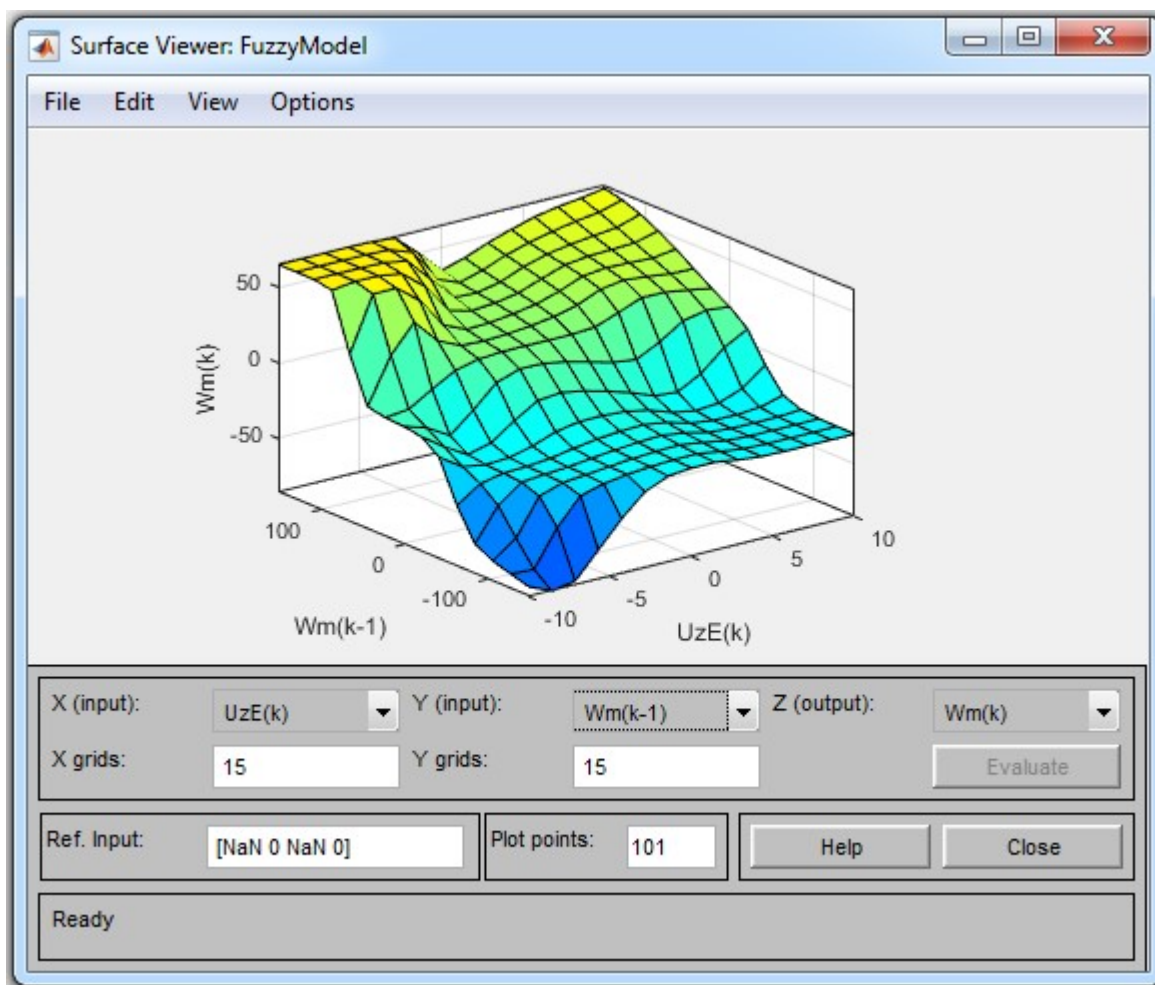


Рис. 2.9. Вікно перегляду поверхні відгуку

Крім того, задаючи у одному з полів (X (input) або Y (input)) значення none, можна проглядати одномірну залежність між однією з вхідних змінних і змінною виходу (рис. 2.10).

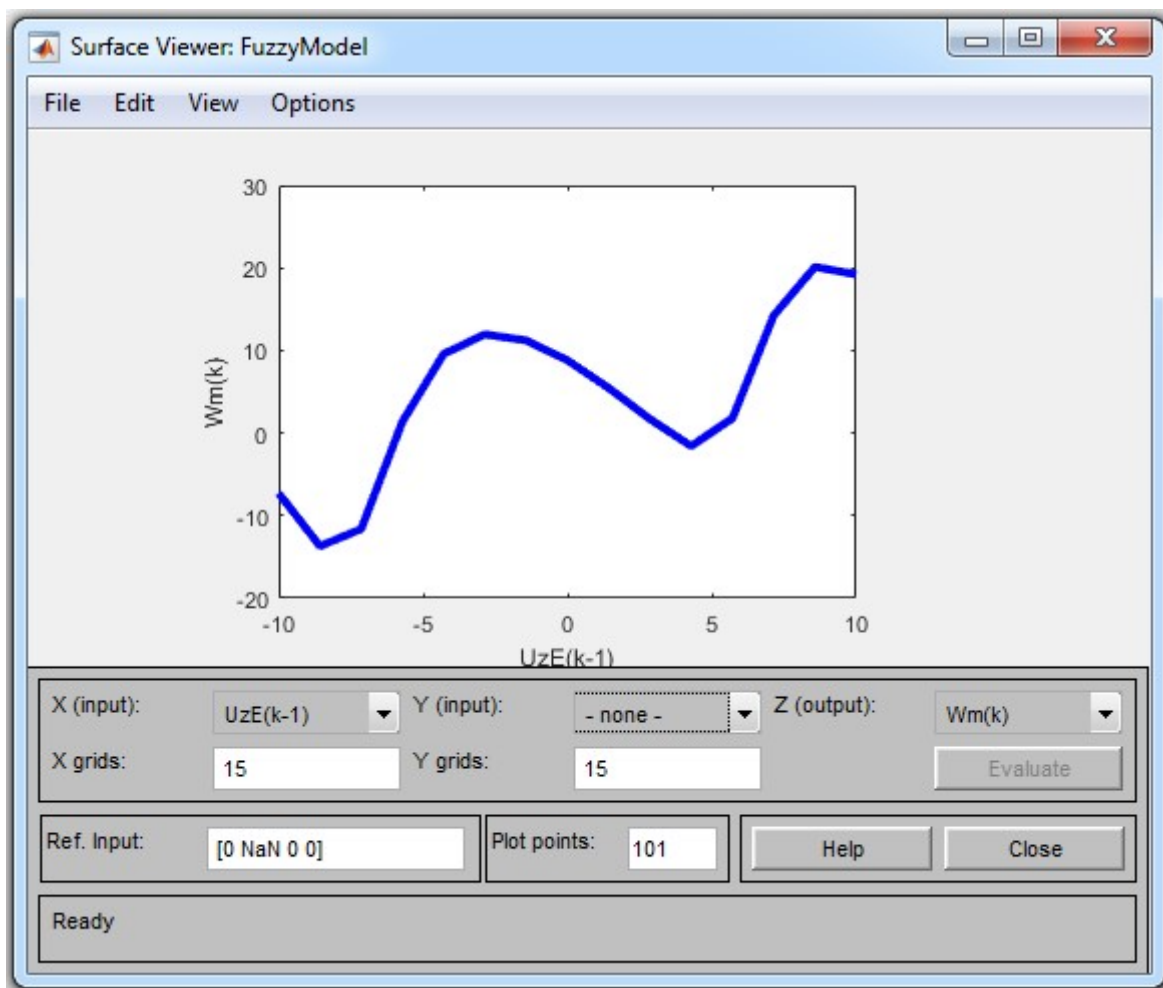


Рис. 2.10. Одновимірну залежність вихідної змінної $\omega_m(k)$ від вхідної $U_{zE}(k-1)$

Слід зазначити, що за допомогою вище приведених програм-редакторів на будь-якому етапі проектування нечіткої моделі в неї можна внести відповідні корективи, аж до завдання будь-якої функції приналежності користувача. З опцій, що встановлені в FIS-редакторі за умовчанням при використанні алгоритму Сугено, можна відзначити наступні:

- логічний висновок організується за допомогою опції добутку (**prod**);
- композиція організується за допомогою операції логічної суми (імовірного АБО, **probor**);
- приведення до чіткості організується дискретним варіантом центроїдного методу (зваженим середнім, **wtaver**).

Використовуючи відповідні поля в лівій нижній частині вікна FIS-редактору, ці опції можна за бажанням змінити.

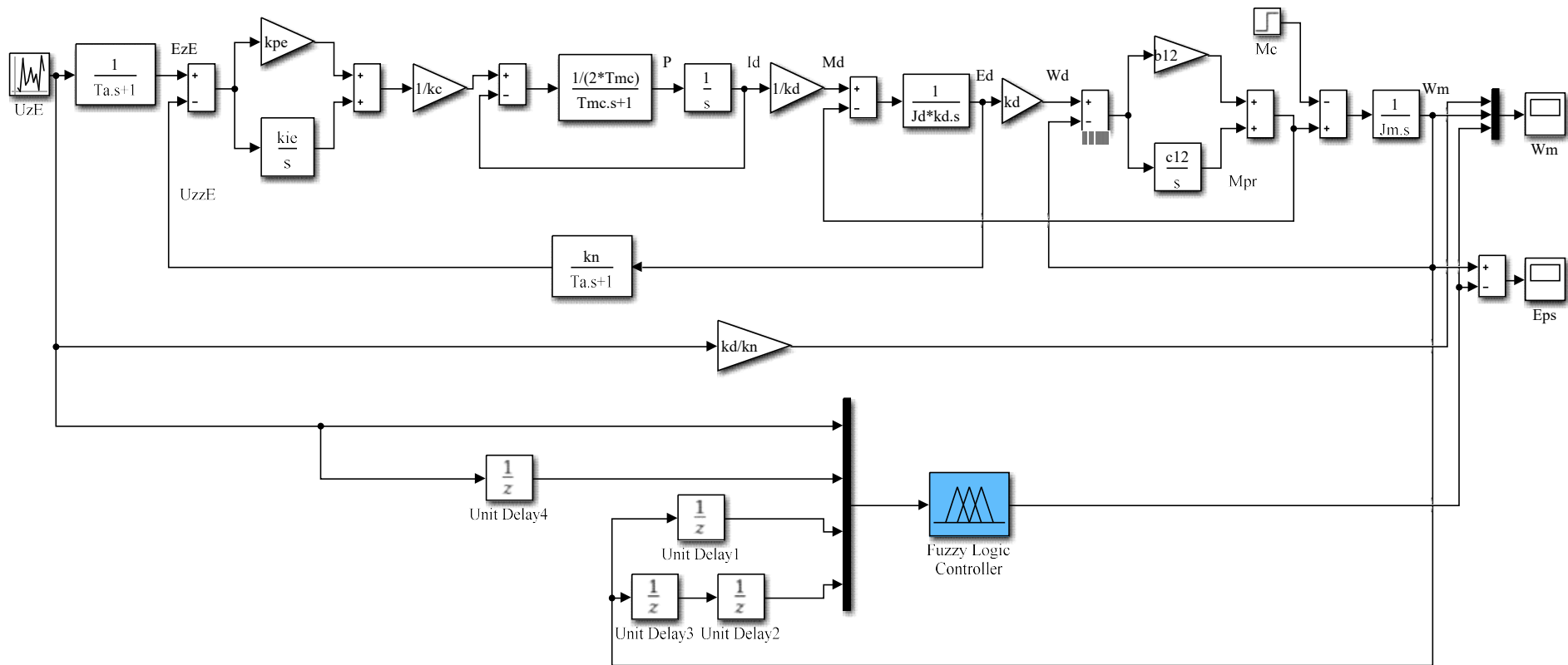


Рис. 2.11. Схема моделі двомасової системи для перевірки адекватності побудованої нечіткої моделі

2.7 Моделювання синтезованої нечіткої моделі

Виконаємо перевірку адекватності побудованої нечіткої моделі. Системи нечіткого висновку, створені за допомогою пакету **Fuzzy Logic Toolbox**, допускають інтерпретацію з інструментальними засобами Simulink, що дозволяє виконувати моделювання системи в рамках останнього. Створимо у вікні Simulink схему, показану на рис.2.11. Схема складається з моделі двомасової системи управління механізмом підйому крана і блоку **Fuzzy Logic Controller**.

У вікні завдання параметрів блоку (рис. 2.12) задамо ім'я згенерованої системи нечіткого виведення **FuzzyModel**.

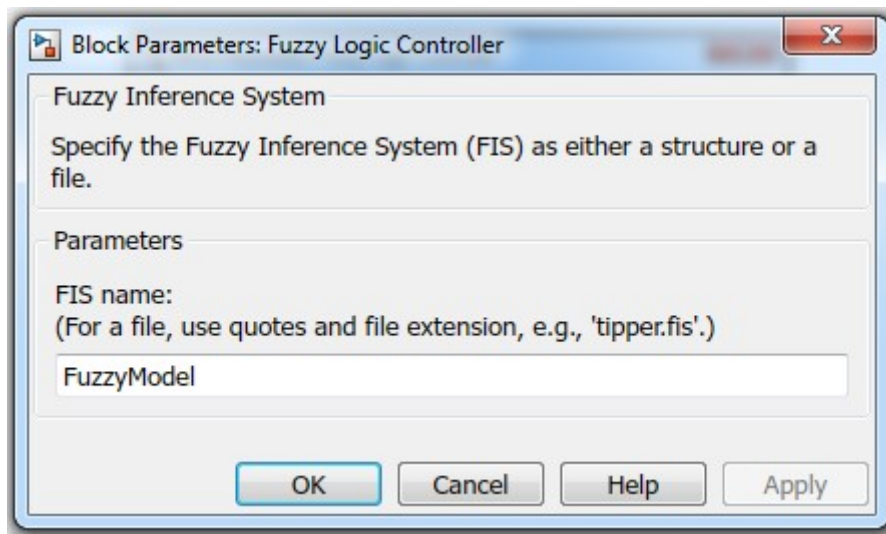


Рис. 2.12. Вікно завдання параметрів блоку Fuzzy Logic Controller

Для одержання затриманих значень вхідної і вихідної змінних використані блоки **Unit Delay**. Встановимо параметр **Sample time** [такт дискретності] вказаних блоків рівним 0,05 (можна залишити -1, що встановлено за умовчанням).

Перед початком моделювання слід завантажити в робочу область MATLAB початкові дані двомасової системи (файл **Dat_Kran**) і розроблену нечітку модель **FuzzyModel**, використовуючи команду меню **File\Export\To Workspace...** у вікні **FIS**-редактора. Результати моделювання представлені на рис. 2.13, де приведені графіки вихідної координати двомасової електромеханічної системи управління механізмом підйому мостового крана – швидкості механізму $\omega_m(t)$. Графік 1 відповідає заданому значенню швидкості, графік 2 – швидкості на виході моделі двомасової системи і графік 3 – вихідній координаті побудованої нечіткої моделі.

Як бачимо, точність ідентифікації не велика. Це пояснюється тим, що число експериментальних точок невелике і параметри функцій приналежності вхідних змінних вибрані, скоріш за все, не оптимальним чином. Досягти високої точності ідентифікації за допомогою нечіткої системи досить складно. Більш перспективним напрямком для вирішення задачі ідентифікації динамічних об'єктів і систем є вико-

ристання нечітких моделей гібридних мереж. Це питання розглядається в наступному розділі.

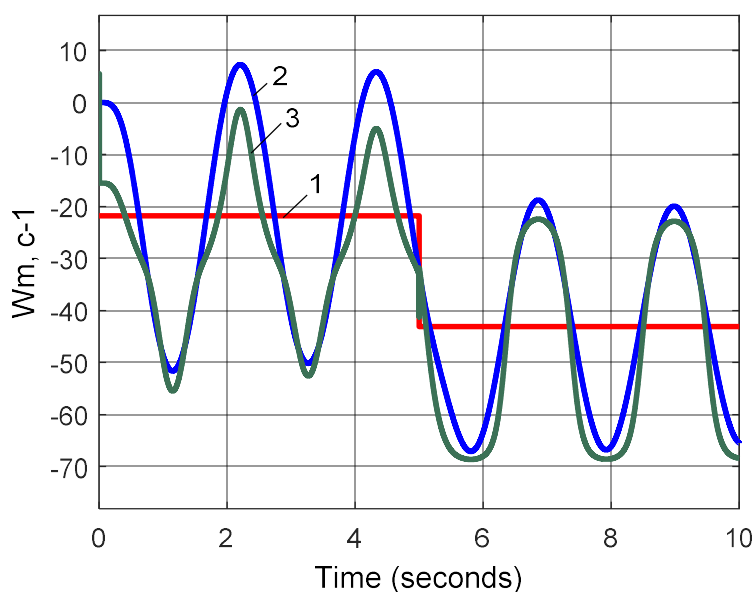


Рис. 2.13. Результати моделювання двомасової електромеханічної системи і нечіткої моделі: 1 – задане значення швидкості; 2 – швидкість на виході моделі двомасової системи; 3 – вихідна координата нечіткої моделі .

2.8 Завдання для самостійного виконання

Варіанти завдань до побудови нечітких моделей двомасових і трьохмасових електромеханічних систем з використанням GUI-інтерфейсу NNTool наведено в додатку А.

2.9 Контрольні питання по темі заняття

1. Наведіть методику синтезу нечіткої моделі двомасової електромеханічної системи.
2. Як вибирається кількість вхідних сигналів системи нечіткого висновку?
3. Як вибирається тип і кількість функцій приналежності вхідних і вихідної змінної?
4. Як виконується формування і введення правил нечіткого висновку?
5. З якою метою використовуються вікна перегляду правил і перегляду поверхні відгуку ?
6. Наведіть і поясніть схему моделі двомасової системи для перевірки адекватності побудованої нечіткої моделі.
7. Як можна використовувати системи нечіткого висновку, створені за допомогою пакету Fuzzy Logic Toolbox, для моделювання в Simulink?

Практичне заняття 3

ПОБУДОВА МОДЕЛЕЙ ДИНАМІЧНИХ ОБ'ЄКТІВ З ВИКОРИСТАННЯМ ГІБРИДНИХ МЕРЕЖ

3.1 Мета заняття

Ознайомити студентів з методами побудови моделей динамічних об'єктів на основі гібридних нейронних мереж. Набути практичних навичок у створенні, налаштуванні та тестуванні гібридних моделей у MATLAB.

3.2 Реалізація ANFIS в середовищі MATLAB

У пакеті Fuzzy Logic Toolbox системи MATLAB гібридні мережі реалізовані у формі адаптивних систем нейро-нечіткого виведення ANFIS. При цьому розробка і дослідження гібридних мереж виявляється можливою:

- у інтерактивному режимі за допомогою спеціального графічного редактора адаптивних мереж, що отримав назву редактора ANFIS;
- у режимі командного рядка за допомогою введення імен відповідних функцій з необхідними аргументами безпосередньо у вікно команд системи MATLAB. Для роботи в режимі командного рядка призначені спеціальні функції [76, 77].

Редактор ANFIS дозволяє створювати або завантажувати конкретну модель адаптивної системи нейро-нечіткого висновку, виконувати її навчання, візуалізувати її структуру, змінювати і налаштовувати її параметри, а також використовувати настроєну мережу отримання результатів нечіткого висновку.

Графічний інтерфейс редактора ANFIS викликається функцією **anfisedit** командного рядка (рис. 3.1).

Головне меню редактора ANFIS достатньо просте і призначено для роботи із заздалегідь створеною системою нечіткого висновку. Основну частину графічного інтерфейсу займає вікно візуалізації даних, яке розташоване нижче за головне меню. Для новостворюваної гібридної мережі це вікно не містить ніяких даних.

Для створення гібридної мережі необхідно завантажити дані. Для цієї мети слід скористатися кнопкою **Load Data** в лівій нижній частині графічного вікна. При цьому дані можуть бути завантажені із зовнішнього файлу (**disk**) або з робочої області (**workspace**). У першому випадку необхідно заздалегідь створити файл з початковими даними (файл з розширенням **.dat**), який є звичайним текстовим файлом. При цьому початкові дані є звичайною числовою матрицею розмірності $m \times (n + 1)$, у якій кількість рядків m відповідає об'єму вибірки N_B , перші n стовпців – значенням вхідних змінних моделі, а останній стовпець – значенню вихідної змінної. Згідно правилам системи MATLAB окремі значення матриці відділяються пропусками, а

кожен рядок матриці завершується символом «переведення каретки» (клавіша **Enter**).

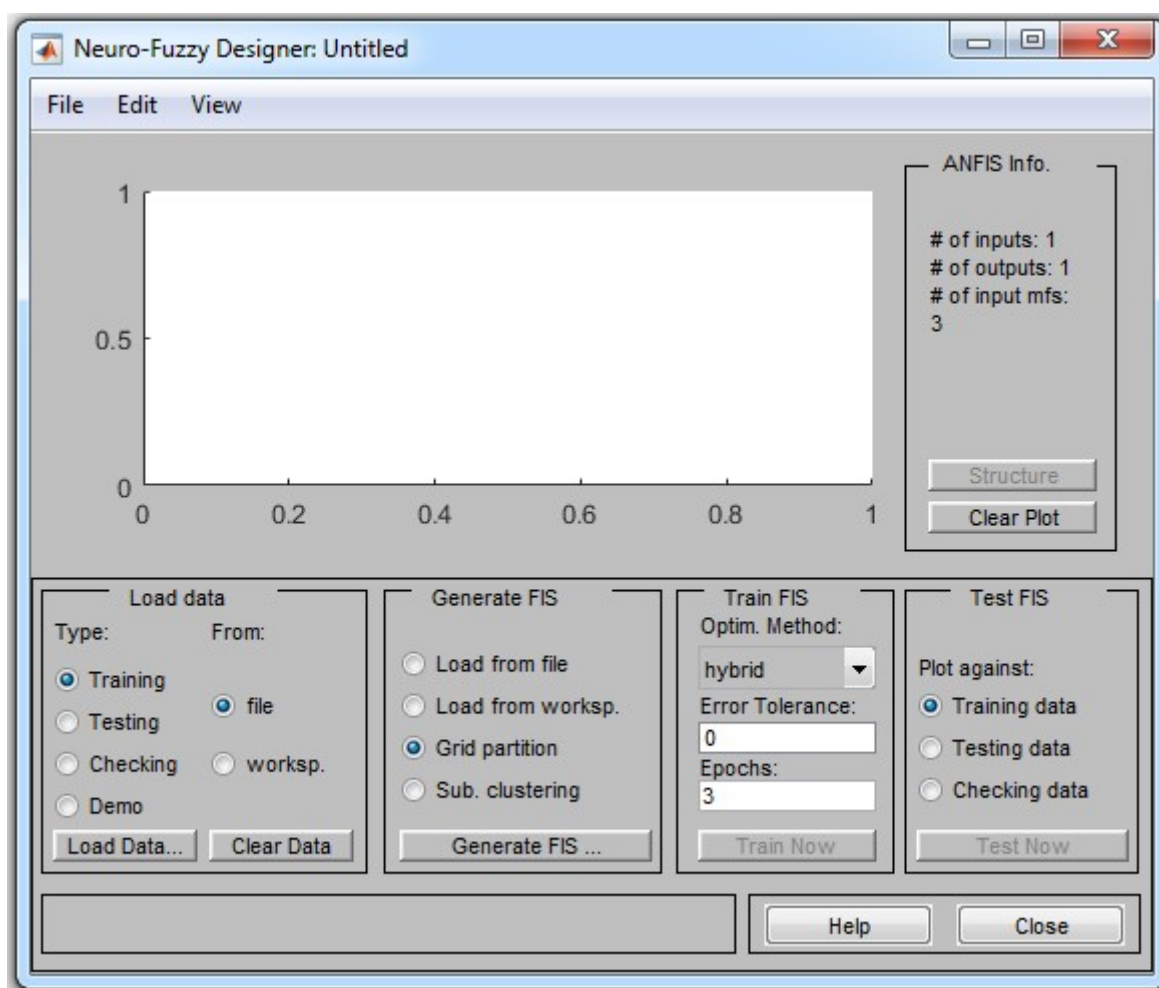


Рис.3.1. Графічний інтерфейс редактора ANFIS, що викликається функцією **anfisedit**

Примітка. Хоча по кількості рядків матриці початкових даних не існує формальних рекомендацій, прийнято вважати, що якість навчання гібридної мережі, а, отже, і точність отримуваних результатів пропорційно залежить від об'єму навчальної вибірки. Що стосується кількості стовпців матриці початкових даних, то слід зазначити можливі проблеми з працездатністю системи MATLAB, якщо кількість вхідних змінних перевищує 5 – 6.

Завантажені початкові дані можуть бути одного з наступних типів:

- навчальні дані (**Training**) – обов'язкові дані, які використовуються для побудови гібридної мережі;
- тестові дані (**Testing**) – необов'язкові дані, які використовуються для тестування побудованої гібридної мережі з метою перевірки якості функціонування побудованої гібридної мережі;
- перевірочні дані (**Checking**) – необов'язкові дані, які використовуються для перевірки побудованої гібридної мережі з метою з'ясування факту перенавчання мережі.

3.3 Вивчення пунктів меню і опції редактора

Пункти меню **File** і **View**, загалом ідентичні аналогічним пунктам редактора нечіткої системи виведення (FIS-редактора) пакету **Fuzzy Logic** системи MATLAB, за тим виключенням, що тут робота може відбуватися тільки з алгоритмом нечіткого виведення **Sugeno**. Пункт меню **Edit** містить єдиний підпункт – **Undo** (Відмінити виконану дію).

Набір опцій **Load Data** (Завантажити дані) в нижній лівій частині вікна редактора розглянутий вище і включає:

- тип (**Type**) завантажуваних даних (для навчання – **Training**, для тестування – **Testing**, для перевірки – **Checking**, демонстраційні – **Demo**);
- місце, звідки повинні завантажуватися дані (з диска – **disk** або з робочої області MATLAB – **workspace**).

До даних опцій відносяться дві кнопки, натиснення на яких приводить до необхідних дій – **Load Data...** (Завантажити дані) і **Clear Data** (очистити, тобто стерти введені дані).

Наступна група опцій (в середині нижньої частини вікна ANFIS-редактора) об'єднана під ім'ям **Generate FIS** (Створення нечіткої системи висновку). Дана група включає опції:

- завантаження структури системи з диска (**Load from disk**);
- завантаження структури системи з робочої області MATLAB (**Load from worksp.**);
- розбиття (ділення) областей визначення вхідних змінних (аргументів) на підобласті – незалежно для кожного аргументу (**Grid partition**);
- розбиття всієї області визначення аргументів (вхідних змінних) на підобласті – в комплексі для всіх аргументів (**Subtract clustering** або **Sub. clustering**), а також кнопку **Generate FIS**, натиснення якої приводить до процесу створення гібридної системи з точністю до ряду параметрів.

Наступна група опцій – **Train FIS** (навчання нечіткої системи висновку) – дозволяє визначити метод «навчання» (**Optim. Method**) системи (тобто метод настройки її параметрів) – гібридний (**hybrid**) або зворотного розповсюдження помилки (**backpropa**), встановити рівень поточної сумарної (по всіх зразках) помилки навчання (**Error Tolerance**), досягши якого процес навчання закінчується і кількість циклів навчання (**Epochs**), тобто кількість «прогонів» всіх зразків (або прикладів) навчальної вибірки; процес навчання, таким чином закінчується або досягши відміченого рівня помилки навчання, або при проведенні заданого кількості циклів.

Кнопка **Train Now** (почати навчання) дозволяє почати процес навчання, тобто процес настройки параметрів гібридної мережі.

У правому верхньому кутку вікна ANFIS-редактора видається інформація (**ANFIS Info.**) про проєктовану систему: про кількість входів, виходів, функцій при-

належності входів; натиснення кнопки **Structure** (Структура) дозволяє побачити структуру мережі. Кнопка **Clear** (очистити) дозволяє стерти всі результати.

Опції **Test FIS** в правому нижньому кутку вікна дозволяють провести перевірку і тестування створеної і навченої системи з виведенням результатів у вигляді графіків (відповідні графіки для навчальної вибірки – **Training data**, тестуючої вибірки – **Testing data** і перевіркової вибірки – **Checking data**. Кнопка **Test Now** дозволяє запустити вказані процеси.

3.4 Генерація навчальних, тестових і перевірочних даних нечіткої моделі гібридної мережі для вирішення завдання ідентифікації двомасової електромеханічної системи

Як приклад побудови адаптивної системи нейро-нечіткого висновку розглянемо процес розробки нечіткої моделі гібридної мережі для вирішення завдання ідентифікації двомасової електромеханічної системи управління механізмом підйому мостового. Цей приклад дозволяє не тільки уточнити зміст і послідовність етапів розробки, але і оцінити точність отриманої нечіткої моделі за допомогою порівняння прогнозованих модельних значень із значеннями, отриманими шляхом моделювання двомасової електромеханічної системи.

Алгоритмічна схема двомасової системи наведена на рис 2.1 у практичному занятті 2.

Загальна послідовність процесу розробки моделі гібридної мережі може бути представлена в наступному вигляді.

I. Спершу підготуємо навчальні **Training**, тестові **Testing** і перевірочні **Checking** дані. Ці дані можуть бути сформовані різними способами. Розглянемо один з них, що дещо відрізняється від способу, який використовувався при розробці нейромережевої моделі двомасової системи.

Створимо у вікні Simulink схему моделі системи (рис.3.2), а в редакторі-відладчику m-файлів сформуємо файл вихідних даних системи і збережемо його на диску під іменем “**Dat_Kran**”. За допомогою команд меню **Debug\Run** введемо початкові дані в робочу область MATLAB.

У схемі моделі рис.3.2 використаний блок **To Workspace** для запису вхідного і вихідного сигналів системи (в даному випадку напруги завдання $U_{зЕ}(t)$ і швидкості механізму $\omega_m(t)$) у робочу область системи MATLAB. У вікні завдання параметрів блоку слід задати наступні параметри.

Variable name [ім'я змінної]. Ім'я масиву, в який записуватимуться дані. Встановимо **Uz_Wm**.

Sample time [такт дискретності]. Величина такту дискретності встановлюється з тих же міркувань, що і при розробці нейромережевої моделі (див. главу 6). Для даного прикладу встановимо такт дискретності $\Delta t = 0,05c$.

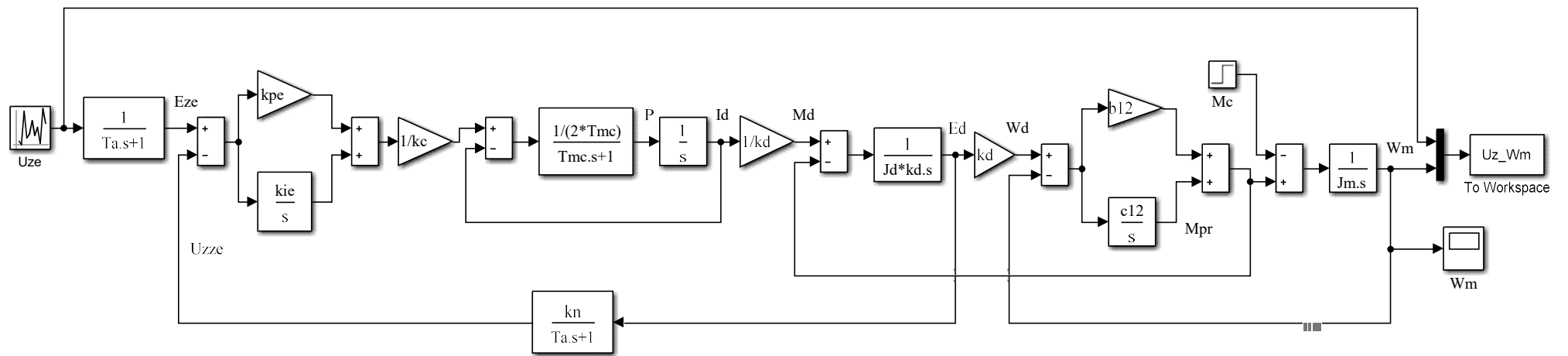


Рис. 3.2. Схема моделі двомасової системи, побудованої в Simulink

Save format [формат даних, що зберігаються]. З запропонованого списку вибираємо формат **Array** – масив. Дані зберігаються як масив, в якому число рядків визначається числом розрахункових точок в часі, а число стовпців – розмірністю вектора, що подається на вхід блоку. У даному прикладі масив міститиме два стовпці: перший стовпець відповідає вхідному сигналу системи $U_{3E}(t)$, другий – вихідному сигналу $\omega_m(t)$.

Решту параметрів блоку залишимо без зміни.

Як джерело сигналу в схемі рис. 3.2 використаний блок **Uniform Random Number** – джерело сигналу з рівномірним розподілом. У вікні завдання параметрів даного блоку слід задати такі ж параметри, що і при розробці нейромережевої моделі системи, а саме:

Minimum [мінімальний рівень сигналу], **Maximum** [максимальний рівень сигналу] – мінімальну і максимальну величину напруги завдання $U_{3E}(t)$ встановлюємо $+10B$ і $-10B$ відповідно.

Sample time [такт дискретності] – встановлюємо $5c$.

Задамо у вікні Simulink моделі системи час моделювання $100c$ і промодельюємо систему.

У вікні **Workspase** системи MATLAB виводяться імена всіх змінних, які знаходяться в робочій області, у тому числі і масиву, що містить значення вхідний і вихідний сигнал системи **Uz_Wm**. Масив має два стовпці і 2001 рядків.

Збережемо масив **Uz_Wm** у файлі з назвою **Dat_Uz_Wm** і розмістимо файл в папку **Fuzzy_Neural_Work** на диску **E**. Команда **save** дозволяє зберегти вміст робочої області в двійковому MAT-файлі, який можна потім викликати командою **load**. Проте для вирішення даного завдання ідентифікації це не прийнятно, оскільки, як було вказано вище, файл початкових даних повинен бути звичайним текстовим файлом з розширенням **.dat**.

Для збереження масиву **Uz_Wm** в ASCII-форматі в командному вікні системи MATLAB слід задати команду:

```
save('E:\FuzzyNeuralWork\Dat_Uz_Wm','Uz_Wm','-ascii')
```

або

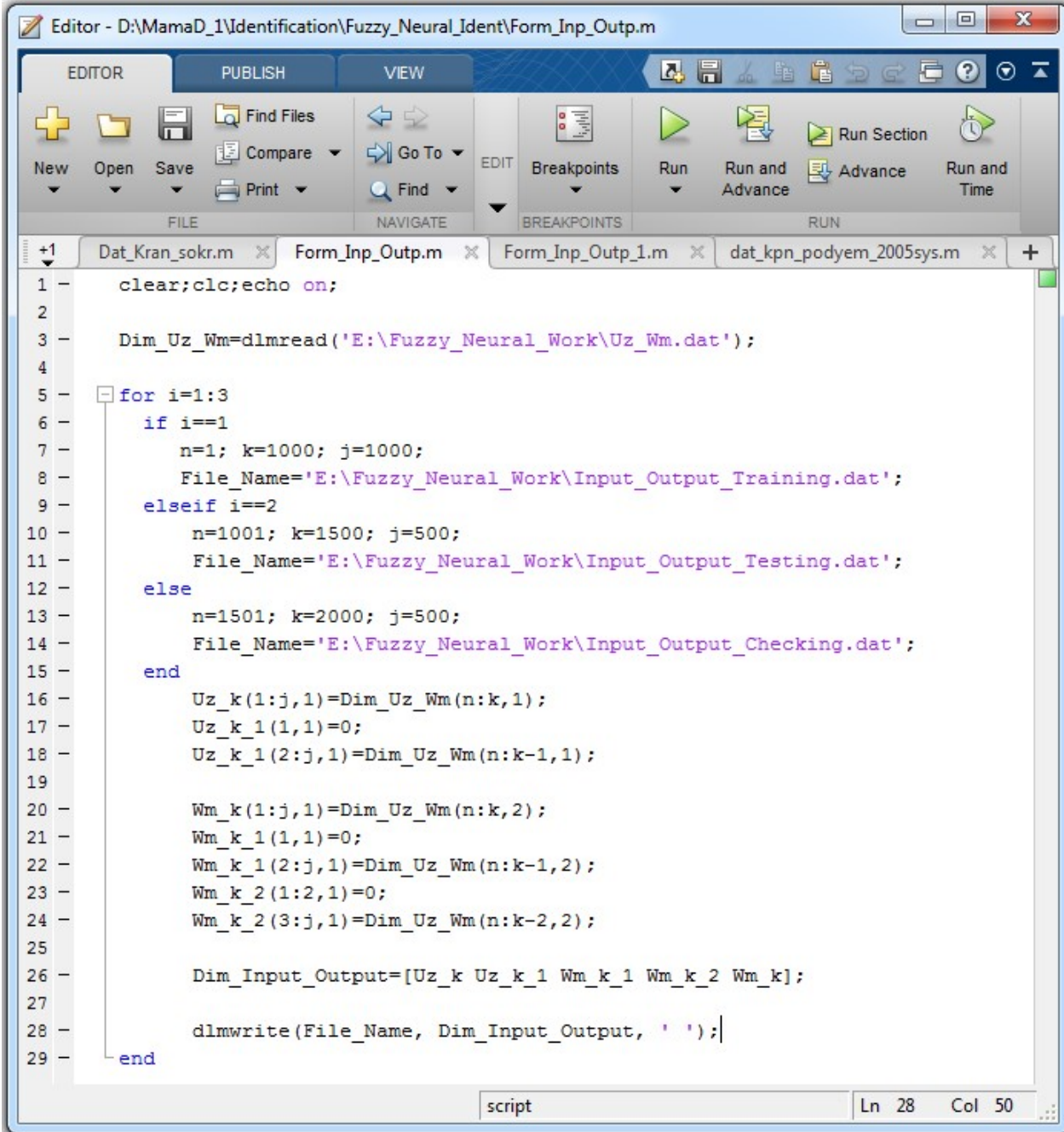
```
save E:\FuzzyNeuralWork\Dat_Uz_Wm Uz_Wm -ascii
```

Використовуємо надалі перші 1000 рядків масиву **Uz_Wm** для формування навчальних даних **Input_Output_Training**, наступні 500 рядків для формування тестових даних **Input_Output_Testing** і потім 500 рядків для формування перевірочних даних **Input_Output_Checking**.

Як було вказано вище, для побудови моделі динамічного об'єкту, необхідно вхідну послідовність задати на основі поточного значення вхідного сигналу об'єкту, і ряду попередніх значень вхідного і вихідного сигналів. Кількість затримок по вхід-

ному і вихідному сигналах заздалегідь вибираються на підставі попередніх знань про об'єкт ідентифікації, а потім корегуються в процесі побудови моделі об'єкту.

У даному прикладі сформуємо вхідну послідовність на основі поточного значення вхідного сигналу системи $U_{zE}(k)$ і вхідного сигналу, затриманого на один крок дискретності $U_{zE}(k-1)$, а також два затриманих на один і два кроки вихідних сигналів, тобто $\omega_M(k-1)$ і $\omega_M(k-2)$ відповідно.



```
1 clear;clc;echo on;
2
3 Dim_Uz_Wm=dlmread('E:\Fuzzy_Neural_Work\Uz_Wm.dat');
4
5 for i=1:3
6     if i==1
7         n=1; k=1000; j=1000;
8         File_Name='E:\Fuzzy_Neural_Work\Input_Output_Training.dat';
9     elseif i==2
10        n=1001; k=1500; j=500;
11        File_Name='E:\Fuzzy_Neural_Work\Input_Output_Testing.dat';
12    else
13        n=1501; k=2000; j=500;
14        File_Name='E:\Fuzzy_Neural_Work\Input_Output_Checking.dat';
15    end
16    Uz_k(1:j,1)=Dim_Uz_Wm(n:k,1);
17    Uz_k_1(1,1)=0;
18    Uz_k_1(2:j,1)=Dim_Uz_Wm(n:k-1,1);
19
20    Wm_k(1:j,1)=Dim_Uz_Wm(n:k,2);
21    Wm_k_1(1,1)=0;
22    Wm_k_1(2:j,1)=Dim_Uz_Wm(n:k-1,2);
23    Wm_k_2(1:2,1)=0;
24    Wm_k_2(3:j,1)=Dim_Uz_Wm(n:k-2,2);
25
26    Dim_Input_Output=[Uz_k Uz_k_1 Wm_k_1 Wm_k_2 Wm_k];
27
28    dlmwrite(File_Name, Dim_Input_Output, ' ');
29 end
```

Рис. 3.3. Текст m-файлу формування масивів навчальних, тестових і перевірочних даних

Для формування масивів вказаних даних може бути використаний будь-який текстовий редактор, наприклад редактор-відладчик m-файлів системи MATLAB (рис. 3.3). У вікні відладчика приведений текст m-файлу формування масивів навча-

льних, тестових і перевірочних даних. **Form_Inp_Outp.m**. Вказані масиви збережені у файлах з назвами **Input_Output_Training.dat**, **Input_Output_Testing.dat** і **Input_Output_Checking.dat** відповідно.

Дані масиви містять 5 стовпців. Перші 4 стовпці відповідають значенням вхідних сигналів моделі ($U_{3E}(k)$, $U_{3E}(k-1)$, $\omega_M(k-1)$, $\omega_M(k-2)$), а 5 стовпець – значенням вихідного сигналу $\omega_M(k)$. У вікні, зображеному на рис.3.4, приведені перші 21 рядки масиву **Input_Output_Training**.

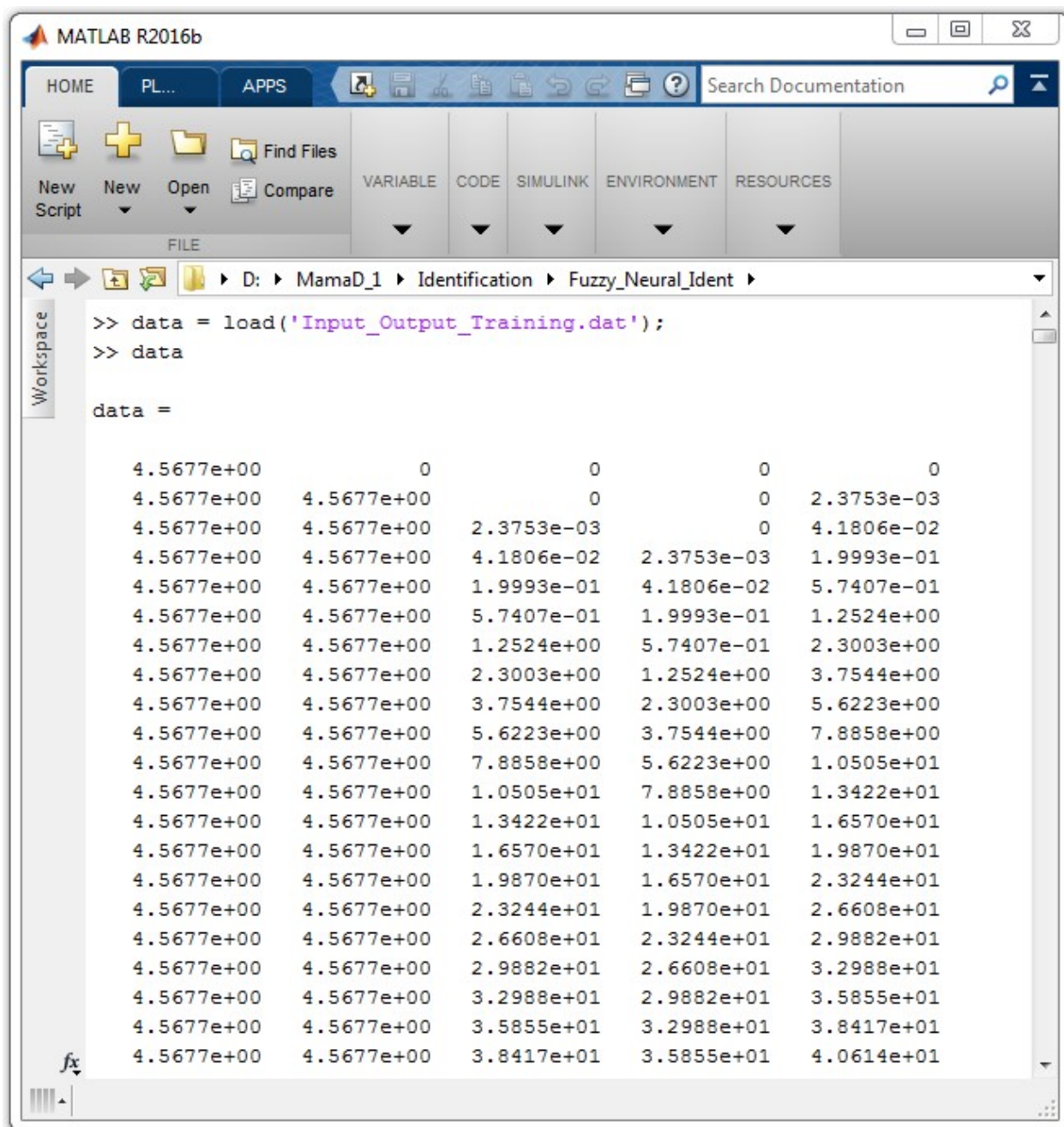


Рис. 3.4. Навчальні дані **Input_Output_Training**

Отже, в папці **E:\FuzzyNeuroWork** є три файли **Input_Output_Training.dat**, **Input_Output_Testing.dat** і **Input_Output_Checking.dat**, які можуть бути використані для розробки гібридної мережі за допомогою редактора **ANFIS**.

Завантажимо навчальні дані з файлу **Input_Output_Training.dat**, скориставшись кнопкою **Load Data** графічного вікна редактора ANFIS, встановивши тип завантажуваних даних **Training**. Після завантаження навчальних даних їх структура буде відображена в робочому вікні редактора (рис. 3.5). При цьому кожному рядку даних відповідає окрема точка графіка, яка для навчальних даних зображується кружком. На горизонтальній осі указуються порядковий номер (індекс) окремого рядка даних, а вертикальна вісь служить для вказівки значень вихідної змінної. У даному прикладі використовується 1000 точок навчальних даних.

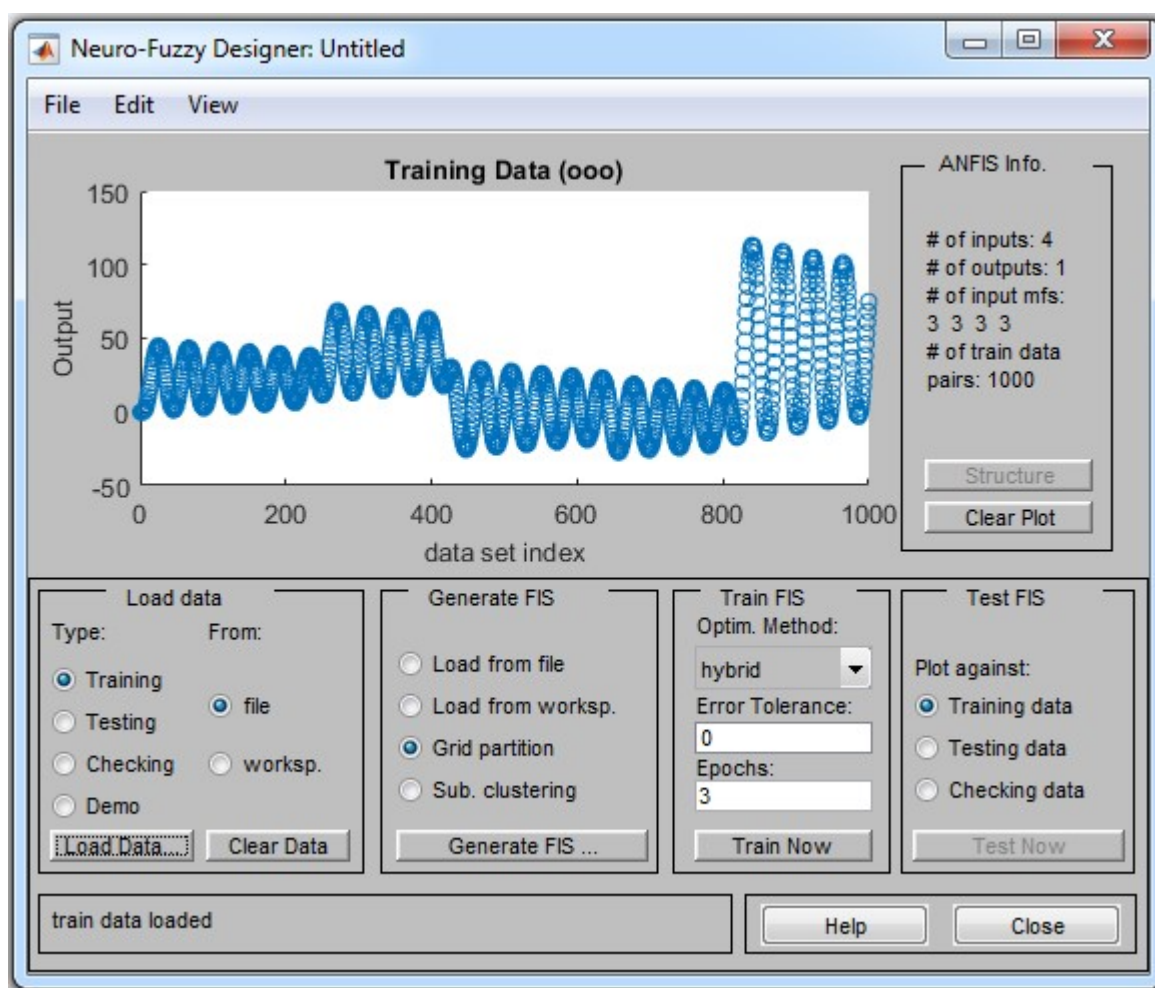


Рис. 3.5. Графічний інтерфейс редактора ANFIS після завантаження файлу з навчальними даними **Input_Output_Training.dat**

Завантажимо тестові дані з файлу **Input_Output_Testing.dat**. Для цього змінимо тип завантажуваних даних на **Testing** і виберемо ім'я вказаного файлу. В результаті тестові дані будуть завантажені в редактор ANFIS і зображені в робочому вікні (рис.3.6). В цьому випадку початковий графік буде доповнений 500 точками тестових даних, кожному рядку яких відповідає окрема точка графіка, зображена плюсом.

Аналогічно завантажимо перевірочні дані з файлу **Input_Output_Checking.dat**, встановивши тип завантажуваних даних **Checking**. В

результаті дані будуть завантажені в редактор ANFIS і зображені в робочому вікні (рис. 3.7). Початковий графік доповнюється 500 точкою перевірочних даних, кожному ряду яких також відповідає окрема точка графіка, зображена жирною точкою.

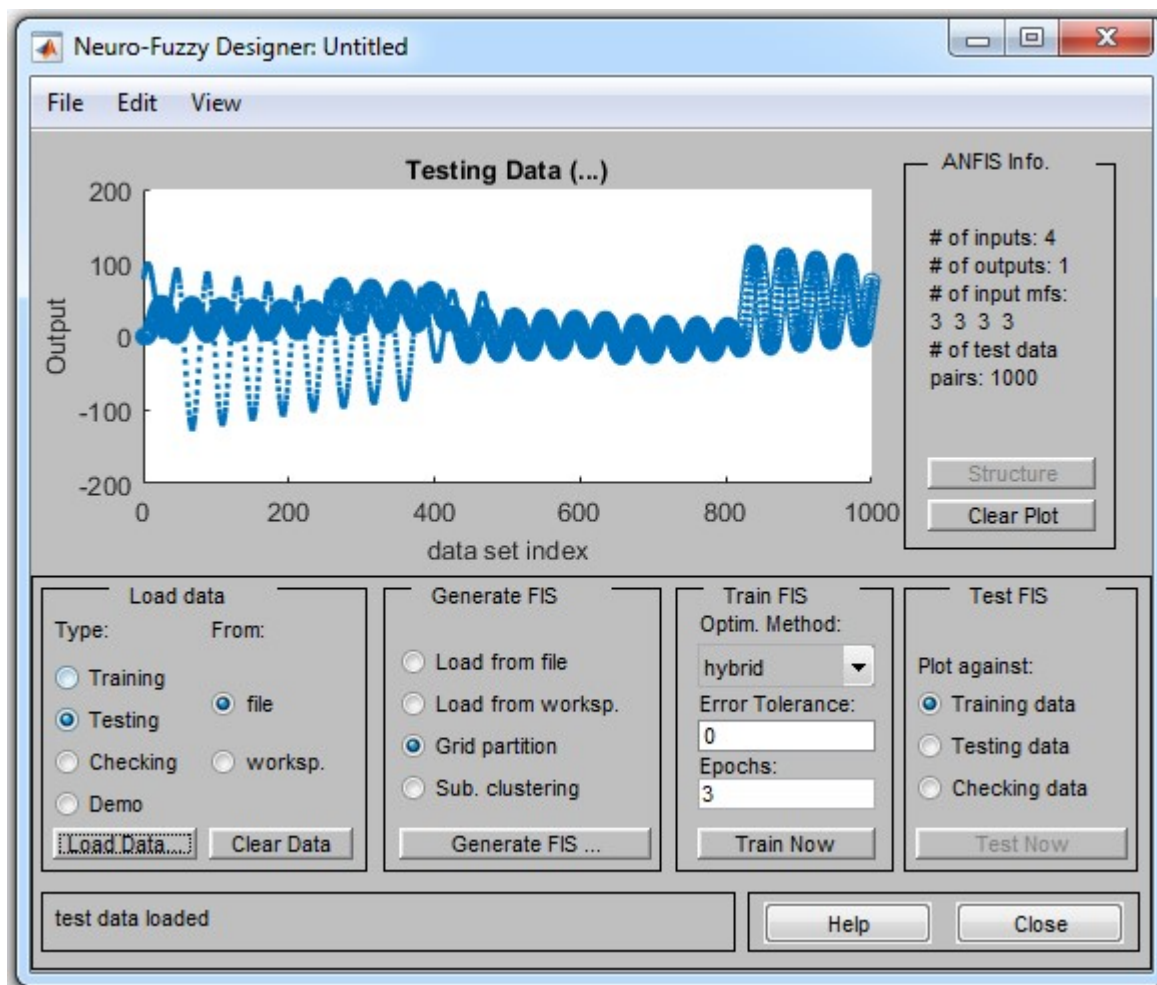


Рис.3.6 .Графічний інтерфейс редактора ANFIS після завантаження файлу з тестовими даними **Input_Output_Testing.dat**

3.5 Генерація системи нечіткого виведення

Після підготовки і завантаження даних можна згенерувати структуру системи нечіткого виведення **FIS** типу **Сугено**, яка є моделлю гібридної мережі в системі MATLAB. Для цієї мети слід скористатися кнопкою **Generate FIS** в нижній частині робочого вікна редактора. При цьому дві перші опції відносяться до заздалегідь створеної структури гібридної мережі, а дві останні – до форми розбиття вхідних змінних моделі.

Завантажити структуру вже створеної **FIS** можна або з диска (**Load from disk**), або з робочої області (**Load from worksp.**). При створенні структури нової **FIS** можна незалежно розбити всі вхідні змінні на області їх значень (**Grid partition**) або скористатися процедурою субтрактивної кластеризації для попереднього розбиття значень вхідних змінних на кластери близьких значень (**Sub. clustering**).

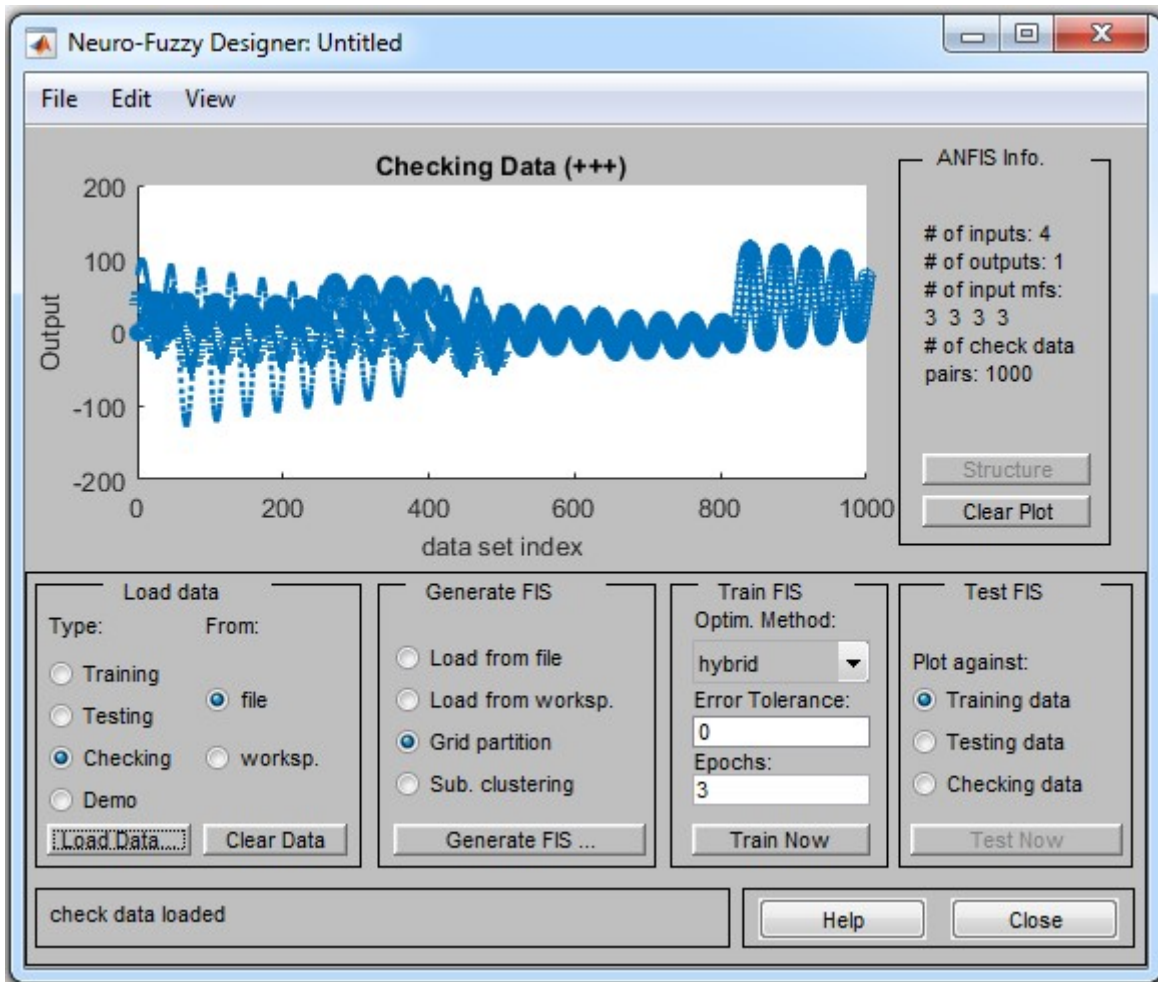


Рис. 3.7. Графічний інтерфейс редактора ANFIS після завантаження файлу з перевірочними даними `Input_Output_Checking.dat`

Натисненням кнопки **Generate FIS** викликається діалогове вікно з вказівкою числа і типу функцій приналежності для окремих термів вхідних змінних і вихідної змінної (рис. 3.8). В цьому випадку можна вибрати будь-який тип функцій приналежності з реалізованих в системі MATLAB.

Після генерації структури гібридної мережі можна візуалізувати її структуру, для чого слід натиснути кнопку **Structure** в правій частині графічного вікна. Структура отриманої в результаті системи нечіткого виведення **FIS** відображається в окремому вікні (рис. 3.9).

Для даного прикладу найбільша точність ідентифікації отримана для випадку, коли система нечіткого висновку містить 4 вхідних змінних з 4 термами, 4 правила нечітких висновків, одну вихідну змінну з 4 термами. На рис.3.9 для наочності показана структура системи нечіткого висновку з 2 термами. Як тип їх функцій приналежності вибрані трикутні функції (встановлені системою MATLAB за умовчанням). Як функція приналежності вихідної змінної задана лінійна функція. Компоненти системи **FIS** зображуються вузлами відповідного кольору.

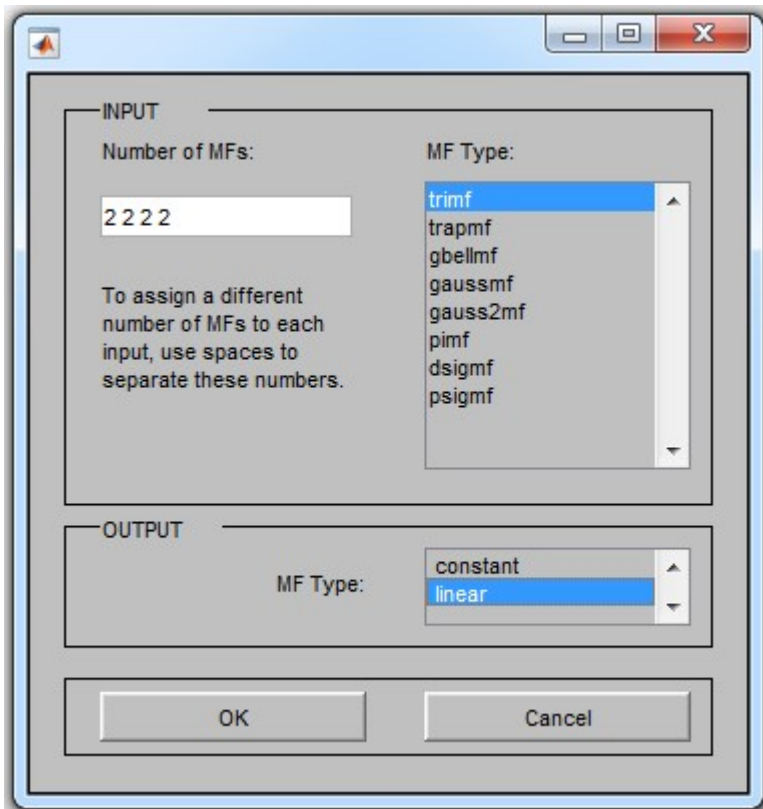


Рис. 3.8 .Діалогове вікно для завдання кількості і типу функцій приналежності

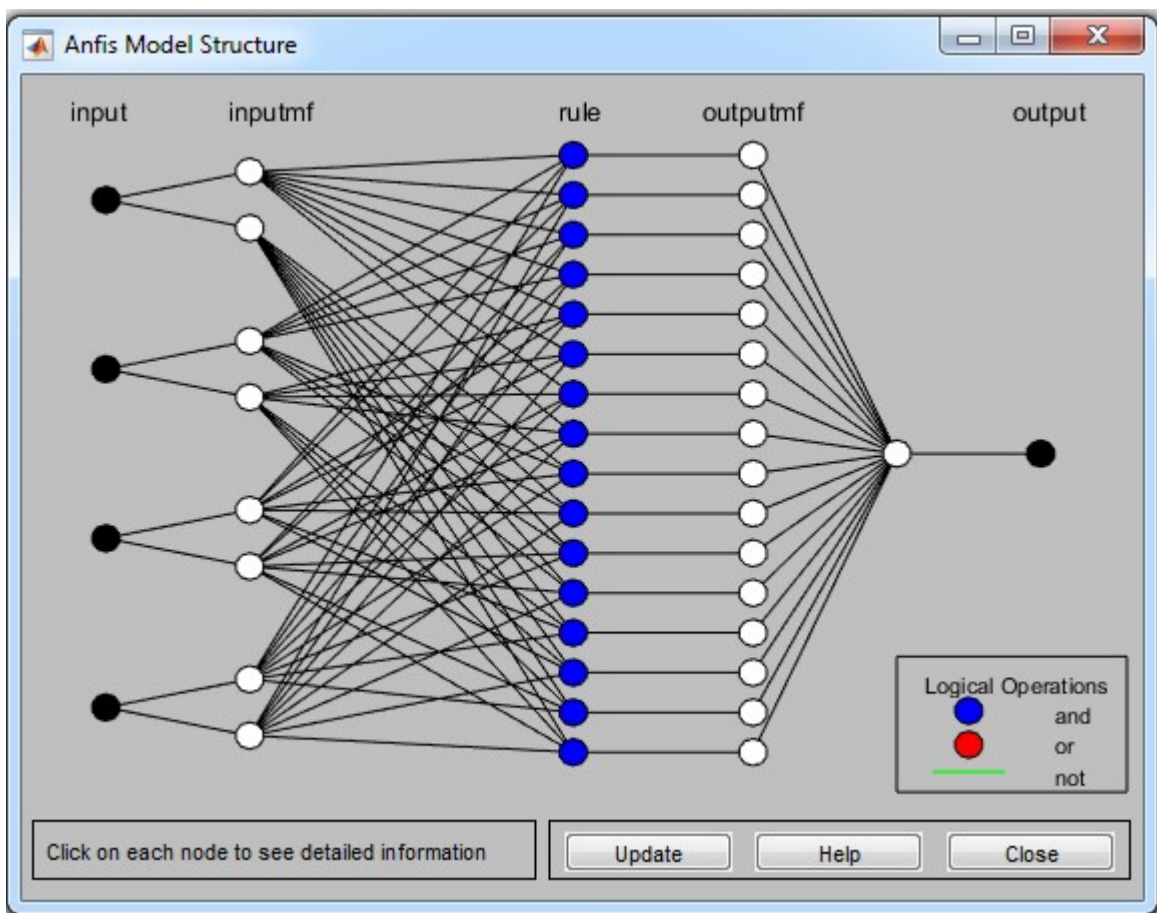


Рис. 3.9. Структура згенерованої системи нечіткого висновку

3.6 Навчання гібридної мережі

Перед навчанням гібридної мережі необхідно задати параметри навчання, для чого слід скористатися наступною групою опцій в правій нижній частині робочого вікна:

Вибрати метод навчання гібридної мережі – зворотного розповсюдження (**backprop**) або гібридний (**hybrid**), що є комбінацією методу найменших квадратів і методу убунання зворотного градієнта.

Встановити рівень помилки навчання (**Error Tolerance**) – за умовчанням значення 0 (змінювати не рекомендується).

Задати кількість циклів навчання (**Epochs**) – за умовчанням значення 3 (рекомендується збільшити і для даного прикладу задамо його значення, рівним 300).

Для навчання мережі слід натиснути кнопку **Train Now**. При цьому хід процесу навчання ілюструється у вікні візуалізації у формі графіка залежності помилки від кількості циклів навчання (рис. 3.10).

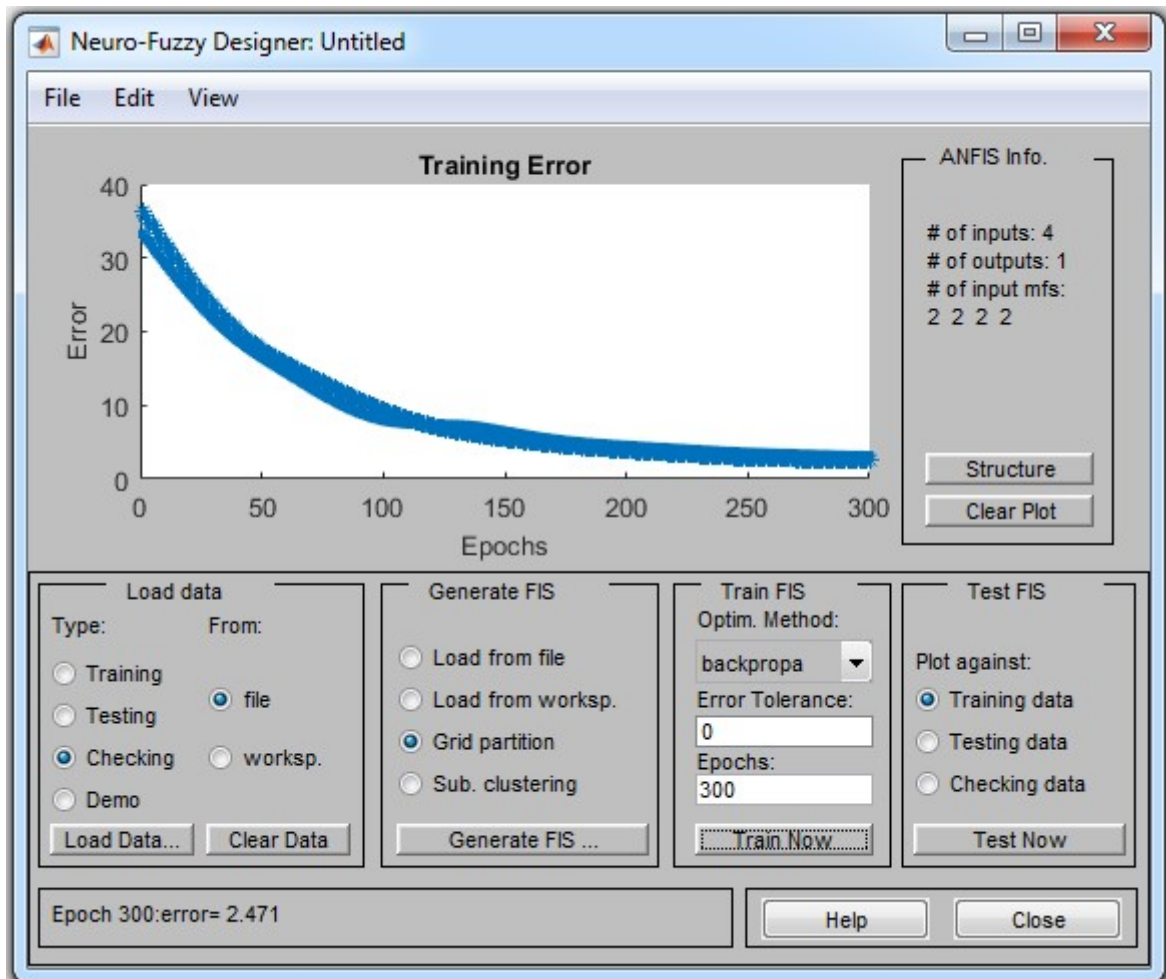


Рис. 3.10. Графік залежності помилок навчання і перевірки від кількості циклів навчання

В цьому випадку на графіці зображеному знаком «*», приведена залежність помилки навчання від кількості циклів навчання, а на графіку, зображеному «•», показана залежність помилки перевірки від кількості циклів навчання.

За допомогою опції **Test FIS** в правому нижньому кутку вікна виконаємо перевірку і тестування створеної і навченої системи з виведенням результатів у вигляді графіків (відповідні графіки для навчальної вибірки – **Training data**, тестової вибірки, – **Testing data** і перевіркової вибірки – **Checking data** приведені на рис. 3.11-3.13.

Як видно з графіків, значення сигналів на виході двомасової системи електро-механічної системи управління механізмом підйому мостового крана і на виході нечіткої моделі гібридної мережі практично повністю співпадають.

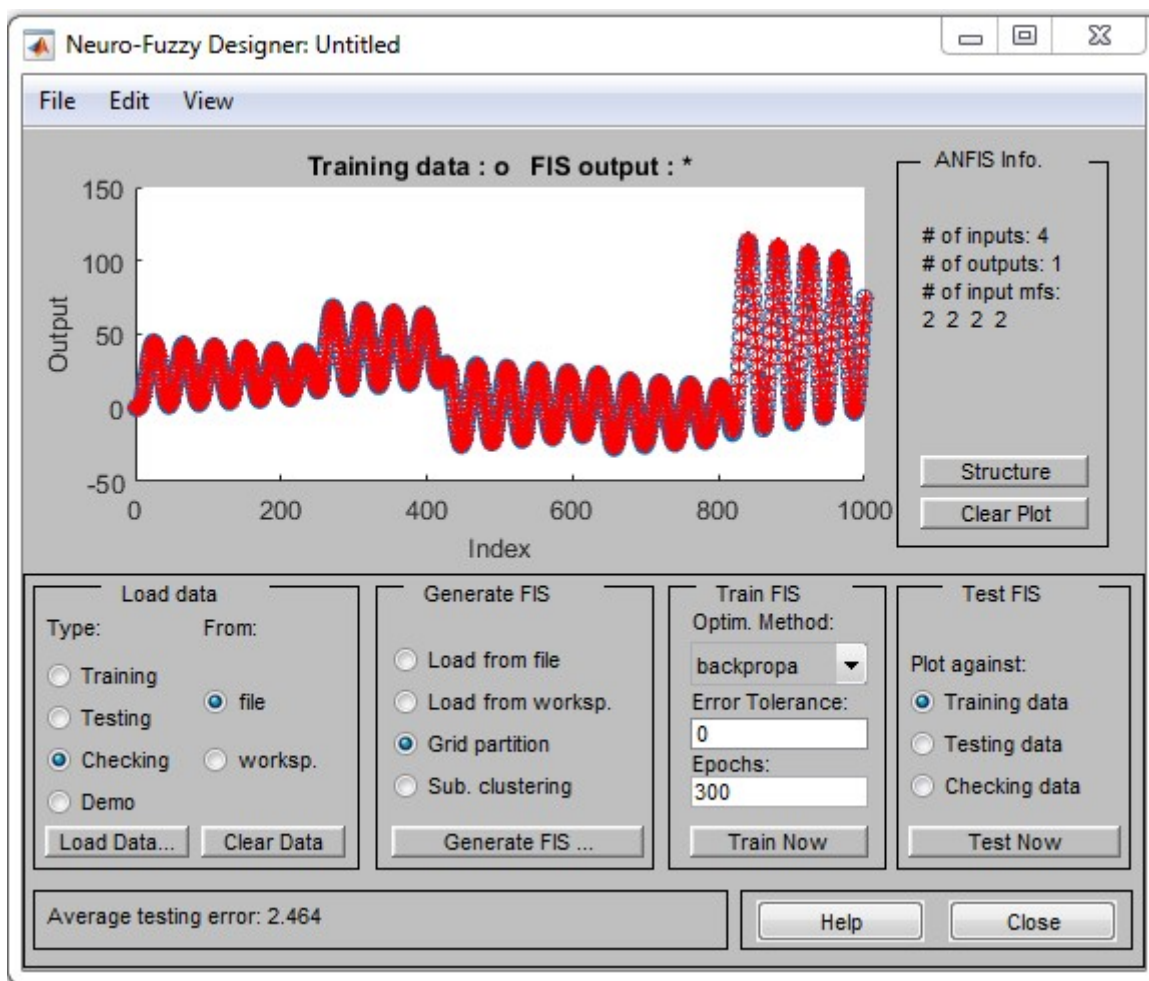


Рис. 3.11. Графіки перевірки створеної і навченої системи для навчальної вибірки – **Training data**

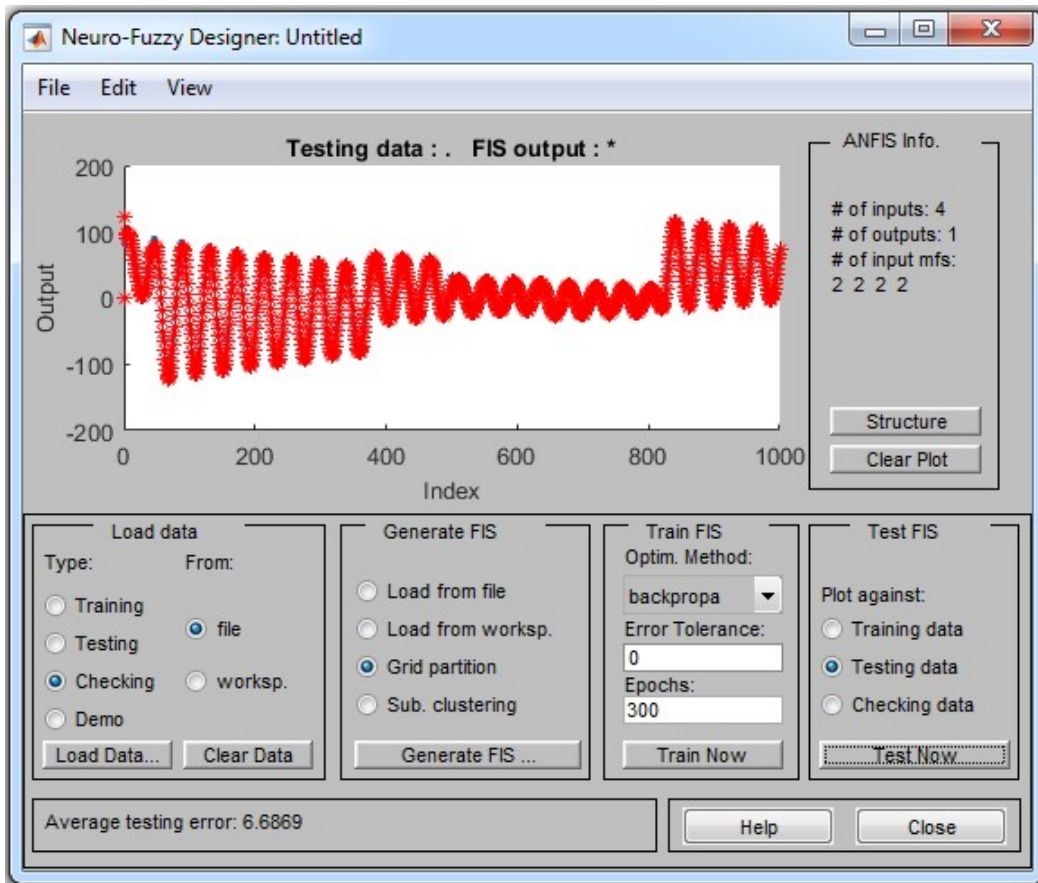


Рис. 3.12 . Графіки перевірки створеної і навченої системи для тестової вибірки – **Testing data**.

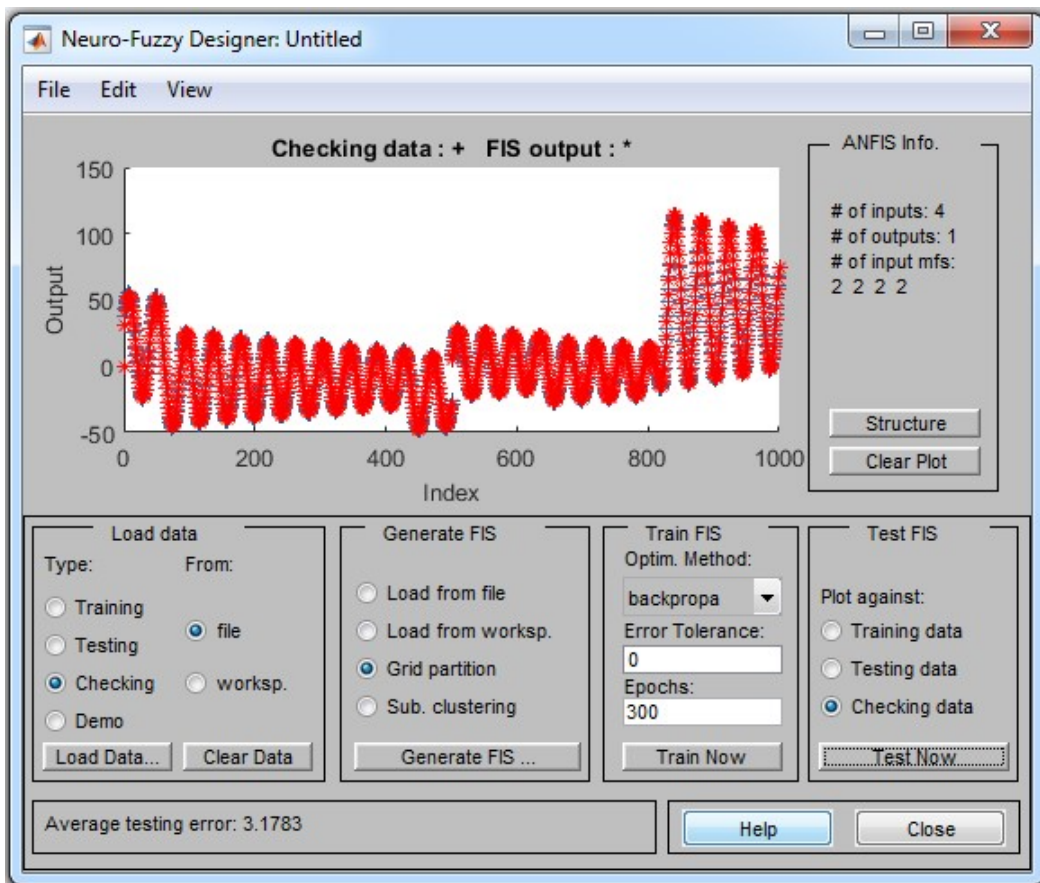


Рис. 3.13.Графіки перевірки створеної і навченої системи для перевірконої вибірки – **Checking data**.

При необхідності подальша настройка параметрів побудованої і навченої гібридної мережі може бути виконана за допомогою стандартних графічних засобів пакету Fuzzy Logic Toolbox. Для цього слід зберегти створену систему нечіткого висновку в зовнішньому файлі з розширенням **fis** (збережемо у файлі з ім'ям **ANFIS_model.fis**), після чого необхідно завантажити цей файл редактор систем нечіткого виведення **FIS** (рис. 3.14).

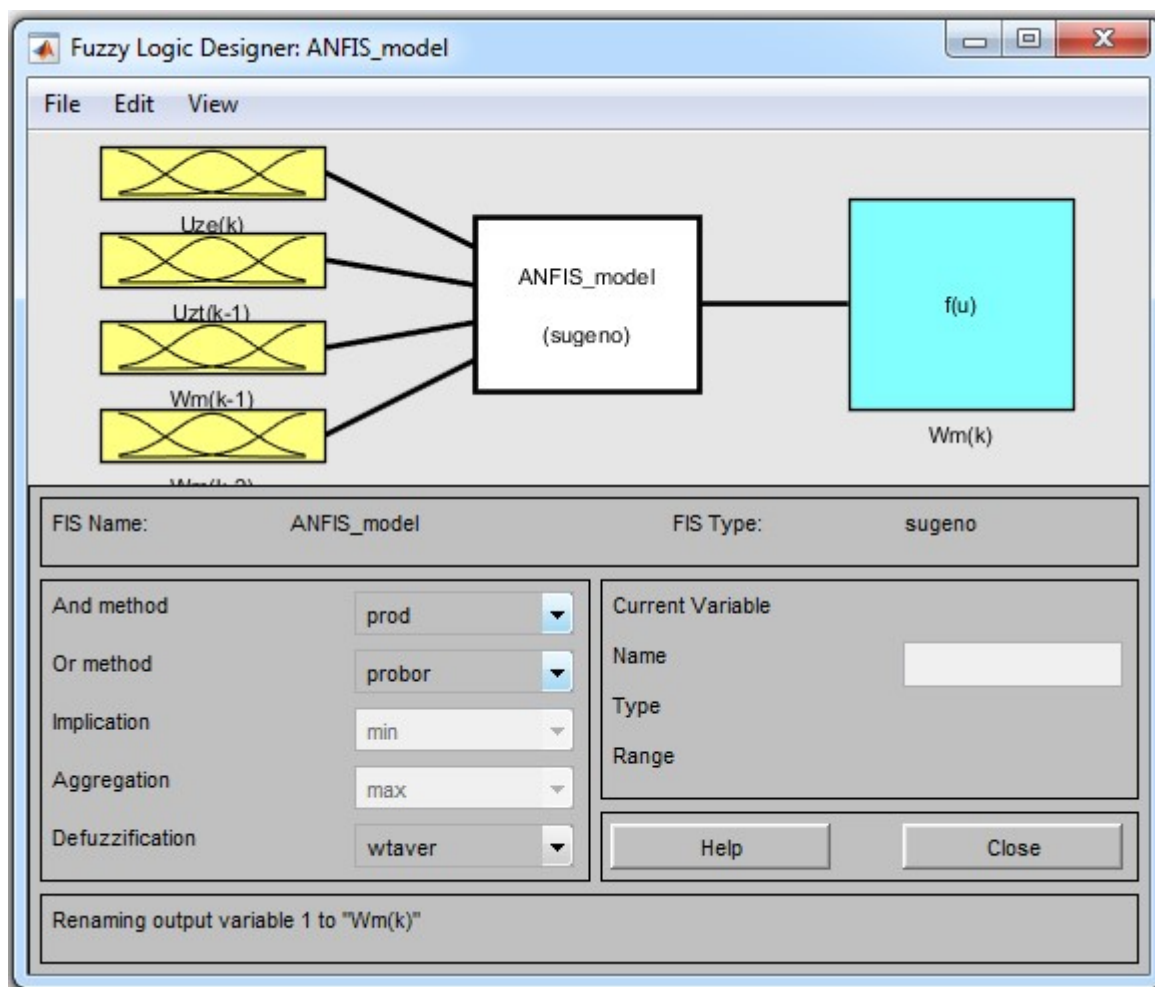


Рис. 3.14. Графічний інтерфейс редактора FIS для згенерованої системи нечіткого висновку

При цьому також стають доступними редактор функцій приналежності системи нечіткого висновку (**Membership Function Editor**), редактор правил системи нечіткого висновку (**Rule Editor**), програма перегляду правил системи нечіткого висновку (**Rule Viewer**) і програма перегляду поверхні системи нечіткого висновку (**Surface Viewer**).

Примітка. У останніх версіях системи MATLAB виклик графічних засобів редагування і перегляду моделей FIS можливий безпосередньо з головного меню редактора ANFIS.

3.7 Дослідження побудованої моделі гібридної мережі

Для дослідження побудованої моделі гібридної мережі можна скористатися програмою перегляду правил (**Rule Viewer**). Для одержання буд-якого значення вихідної змінної, необхідно задати конкретні значення вхідних змінних аналогічно загальним рекомендаціям систем нечіткого висновку. При цьому на графіці функцій приналежності вихідної змінної буде вказане шукане значення вихідної змінної (рис. 3.15).

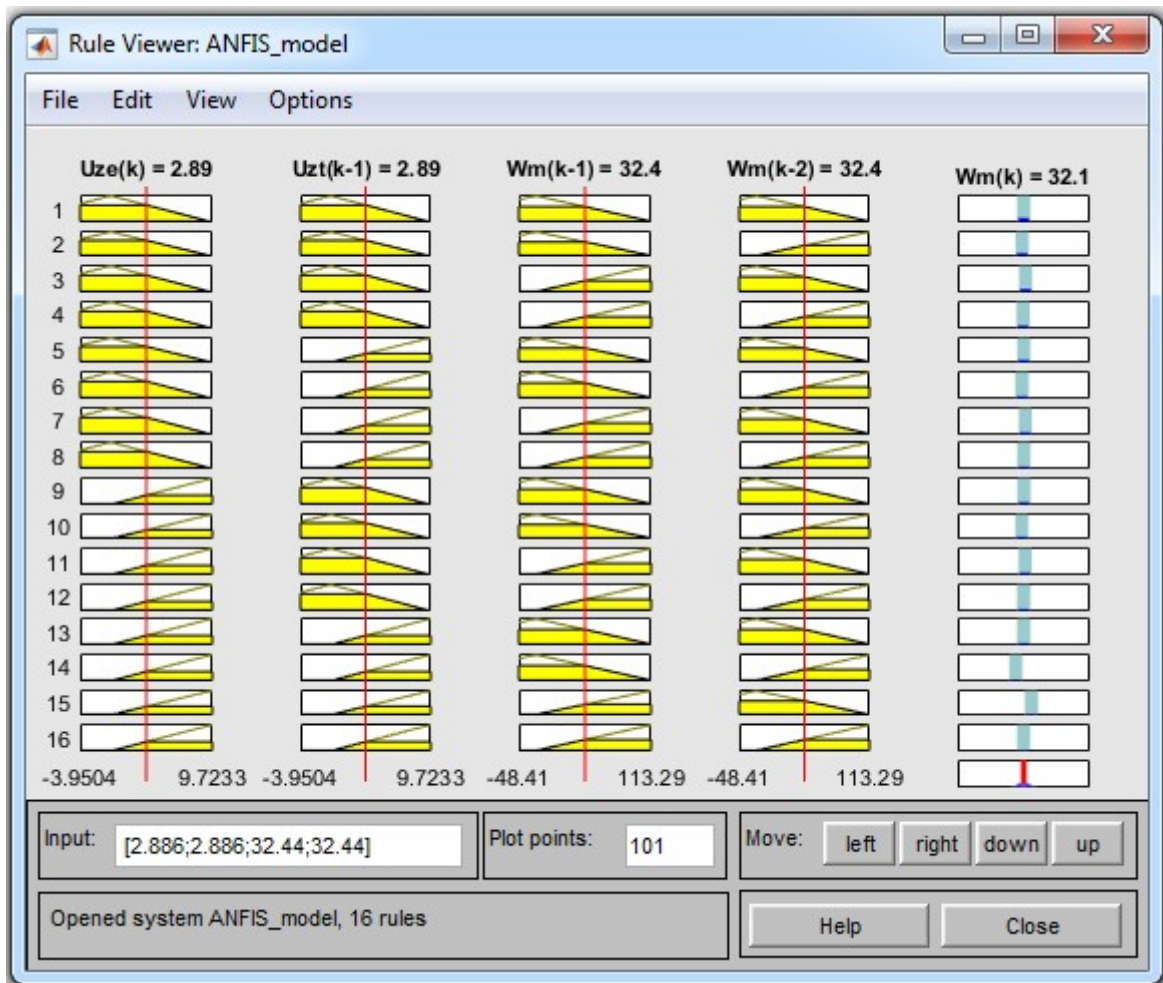


Рис. 3.15. Графічний інтерфейс перегляду правил згенерованої системи нечіткого висновку

На додаток до цього можна виконати візуальний аналіз поверхні висновку для побудованої гібридної мережі, яка також дозволяє оцінити значення вихідної змінної. Виконати аналіз навченої і настроєної гібридної мережі можна за допомогою візуалізації поверхні нечіткого висновку. Для цієї мети слід скористатися програмою перегляду поверхні відгуку системи нечіткого висновку (рис. 3.16).

Зображена поверхня висновку може бути інтерпретована як графік функціональної залежності вихідної змінної від будь-яких двох вхідних змінних.

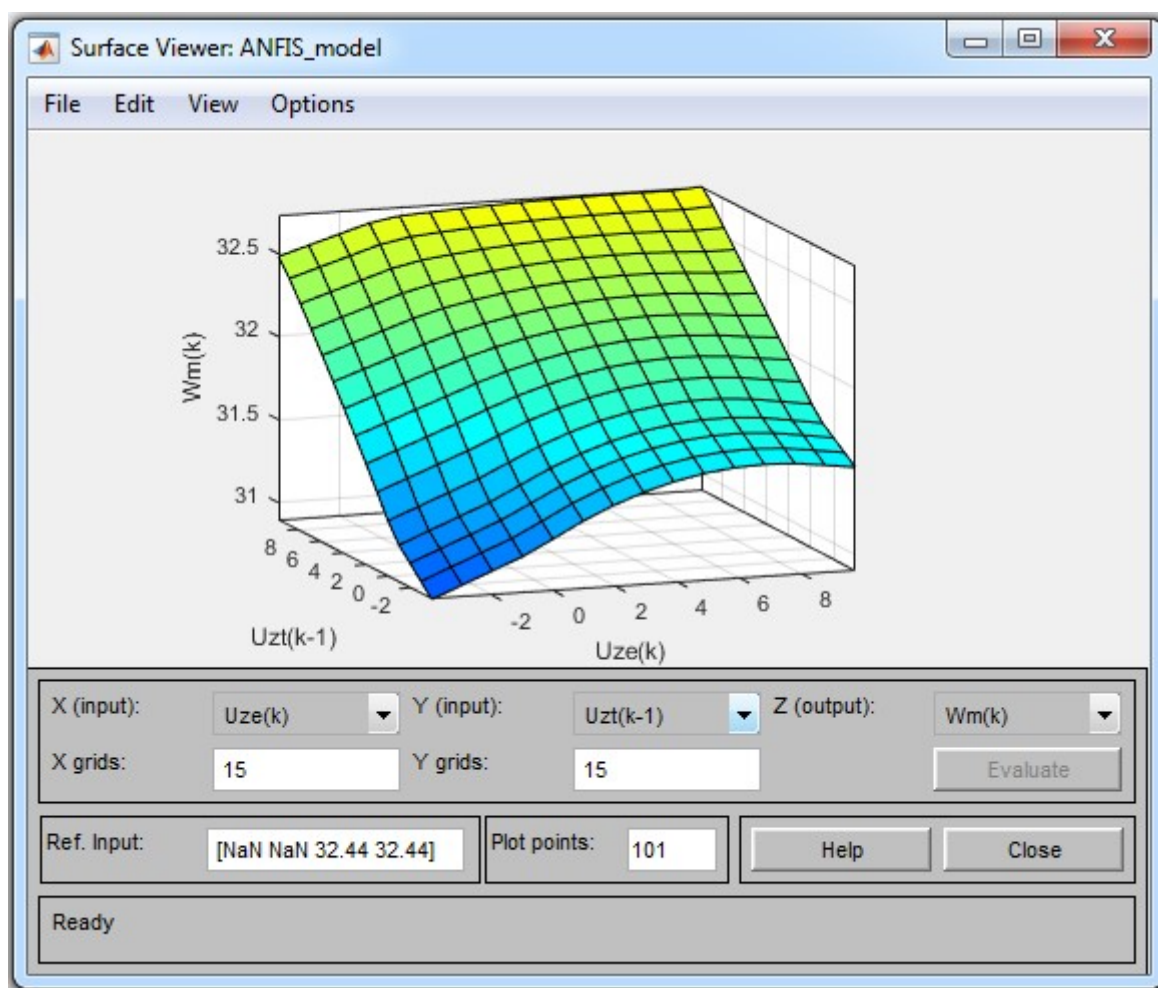


Рис. 3.16. Графічний інтерфейс перегляду поверхні висновку згенерованої системи нечіткого висновку

3.8 Моделювання нечіткої моделі гібридної мережі

Виконаємо перевірку адекватності побудованої нечіткої моделі гібридної мережі. Для цієї мети використаємо схему, наведену на рис.3.11 в темі 3.

У вікні завдання параметрів блоку **Fuzzy Logic Controller** у полі **FIS File or Structure** задамо ім'я згенерованої системи нечіткого виводу **ANFIS_model**.

Перед початком моделювання слід завантажити в робочу область MATLAB початкові дані двомасової системи (файл **Dat_Kran**) і розроблену нечітку модель гібридної мережі **ANFIS_model**, використовуючи команду меню **File\Export\To Workspace** у вікні редактора **ANFIS** (див. рис. 3.1). Результати моделювання представлені на рис. 3.17 і рис. 3.18.

На рис. 3.17 приведені графіки вихідної координати двомасової електромеханічної системи управління механізмом підйому мостового крана – швидкості механізму $\omega_m(t)$. Графік 1 відповідає заданому значенню швидкості $\omega_{m_3}(t)$, графік 2 – швидкості на виході моделі двомасової системи $\omega_{m_{dc}}(t)$ і графік 3 – вихідній координаті

наті побудованої нечіткої моделі гібридної мережі $\omega_{M_{HM}}(t)$. Як бачимо, графіки 2 і 3 практично співпадають.

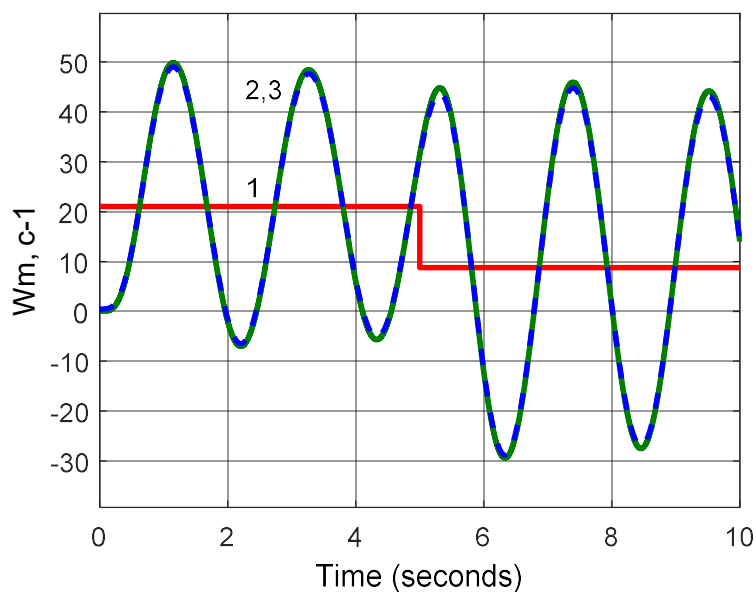


Рис. 3.17. Графіки швидкості механізму двомасової електромеханічної системи:

1 – задане значення швидкості $\omega_{M_s}(t)$; 2 – швидкість на виході моделі двомасової системи $\omega_{M_{dc}}(t)$;

3 – вихідна координата нечіткої моделі гібридної мережі $\omega_{M_{HM}}(t)$.

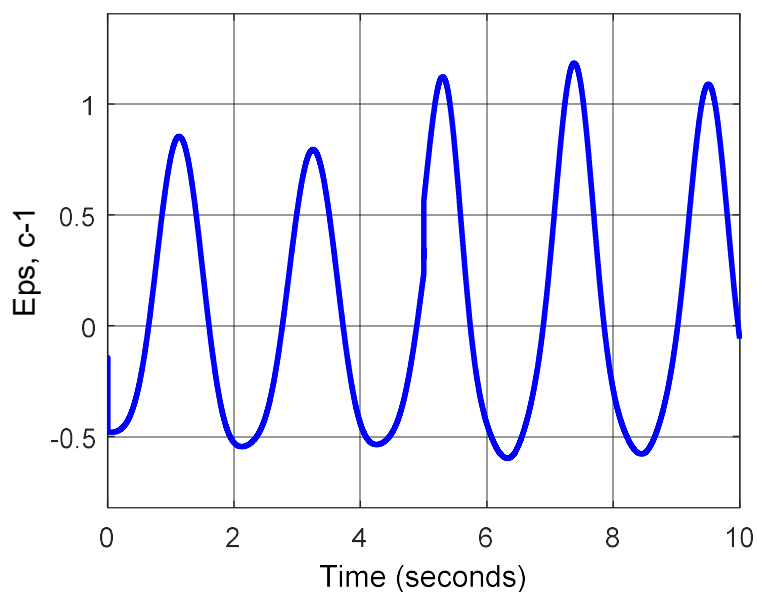


Рис. 3.18. Графік помилки ідентифікації $\varepsilon(t) = \omega_{M_{dc}}(t) - \omega_{M_{HM}}(t)$

На рис. 3.18 зображений графік різниці вказаних швидкостей: $\varepsilon(t) = \omega_{\text{мдс}}(t) - \omega_{\text{мнм}}(t)$, тобто графік помилки ідентифікації. З аналізу графіків виходить, що при зміні швидкості механізму в межах від $+50\text{с}^{-1}$ до -30с^{-1} значення $\varepsilon(t)$ знаходиться в межах від $+1,5\text{с}^{-1}$ до $-0,5\text{с}^{-1}$, тобто помилка ідентифікації не перевищує 4%.

Таким чином, перевірка побудованої нечіткої моделі гібридної мережі показує достатньо високий ступінь її адекватності реальним даним, що дозволяє зробити висновок про можливість її практичного використання для вирішення завдань ідентифікації. В цьому випадку нечіткі моделі адаптивних систем нейро-нечіткого висновку можуть вважатися новим і конструктивним інструментом побудови моделей динамічних об'єктів з складними кінематичними ланцюгами.

Розглянутий підхід є перспективним напрямом для побудови і використання відповідних нечітких моделей динамічних об'єктів і систем. Дійсно, загальним для всіх цих інструментів з позицій технічного аналізу є відсутність апріорних припущень про структуру і параметри об'єктів, що цілком узгоджується з початковими передумовами побудови нечітких моделей адаптивних систем нейро-нечіткого висновку.

3.9 Завдання для самостійного виконання

Варіанти завдань до побудови нейрон-нечітких моделей двомасових і трьохмасових електромеханічних систем з використанням GUI-інтерфейсу NNTool наведено в додатку А.

3.10 Контрольні питання по темі заняття

1. Як реалізовані гібридні мережі в пакеті прикладних програм Fuzzy Logic Toolbox системи MATLAB?
2. Поясніть призначення і можливості редактора ANFIS.
3. Перечисліть і поясніть пункти меню редактора ANFIS.
4. Наведіть методику синтезу гібридної мережі для побудови моделі двомасової електромеханічної системи.
5. Як формуються навчальні, тестові і перевіірочні дані?
6. Як вибирається кількість термів вхідних і вихідного сигналів системи нечіткого висновку?
7. Як можна візуалізувати систему нечіткого висновку?
8. Які методи використовуються для навчання гібридної мережі?
9. За допомогою яких засобів можна виконувати корекцію параметрів гібридної мережі?
10. Як виконується моделювання нечіткої моделі гібридної мережі в Simulink?

Практичне заняття 4

СИНТЕЗ НЕЧІТКИХ РЕГУЛЯТОРІВ З ВИКОРИСТАННЯМ СИСТЕМИ MATLAB

4.1 Мета заняття

Ознайомлення з принципами синтезу нечітких регуляторів та їх застосуванням для управління динамічними об'єктами. Набути практичних навичок у створенні, налаштуванні та моделюванні нечітких регуляторів за допомогою інструментів Fuzzy Logic Toolbox MATLAB. Засвоєння методів формування бази правил, вибору функцій приналежності та оцінки ефективності роботи нечітких регуляторів у системах автоматичного керування.

4.2 Розробка моделі системи управління з нечітким регулятором

Синтез нечітких регуляторів розглянемо на прикладі управління двомасовою електромеханічною системою механізму підйому мостового крану. Визначимо мету управління як забезпечення задовільних показників якості функціонування двомасової електромеханічної системи. Будемо оцінювати якість функціонування системи по величині миттєвої помилки регулювання $\varepsilon(t)$, тобто різниці між заданим значенням швидкості обертання механізму $\omega_z(t)$ і реальною швидкістю обертання механізму $\omega_m(t)$. Очевидно, в регулятор, що забезпечує досягнення мети управління, повинна поступати інформація про величину помилки регулювання, при цьому регулятор повинен виробляти управляючий сигнал $U_{\text{рег}}(t)$, що подається на двомасову електромеханічну систему.

Оскільки двомасова система є інерційною, то при регулюванні слід враховувати похідну від помилки регулювання, в якості якої можна використовувати момент пружності $M_{\text{пр}}(t)$ в кінематичному ланцюзі електроприводу. Тобто, на вхід регулятора слід подавати також сигнал $M_{\text{пр}}(t)$.

Системи нечіткого висновку, створені за допомогою пакету Fuzzy Logic Toolbox, допускають інтерпретацію з інструментальними засобами Simulink. Модель системи управління двомасовою електромеханічною системою з нечітким регулятором, побудована в Simulink показана на рис. 4.1. Схема містить блок **Fuzzy Logic Controller**, який співставляється з системою нечіткого висновку, записаною у файлі Kran.fis (порядок формування системи нечіткого висновку наведено далі) і підсистеми Subsystem, тобто моделі двомасової електромеханічної системи (рис. 4.2). Дана модель з'являється у відповідному вікні при активізації блоку Subsystem подвійним клацанням лівої кнопки миші.

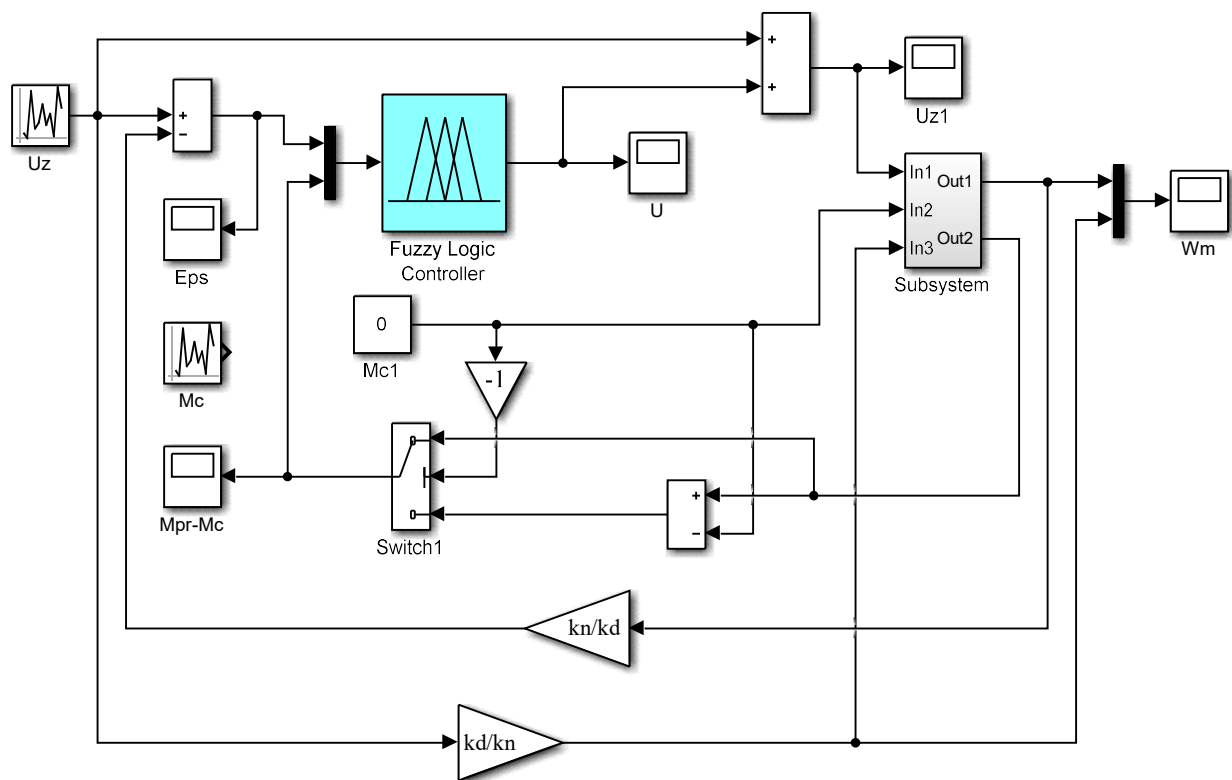


Рис. 4.1. Модель системи управління з нечітким регулятором

Як зазначалося, на входи нечіткого регулятора подається помилка регулювання $\varepsilon(t)$ і момент пружності $M_{пр}(t)$ (точніше – різниця між моментом пружності $M_{пр}(t)$ і моментом статичного навантаження $M_c(t)$, тобто $M_{пр}(t) - M_c(t)$). За допомогою регулятора формується управляюча дія $U_{пер}(t)$, що алгебраїчно підсумовується з напругою завдання $U_3(t)$ і визначається напруга $U_{31}(t)$, що діє на двомасову систему.

Виконаємо розробку системи нечіткого висновку в інтерактивному режимі за допомогою графічних засобів пакету Fuzzy Logic Toolbox.

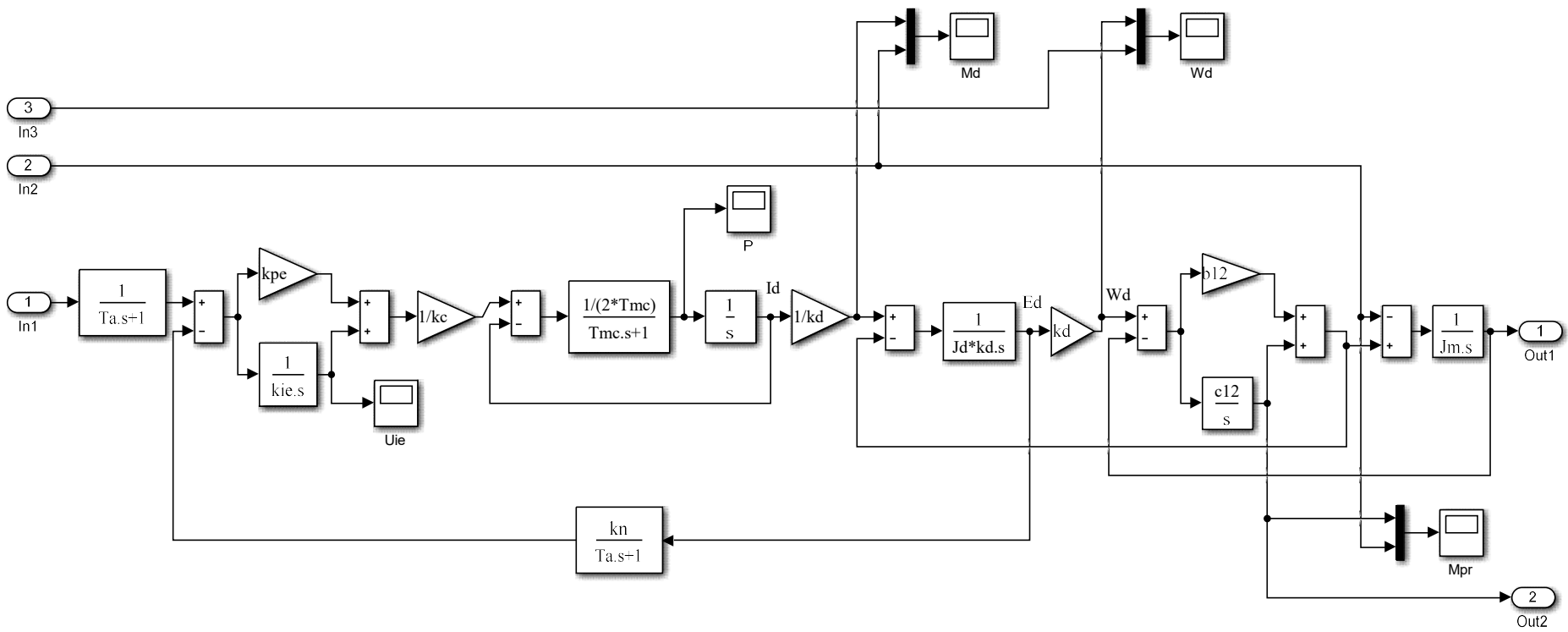


Рис. 4.2 – Модель керованого об'єкту нечіткого регулятора

4.3 Формування бази правил системи нечіткого висновку

Для формування бази правил системи нечіткого висновку необхідно заздалегідь визначити вхідні і вихідні лінгвістичні змінні. Очевидно, в даному випадку вхідні лінгвістичні змінні – це помилка регулювання $\varepsilon(t)$: ε – "помилка регулювання", і момент пружності $M_{\text{пр}}(t)$: $M_{\text{пр}}$ – "момент пружності". Вихідна лінгвістична змінна – напруга на виході регулятора $U_{\text{рег}}(t)$: $U_{\text{рег}}$ – "напруга регулятора".

Як терм-множини першої лінгвістичної змінної – помилка регулювання $\varepsilon(t)$ – використовуватимемо множину $T_1 = \{\text{"негативна велика помилка"}, \text{"негативна мала помилка"}, \text{"близька до нуля помилка"}, \text{"позитивна мала помилка"}, \text{"позитивна велика помилка"}\}$, яку можна скорочено записати в символічному виді $T_1 = \{\text{NGE}, \text{NSE}, \text{ZRE}, \text{PSE}, \text{PGE}\}$. В якості терм-множини другої лінгвістичної змінної – моменту пружності $M_{\text{пр}}(t)$ – використовуватимемо множину $T_2 = \{\text{"негативний великий момент"}, \text{"негативний малий момент"}, \text{"близький до нуля момент"}, \text{"позитивний малий момент"}, \text{"позитивний великий момент"}\}$, яка записується в символічному виді $T_2 = \{\text{NGM}, \text{NSM}, \text{ZRM}, \text{PSM}, \text{PGM}\}$. В якості терм-множини вихідної лінгвістичної змінної – напруги на виході регулятора $U_{\text{рег}}(t)$ – використовуватимемо множину: $T_3 = \{\text{"негативна велика напруга"}, \text{"негативна мала напруга"}, \text{"близька до нуля напруга"}, \text{"позитивна мала напруга"}, \text{"позитивна велика напруга"}\}$, яка записується в символічному виді $T_3 = \{\text{NGU}, \text{NSU}, \text{ZRU}, \text{PSU}, \text{PGU}\}$.

Функції приналежності кожного терма кожної множини задаються далі (див. рис. 4.5, рис. 4.7 і рис. 4.8).

На підставі аналізу експериментальних перехідних процесів при пуску і набросі навантаження сформуємо базу правил системи нечіткого висновку. Оскільки розглядаються дві вхідні лінгвістичні змінні, кожна з яких має 5 термів, то система нечіткого висновку міститиме 25 правил нечітких висновків наступного вигляду:

ПРАВИЛО_1: ЯКЩО " $\varepsilon \in \text{NGE}$ " І " $M_{\text{пр}} \in \text{NGM}$ " ТО " $U_{\text{рег}} \in \text{PGU}$ "

ПРАВИЛО_2: ЯКЩО " $\varepsilon \in \text{NGE}$ " І " $M_{\text{пр}} \in \text{NSM}$ " ТО " $U_{\text{рег}} \in \text{PSU}$ "

ПРАВИЛО_3: ЯКЩО " $\varepsilon \in \text{NGE}$ " І " $M_{\text{пр}} \in \text{ZRM}$ " ТО " $U_{\text{рег}} \in \text{PSU}$ "

ПРАВИЛО_4: ЯКЩО " $\varepsilon \in \text{NGE}$ " І " $M_{\text{пр}} \in \text{PSM}$ " ТО " $U_{\text{рег}} \in \text{ZRU}$ "

ПРАВИЛО_5: ЯКЩО " $\varepsilon \in \text{NGE}$ " І " $M_{\text{пр}} \in \text{PGM}$ " ТО " $U_{\text{рег}} \in \text{NSU}$ "

ПРАВИЛО_6: ЯКЩО " $\varepsilon \in \text{NSE}$ " І " $M_{\text{пр}} \in \text{NGM}$ " ТО " $U_{\text{рег}} \in \text{PSU}$ "

ПРАВИЛО_7: ЯКЩО " $\varepsilon \in \text{NSE}$ " І " $M_{\text{пр}} \in \text{NSM}$ " ТО " $U_{\text{рег}} \in \text{PSU}$ "

ПРАВИЛО_8: ЯКЩО " $\varepsilon \in \text{NSE}$ " І " $M_{\text{пр}} \in \text{ZRM}$ " ТО " $U_{\text{рег}} \in \text{PSU}$ "

ПРАВИЛО_9: ЯКЩО " $\varepsilon \in \text{NSE}$ " І " $M_{\text{пр}} \in \text{PSM}$ " ТО " $U_{\text{рег}} \in \text{ZRU}$ "

ПРАВИЛО_10: ЯКЩО " $\varepsilon \in \text{NSE}$ " І " $M_{\text{пр}} \in \text{PGM}$ " ТО " $U_{\text{рег}} \in \text{NSU}$ "

ПРАВИЛО_11: ЯКЩО " $\varepsilon \in \text{ZRE}$ " І " $M_{\text{пр}} \in \text{NGM}$ " ТО " $U_{\text{рег}} \in \text{PSU}$ "

ПРАВИЛО_12: ЯКЩО " $\varepsilon \in ZRE$ " І " $M_{np} \in NSM$ " ТО " $U_{per} \in ZRU$ "
 ПРАВИЛО_13: ЯКЩО " $\varepsilon \in ZRE$ " І " $M_{np} \in ZRM$ " ТО " $U_{per} \in ZRU$ "
 ПРАВИЛО_14: ЯКЩО " $\varepsilon \in ZRE$ " І " $M_{np} \in PSM$ " ТО " $U_{per} \in ZRU$ "
 ПРАВИЛО_15: ЯКЩО " $\varepsilon \in ZRE$ " І " $M_{np} \in PGM$ " ТО " $U_{per} \in NSU$ "
 ПРАВИЛО_16: ЯКЩО " $\varepsilon \in PSE$ " І " $M_{np} \in NGM$ " ТО " $U_{per} \in PSU$ "
 ПРАВИЛО_17: ЯКЩО " $\varepsilon \in PSE$ " І " $M_{np} \in NSM$ " ТО " $U_{per} \in ZRU$ "
 ПРАВИЛО_18: ЯКЩО " $\varepsilon \in PSE$ " І " $M_{np} \in ZRM$ " ТО " $U_{per} \in NSU$ "
 ПРАВИЛО_19: ЯКЩО " $\varepsilon \in PSE$ " І " $M_{np} \in PSM$ " ТО " $U_{per} \in NSU$ "
 ПРАВИЛО_20: ЯКЩО " $\varepsilon \in PSE$ " І " $M_{np} \in PGM$ " ТО " $U_{per} \in NSU$ "
 ПРАВИЛО_21: ЯКЩО " $\varepsilon \in PGE$ " І " $M_{np} \in NGM$ " ТО " $U_{per} \in PSU$ "
 ПРАВИЛО_22: ЯКЩО " $\varepsilon \in PGE$ " І " $M_{np} \in NSM$ " ТО " $U_{per} \in ZRU$ "
 ПРАВИЛО_23: ЯКЩО " $\varepsilon \in PGE$ " І " $M_{np} \in ZRM$ " ТО " $U_{per} \in NSU$ "
 ПРАВИЛО_24: ЯКЩО " $\varepsilon \in PGE$ " І " $M_{np} \in PSM$ " ТО " $U_{per} \in NSU$ "
 ПРАВИЛО_25: ЯКЩО " $\varepsilon \in PGE$ " І " $M_{np} \in PGM$ " ТО " $U_{per} \in NGU$ "

Базу правил системи нечіткого висновку представимо для зручності у вигляді таблиці 4.1.

Таблиця 4.1

База правил системи нечіткого висновку

		Помилка регулювання				
		NGW	NSW	ZRW	PSW	PGW
Момент пружності	NGM	PGU	PSU	PSU	PSU	PSU
	NSM	PSU	PSU	ZRU	ZRU	ZRU
	ZRM	PSU	PSU	ZRU	NSU	NSU
	PSM	ZRU	ZRU	ZRU	NSU	NSU
	PGM	NSU	NSU	NSU	NSU	NGU

Представленої інформації досить для проектування нечіткої системи. Така система матиме 2 входи, один вихід, двадцять п'ять правил типу "якщо... то" і по п'ять функцій приналежності входів і виходу. Побудуємо дану систему, використовуючи алгоритм виведення Mamdani.

4.4 Розробка системи нечіткого висновку

Процес розробки системи нечіткого висновку в інтерактивному режимі полягає у виконанні наступної послідовності дій.

1. Викликати редактор систем нечіткого виведення **FIS**, для чого у вікні команд набрати ім'я відповідної функції fuzzy. Після виконання цієї команди на екрані з'явиться графічний інтерфейс редактора **FIS** з ім'ям системи нечіткого виведення **Untitled** і типом системи нечіткого висновку (Мамдані), запропонованими за умовчанням (рис. 4.3).

2. Оскільки в прикладі, що розглядається система нечіткого висновку з двома входами, необхідно додати в систему **FIS**, що розробляється, ще одну вхідну змінну. Для цього слід виконати команду меню **Edi\Add Variable...Input**. В результаті виконання цієї команди на діаграмі системи нечіткого висновку з'явиться новий жовтий прямокутник з ім'ям другої вхідної змінної: **input2**.

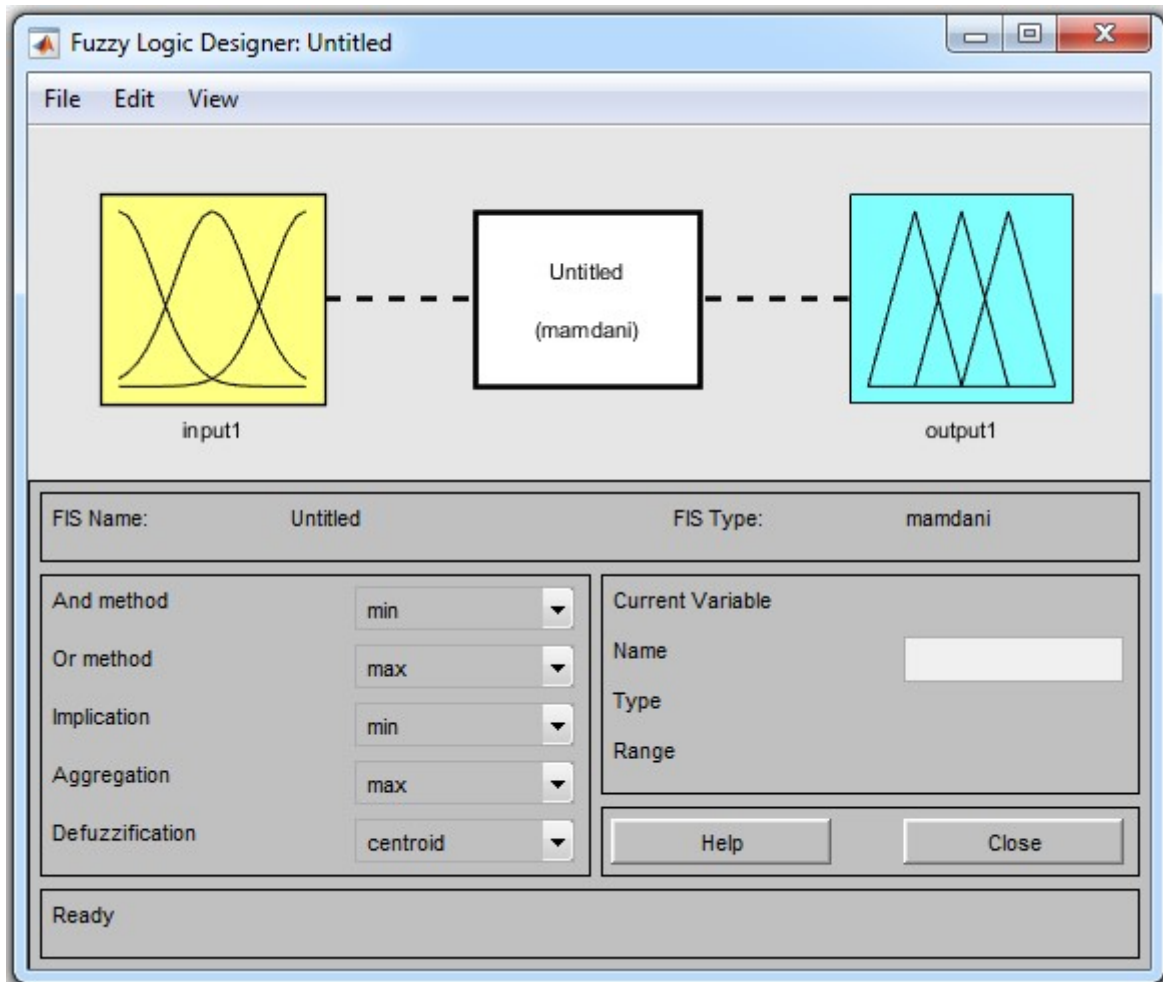


Рис. 4.3. Вид редактора **FIS Editor** після додавання другої вхідної змінної

3. Змінимо імена вхідних і вихідних змінних, запропонованих системою MATLAB за умовчанням. Для цього необхідно виділити прямокутник з ім'ям відповідної змінної, виконавши клацання на його зображенні на діаграмі (сторони виділеного прямокутника мають червоний колір). Після чого слід набрати нове ім'я змінної в полі введення **Name** в правій частині редактора **FIS**. В прикладі, що розглядається, роблячи одноразове клацання на блоку input 1, міняємо його ім'я на «Epsilon» (помилка регулювання $\varepsilon(t)$), завершуючи введення нового імені натисненням клавіші Enter. Аналогічним чином встановлюємо ім'я «Mpr» ($M_{pr}(t)$) блоку input2 і «U» ($U(t)$) – вихідному блоку (справа вгорі) output 1. Вид вікна редактора після вказаних дій приведений на рис. 4.4.

Залишимо без зміни запропоновані системою MATLAB за умовчанням: метод нечіткого логічного **I (And method)** – значення "min", метод нечіткого логічного **АБО (Or method)** – значення "max", метод імплікації (**Implication**) – значення "min", метод агрегації (**Aggregation**) – значення "max" ' і метод дефазифікації (**Defuzzification**) – значення "centroid". Очевидно, ці значення можуть бути змінені при необхідності.

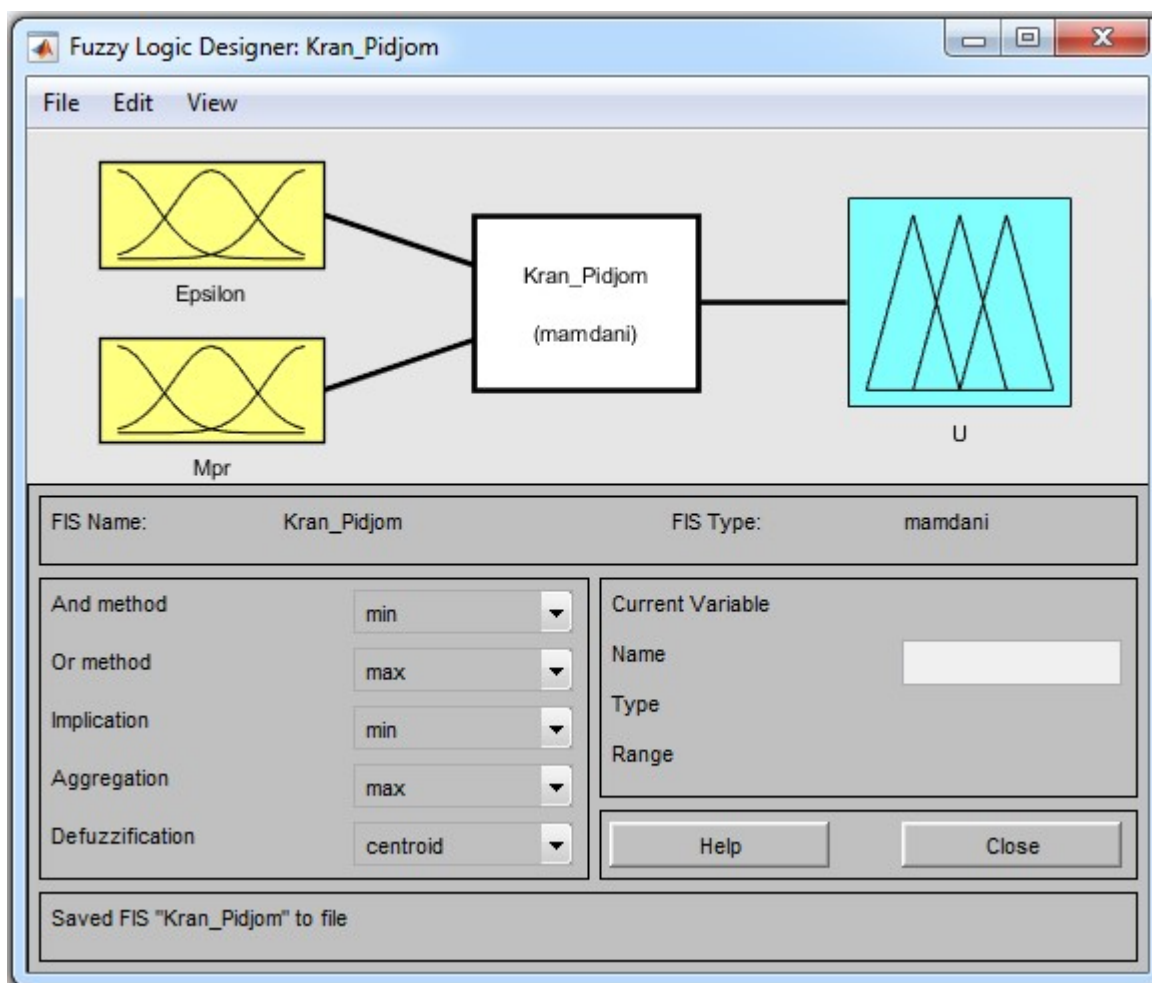


Рис. 4.4. Вид вікна **FIS Editor** після завдання структури системи

4. Змінимо ім'я системи нечіткого висновку (Untitled), запропоноване за умовчанням. Для цього збережемо створювану структуру FIS з ім'ям Kran_Pidjom.fis, в робочому просторі командою меню **File\Export\To Workspace...**, або на диску, виконавши команду меню **File\Export\To Disk...** При цьому буде викликано стандартне діалогове вікно збереження файлу, в якому користувачеві пропонується ввести ім'я відповідного файлу (розширення файлу приписується автоматично).

5. Задамо тепер терми і їх функції приналежності для вхідних і вихідних змінних системи нечіткого висновку. Вкажемо, що програму – редактор функцій приналежності – можна відкрити трьома способами:

- через пункт меню **Edit\Membership functions**;
- подвійним клацанням на значку, що відображає відповідну змінну;
- натисненням клавіш **Ctrl+2**.

Будь-яким з приведених способів перейдемо до даної програми.

Завдання і редагування функцій приналежності почнемо із змінної «Epsilon» (рис. 4.5).

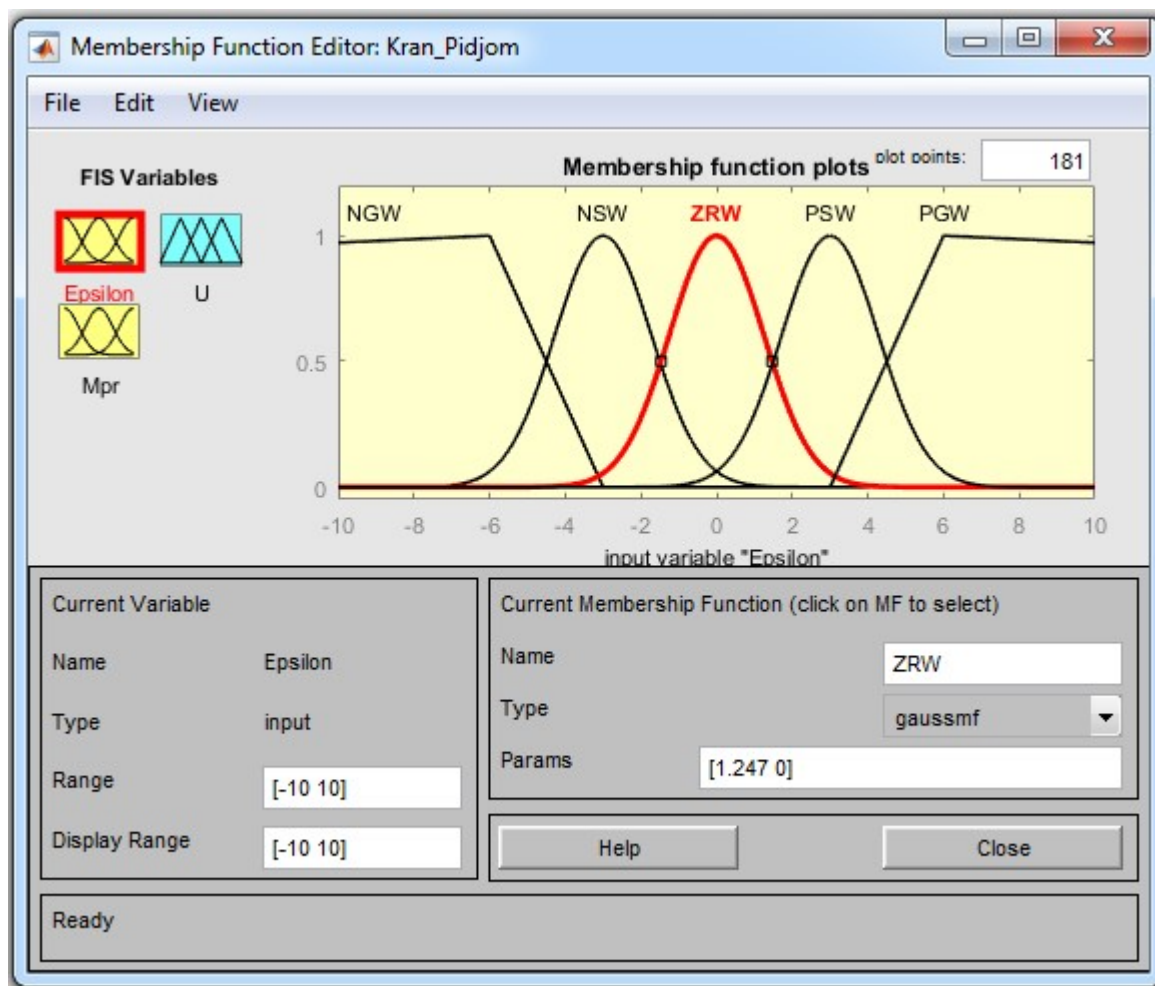


Рис. 4.5. Функції приналежності змінної «Epsilon» (помилка регулювання $\varepsilon(t)$)

Спочатку в полях **Range i Display Range** встановимо діапазон зміни і відображення цієї змінної від -10 до $+10$, підтверджуючи введення натисненням клавіші **Enter**. Потім через пункти меню **Edit\Remove All MFs**, **Edit\Add MFs** перейдемо до діалогового вікна виду рис. 4.6 і задамо в ньому п'ять функцій приналежності трикутної форми (trimf) (або гауссова типу (gaussmf)).

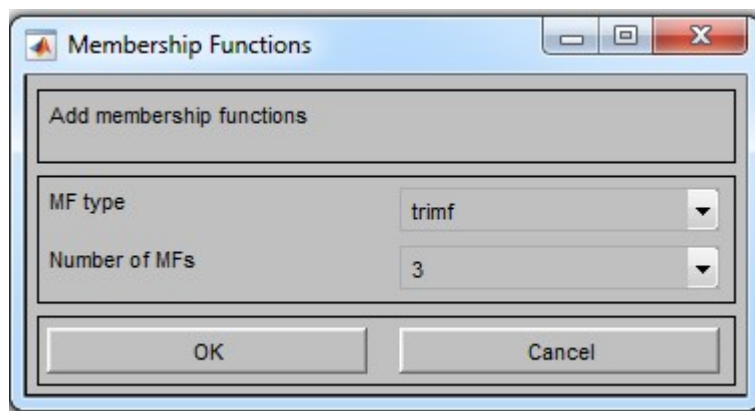


Рис. 4.6. Вид вікна завдання типа і кількості функцій приналежності

Натиснемо кнопку **OK** і повернемося у вікно редактора функцій приналежності. Замінімо імена заданих функцій, для чого підводимо покажчик до потрібної кривої і клацаємо лівою кнопкою миші. Крива вибирається, забарвлюючись в червоний колір, після чого за допомогою миші її можна посунути в потрібну сторону (точнішу установку можна провести, змінюючи числові значення в полі **Params** (Параметри) – в даному випадку кожній функції приналежності відповідають два параметри, при цьому перший визначає розмах кривої, а другий – положення її центру). У полі **Name** змінимо ім'я для вибраної кривої (завершуючи введення кожного імені натисненням клавіші **Enter**). Задамо всім п'яти кривим нові імена, наприклад, так:

- найлівішій – NGE (негативна велика для помилки);
- наступній – NSE (негативна мала для помилки);
- центральній – ZRE (дуже мала для помилки);
- наступній за нею справа – PSE (позитивна мала для помилки);
- найправішій – PGE (позитивна велика для помилки).

Тип, розмах і положення заданих функцій вибирається в процесі настройки Fuzzy регулятора шляхом багатократного розрахунку перехідних процесів багатомасової системи при різних типах і параметрах функцій приналежності і вибору варіанту, при якому показники якості перехідних процесів мають найкращі значення.

Клацанням на значку «Mpr» ($M_{pr}(t)$) увійдемо до вікна редагування функцій приналежності для цієї змінної (рис.4.7). Задамо спочатку діапазон її зміни від -60 до $+60$, а потім, поступаючи як раніше, задамо п'ять функцій приналежності трикутної форми (trimf) з іменами: «NGM» («негативна велика для моменту»), «NSM» («негативна мала для моменту»), ZRM («дуже мала для моменту»), «PSM» («позитивна мала для моменту») і «PGM» («позитивна велика для моменту»).

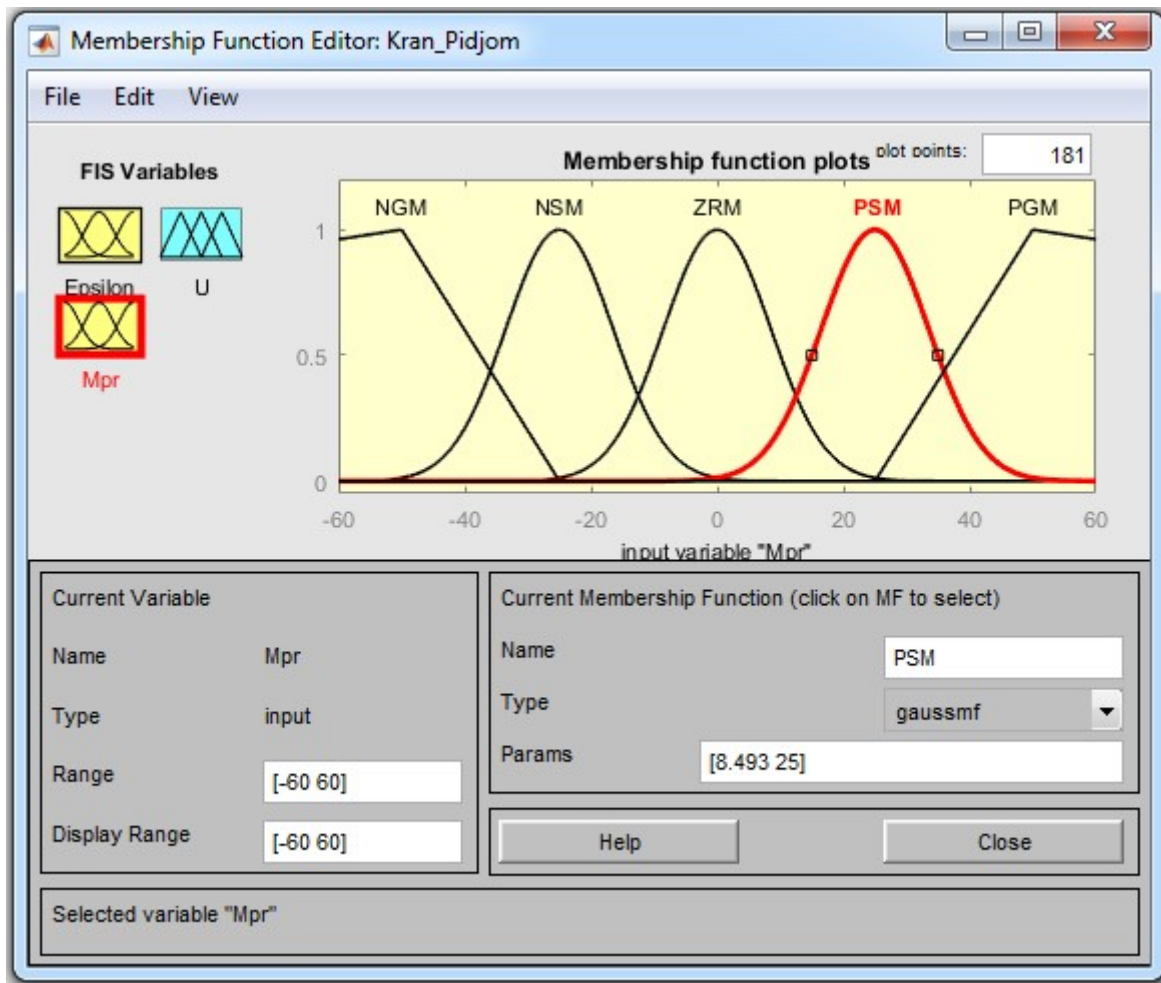


Рис. 4.7. Функції приналежності змінної «Mpr» (моменту пружності $M_{пр}(t)$)

Для вихідної змінної $U(U(t))$ вкажемо спочатку діапазон зміни (від -5 до $+5$), потім задамо п'ять функцій приналежності гауссова типу: для центральної функції `gauss2mf`, а для інших – `gaussmf` з іменами «NGU («негативна велика для вихідної змінної $U(t)$ »), «NSU» («негативна мала для вихідної змінної $U(t)$ »), ZRU («дуже мала для вихідної змінної $U(t)$ »), «PSU» («позитивна мала для вихідної змінної $U(t)$ ») і «PGU» («позитивна велика для вихідної змінної $U(t)$ ») так, як це представлено на рис. 4.8.

Як указувалося раніше, тип і параметри функцій уточнюються в процесі настройки Fuzzy регулятора.

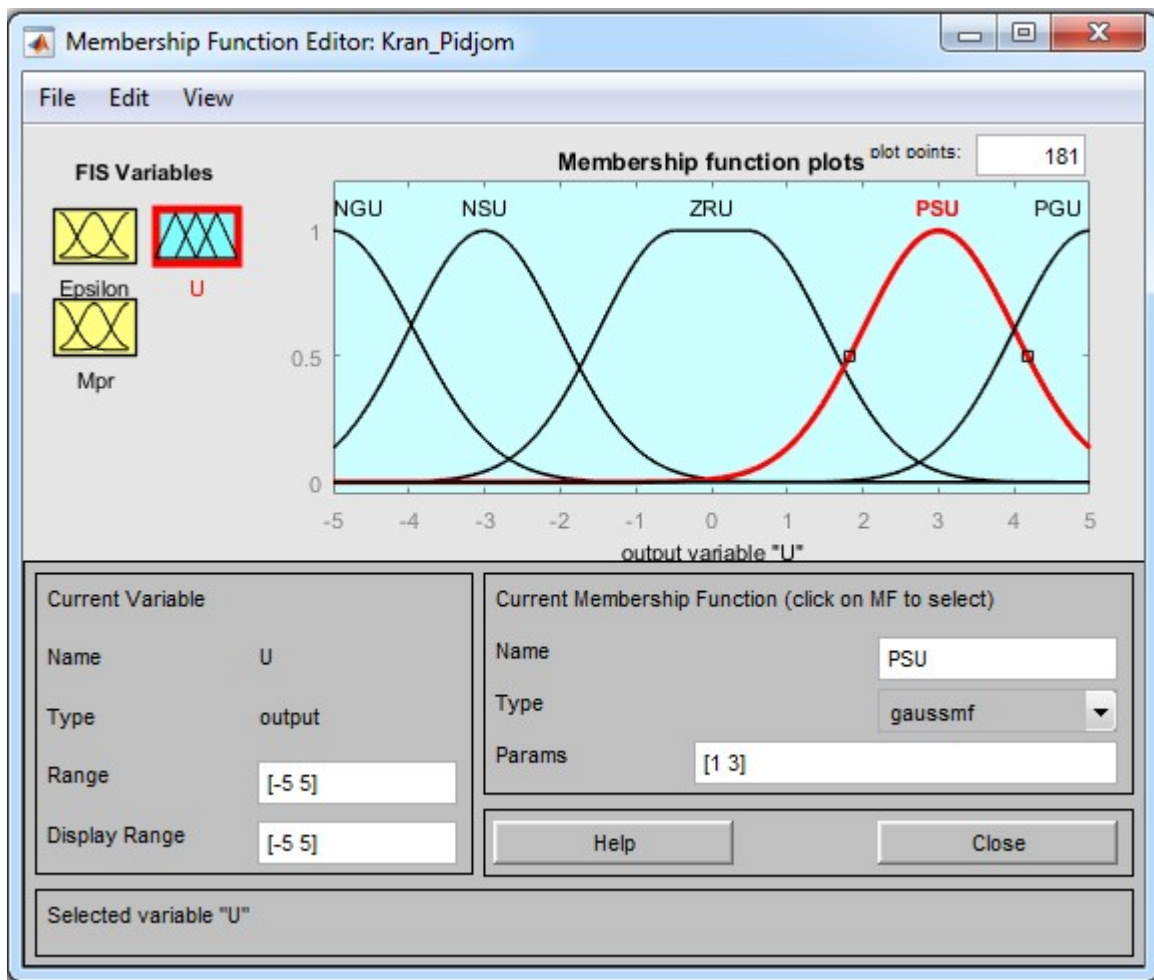


Рис. 4.8. Функції приналежності «U» (вихідної змінної $U(t)$)

6. Тепер настала черга визначити правила нечіткого висновку для нечіткої системи, що розроблялася. Для цієї мети слід скористатися редактором правил, який може бути викликаний одним з наступних способів:

- подвійним клацанням на значку квадрата в центрі з ім'ям створюваної системи нечіткого висновку (Kran);
- командою меню **Edit\Rules...**;
- натисненням клавіш **Ctrl+3**.

Оскільки спершу база правил нечіткого висновку порожня, то після виклику редактора правил центральне поле введення не містить ніяких правил. Для їх визначення слід використовувати поля меню і перемикачі в нижній частині графічного інтерфейсу редактора правил. Для завдання першого правила слід залишити виділене за умовчанням поле з ім'ям терма "NGE" для першої вхідної змінної «Epsilon», поле з ім'ям терма "NGM" для другої вхідної змінної «Mpr» і поле з ім'ям терма "PGU" для вихідної змінної «Ureg». Перемикач **Connection** слідує залишити в положенні **and** (логічне **I**) і натиснути на кнопку **Add rule**. Після цього перше правило відобразиться у першому рядку вікна.

Для завдання другого правила, для якого слід виділити імена термів "NGE", "NSM" і "PSU". Аналогічним чином задаються інші правила. Вид вікна редактора правил після їх визначення зображений на рис. 4.9.

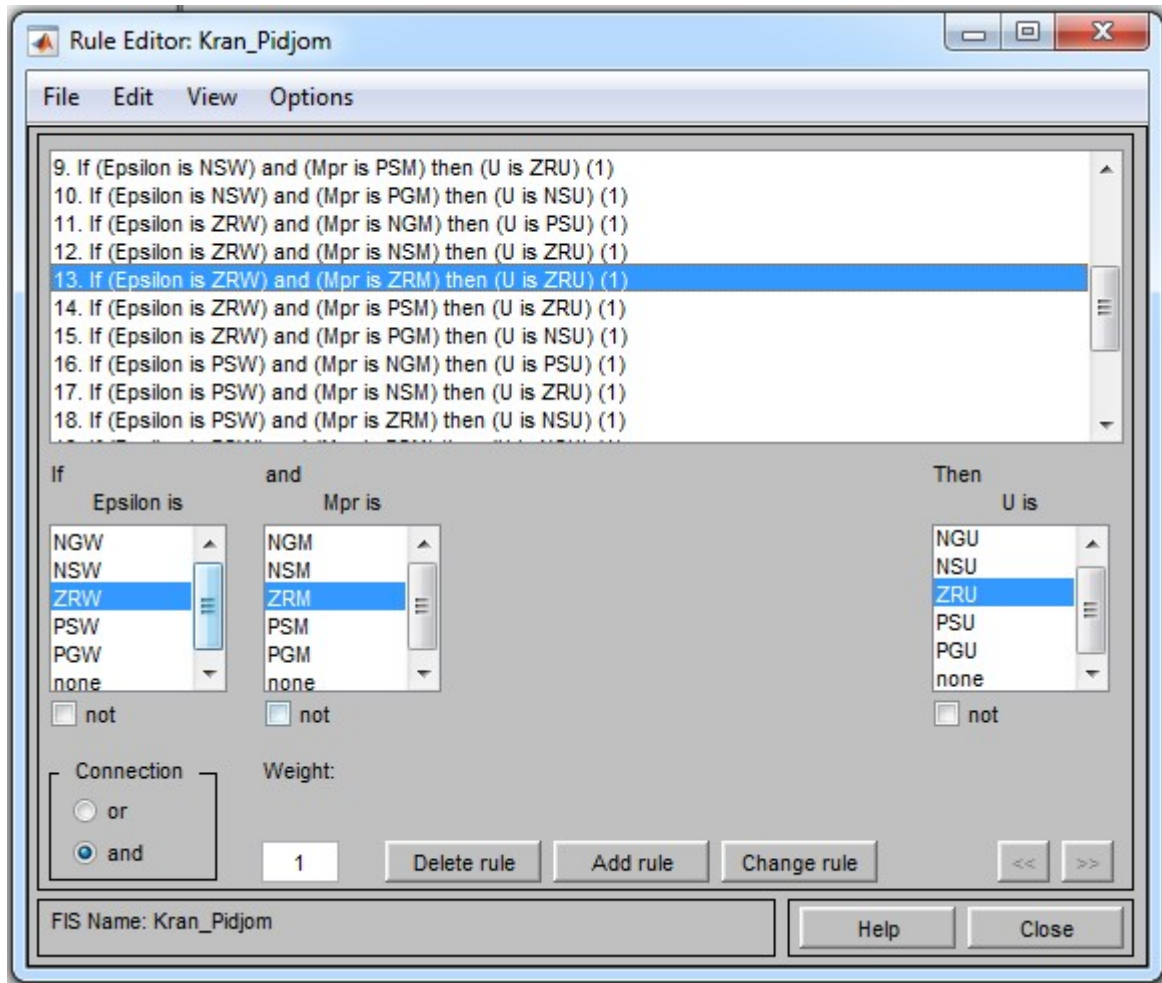


Рис. 4.9. Вікно редактора правил

Відмітимо, що в полі введення **Weight** відображається вага кожного правила, яку можна змінювати в межах інтервалу $[0, 1]$ (залишимо без зміни його значення за умовчанням, рівне 1 для всіх правил). Ця ж вага правил записується в круглих дужках у вікні правил після кожного з правил нечіткого висновку.

7. Після завдання правил нечіткого висновку виявляється можливим отримати результат нечіткого висновку (значення вихідної змінної) для конкретних значень вхідних змінних. З цією метою необхідно відкрити програму перегляду правил одним з наступних способів:

- командою меню **View\Rules** редактора **FIS**;
- командою меню **View\Rules** редактора функцій приналежності;
- командою меню **View\Rules** редактора правил;
- натисненням клавіш **Ctrl+5**.

Після виклику програми перегляду правил для системи нечіткого висновку, що розглядається в прикладі, відкриється вікно (рис. 4.10). У правій частині вікна в

графічній формі представлені функції приналежності аргументів «Epsilon» (помилки регулювання $\varepsilon(t)$), і «Mpr» (моменту пружності $M_{пр}(t)$), в лівій – функції приналежності змінної виходу «U» ($U(t)$) з поясненням механізму ухвалення рішення.

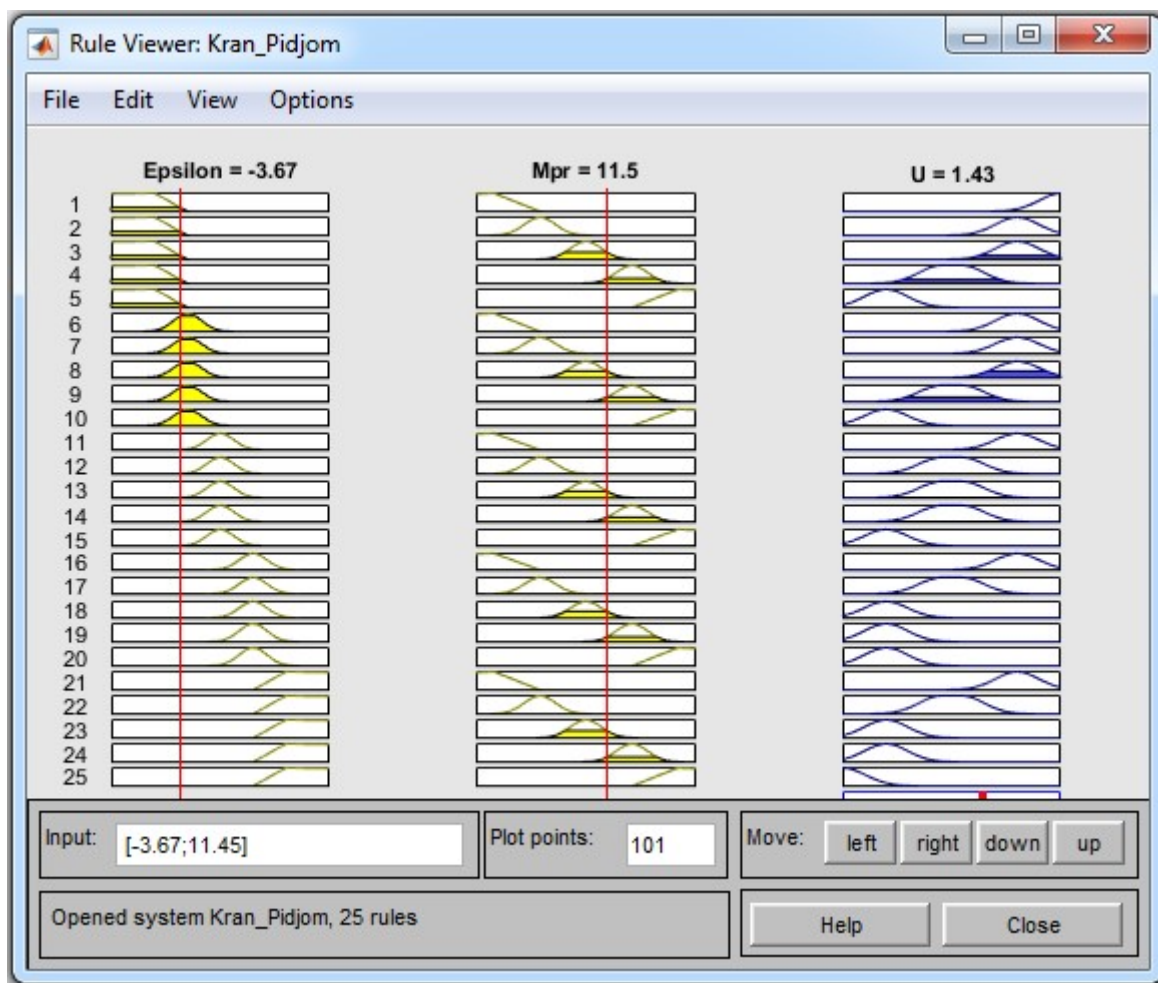


Рис. 4.10. Вікно перегляду правил

Червона вертикальна межа, що перетинає графіки в правій частині вікна, яку можна переміщати за допомогою миші, дозволяє змінювати значення змінних входу (це ж можна робити, задаючи числові значення в полі **Input** (Вхід)), при цьому відповідно змінюються значення «Epsilon» і «Mpr» в правій верхній частині вікна і, відповідно, значення вихідної змінної «Ureg» ($U_{пер}(t)$). Таким чином, за допомогою побудованої моделі і вікна перегляду правил можна вирішувати задачу синтезу Fuzzy регулятора. Зміна аргументу шляхом переміщення червоної вертикальної лінії дуже наочно демонструє, як система визначає значення виходу.

8. Для остаточного аналізу розробленої нечіткої моделі може виявитися корисною програма перегляду поверхні нечіткого висновку, яка може бути викликана одним з наступних способів:

- командою меню **View\Surface** з редактора **FIS**;
- командою меню **View\Surface** редактора функцій приналежності;
- командою меню **View\Surface** редактора правил;

командою меню **View\Surface** програми перегляду правил;
натисненням клавіш Ctrl+6.

Графічний інтерфейс програми перегляду поверхні нечіткого висновку для розробленої нечіткої моделі зображений на рис. 4.11.

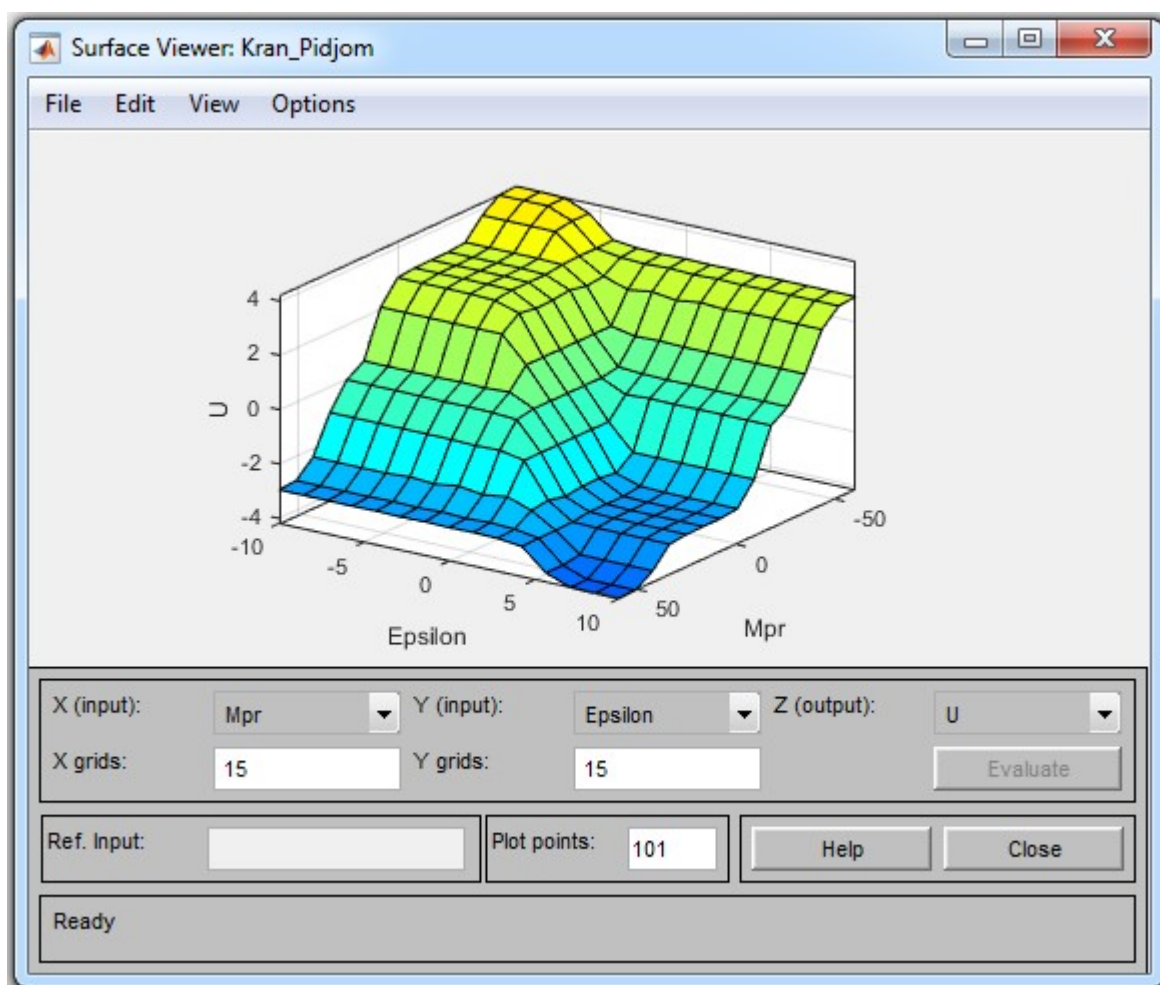


Рис. 4.11. Графічний вид залежності вихідної змінної від вхідних

Ця програма служить для загального аналізу адекватності нечіткої моделі, дозволяючи оцінити вплив зміни значень вхідних нечітких змінних на значення вихідної змінної. За допомогою мишки графік можна повертати на всі боки.

У разі потреби можна отримати графік залежності вихідної змінної від однієї з вхідних змінних. Для цього необхідно вибрати потрібну змінну в списку, що розкривається, **X (input)**, а в списку, що розкривається, **Y (input)**, вибрати значення - **none**-. Наприклад, графік залежності $U_{reg}(U_{per}(t))$ від $M_{np}(M_{np}(t))$ зображений на рис. 4.12.

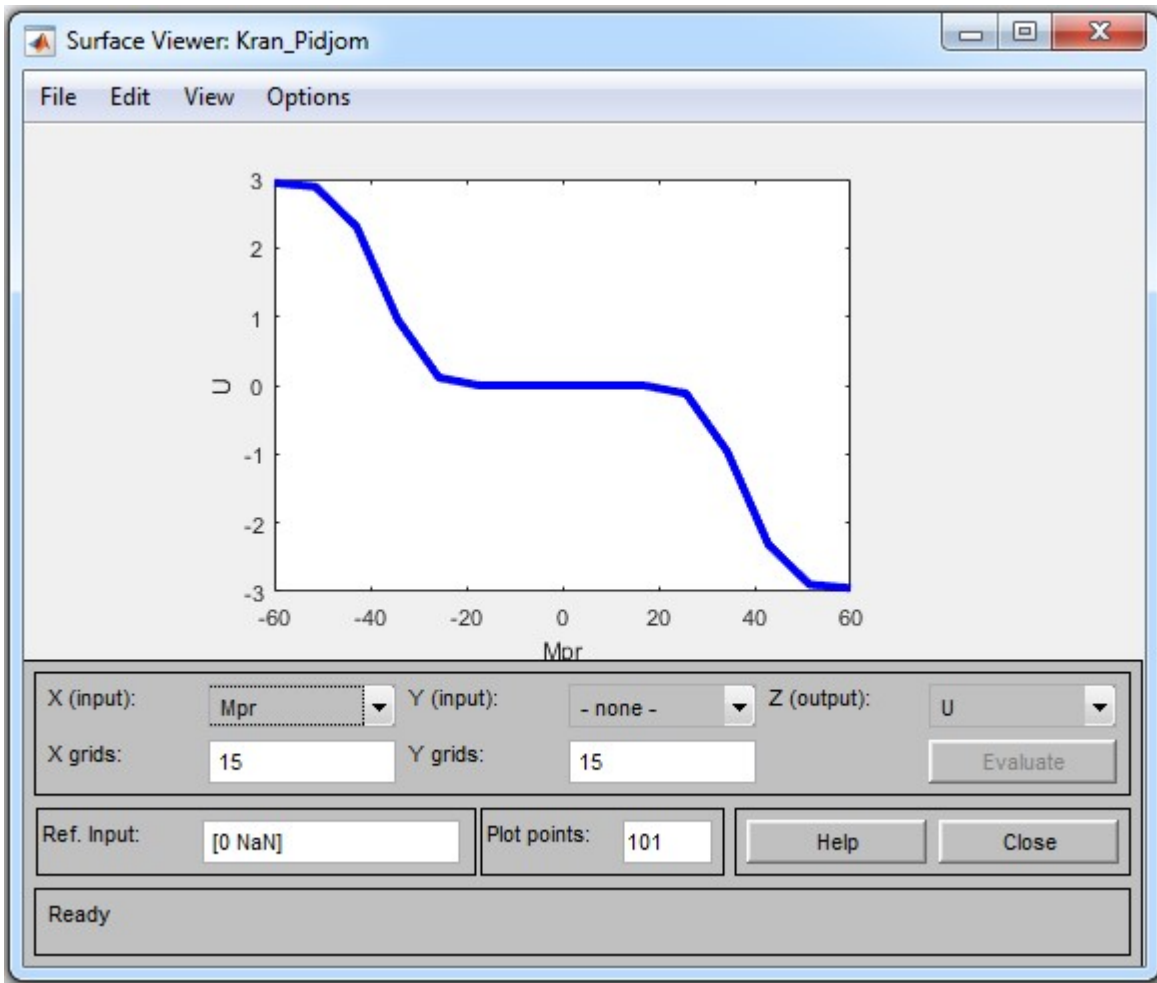


Рис. 4.12. Одновимірна залежність вихідний змінної «U» ($U(t)$) від вхідної – моменту пружності «Mpr» ($M_{пр}(t)$)

Як показали дослідження, змінюючи тип і параметри функцій приналежності, діапазон їх зміни **Range** можна сконструювати нечіткий регулятор, що дозволяє понизити динамічні навантаження в багатомасовій системі.

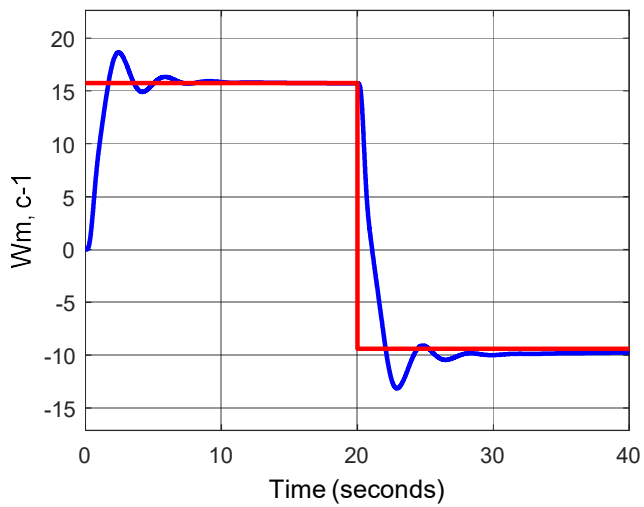
Закінчуючи розгляд процесу розробки простої системи нечіткого висновку в інтерактивному режимі, слід відмітити, що найбільш ефективним цей спосіб виявляється для складних нечітких моделей з великим числом змінних і правил нечіткого висновку. В цьому випадку завдання змінних і функцій приналежності їх термів в графічному режимі, а також візуалізація правил дозволяють істотно зменшити трудомісткість розробки нечіткої моделі, понизити кількість можливих помилок і скоротити загальний час нечіткого моделювання.

В той же час слід пам'ятати, що кількість змінних і правил в нечіткій моделі, які можуть бути візуалізовані, обмежена. Зокрема, якщо число вхідних змінних перевищує 7, то їх відображення у відповідних графічних редакторах відбувається із спотвореннями.

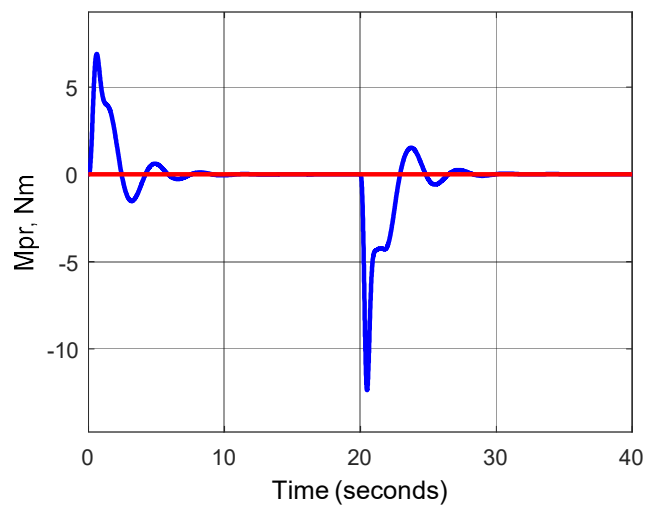
4.5 Моделювання двомасової електромеханічної системи з нечітким регулятором

Для моделювання двомасової системи з нечітким регулятором можуть бути використані моделі Simulink, приведені на рис.4.1 або 4.13. Перш ніж почати моделювання, необхідно зберегти створену структуру **FIS** з ім'ям **Kran.fis** в робочому просторі командою меню **File\Export\To Workspace....** Потім необхідно співставити розроблену систему нечіткого висновку, записану у файлі **Kran.fis** з блоком **Fuzzy Logic Controller**. Для цього шляхом подвійного клацання мишею на зображенні блока відкриємо вікно завдання параметрів блоку. Єдиним параметром **FIS File or Structure**, що задається в вікні, є ім'я системи нечіткого висновку, в даному випадку **Kran**.

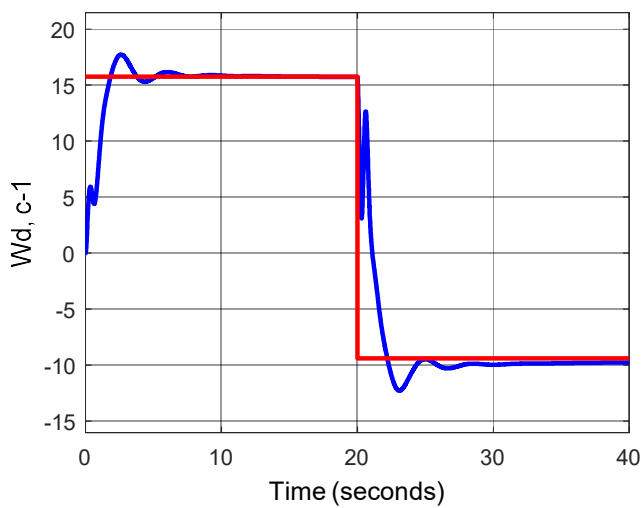
На рис. 4.14 і рис. 4.15 показані графіки перехідних процесів основних змінних стану системи з Fuzzy регулятором в режимі пуску і набросу навантаження. Як видно з графіків, перехідні процеси мають задовільний характер, що істотно знижує динамічні навантаження двомасової системи. (Перехідні процеси двомасової системи, що розглядається, без регулятора мають характер слабо затухаючих коливань).



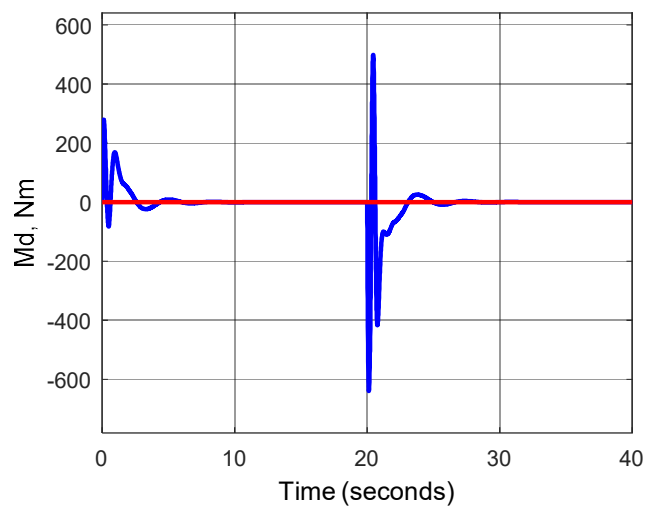
а)



б)



в)



г)

Рис. 4.14. Графіки перехідних процесів двомасової системи з нечітким регулятором по задаючій дії

а) швидкість механізму $\omega_m = f(t)$; б) момент пружності $M_{пр} = f(t)$;

в) швидкість двигуна $\omega_d = f(t)$; г) момент двигуна $M_d = f(t)$

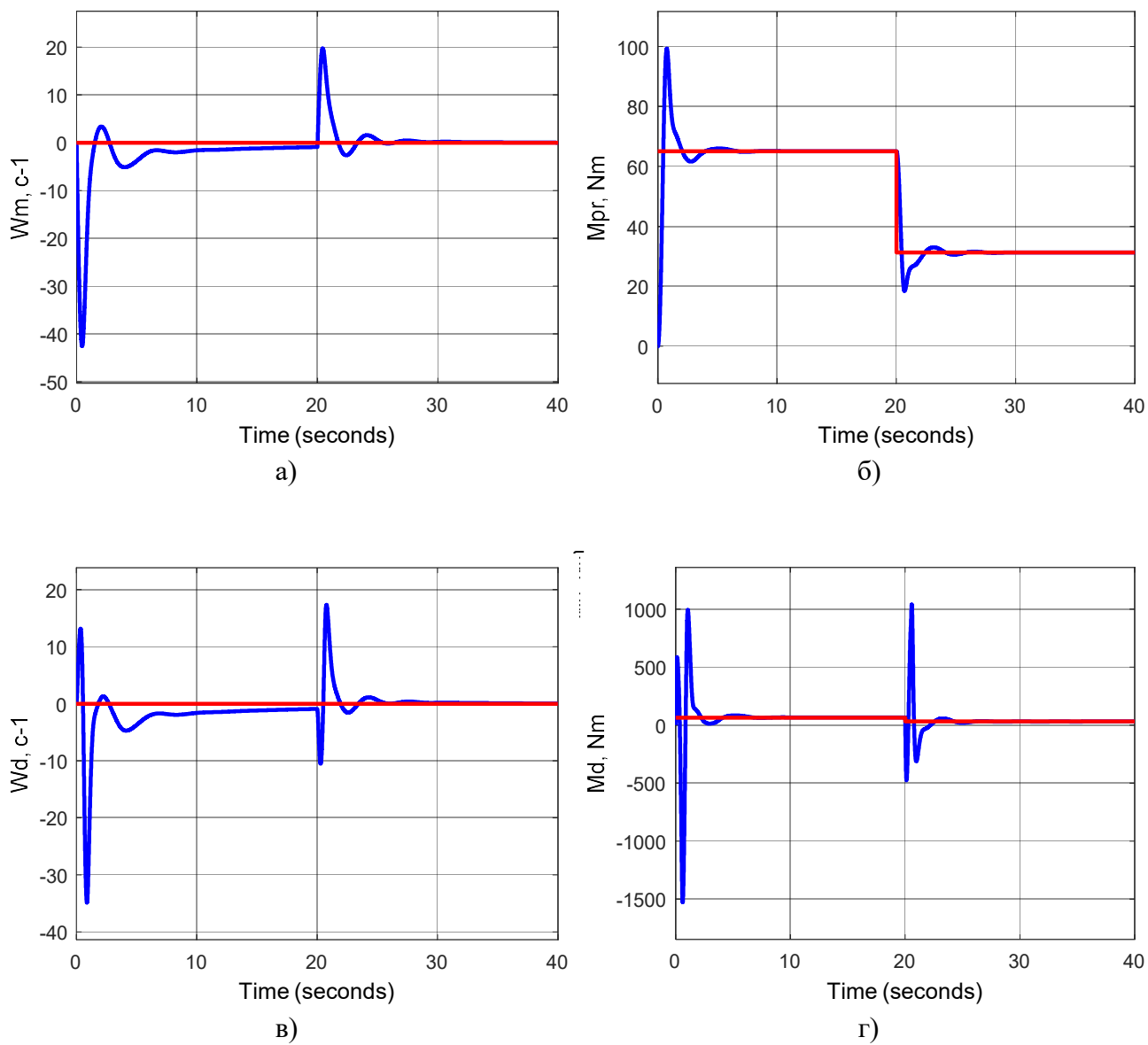


Рис. 4.15 .Графіки перехідних процесів двомасової системи з нечітким регулятором по обурюючій дії

а) швидкість механізму $\omega_m = f(t)$; б) момент пружності $M_{пр} = f(t)$;

в) швидкість двигуна $\omega_d = f(t)$; г) момент двигуна $M_d = f(t)$

4.6 Завдання для самостійного виконання

Варіанти завдань до побудови нечітких моделей двомасових і трьохмасових електромеханічних систем з використанням GUI-інтерфейсу NNTool наведено в додатку А.

4.7 Контрольні питання по темі заняття

1. Зобразіть схему системи управління двомасовою електромеханічною системою з нечітким регулятором?
2. Які сигнали слід подавати на вхід нечіткого регулятора?
3. Як визначаються вхідні і вихідні лінгвістичні змінні системи нечіткого висновку?
4. Які множини використовуються як терм-множини всіх лінгвістичних змінних?
5. Як формується база правил системи нечіткого висновку? Як визначається їх кількість?
6. Поясніть порядок синтезу нечітких регуляторів в середовищі MATLAB.
7. Як задаються терми і їх функції приналежності для вхідних і вихідних змінних системи нечіткого висновку?
8. Як задаються діапазони зміни вхідних і вихідних змінних?
9. Як виконується настройка функцій приналежності вхідних і вихідних змінних?
10. Як задаються правила нечіткого висновку?
11. Якими способами може бути викликана програма перегляду поверхні нечіткого висновку?
12. Як можна отримати графік залежності вихідної змінної від однієї з вхідних змінних?
13. Як виконується моделювання системи управління з нечітким регулятором?

СПИСОК ЛІТЕРАТУРИ

1. Технології нейронних мереж і нечіткого моделювання в системах управління : підруч. для здобувачів вищої освіти спец. 151 Автоматизація та комп'ютерно-інтегровані технології / Г.І. Канюк, Б.І. Кузнецов, Т.Ю. Василець, А.Ю. Мезеря, О.О. Варфоломійєв. – Харків : Друкарня Мадрид, 2020. – 306 с.
2. Нейромережеві технології в системах управління: Підручник для вузів./ Б. І. Кузнецов, Т.Ю. Василець, Т.Б. Нікітіна, В. В. Коломиєць, О.О. Варфоломійєв; Укр. інж.-пед. акад.. - Харків: УПА, 2014. - 232 с.
3. Кирик В. В. Математичний апарат штучного інтелекту в електроенергетичних системах:підручник.– Київ: КПІ ім. Ігоря Сікорського, Вид-во «Політехніка», 2019. –224 с.
4. Штучні нейронні мережі: навчальний посібник / С. В. Ткаліченко. – Кривий Ріг, 2023. –150 с.
5. Штучні нейронні мережі: навчальний посібник для студентів вищих навчальних закладів / О.Г. Руденко, Є.В. Бодяньський. – К: Компанія СМІТ, 2006, 404 с.
6. Желдак Т.А. Нечіткі множини в системах управління та прийняття рішень: навч. посіб. / Т.А. Желдак, Л.С. Коряшкіна, С.А. Ус, за редакцією С.А. Ус ; М-во освіти і науки України, Нац. техн. ун-т «Дніпровська політехніка». – Дніпро : НТУ «ДП», 2020. – 387 с.
7. Кирик В. В. К43 Математичний апарат штучного інтелекту в електроенергетичних системах: підручник / В. В. Кирик.– Київ : КПІ ім. Ігоря Сікорського, Вид-во «Політехніка» 2019.– 224с.
8. Антоненко В. М., Мамченко С.Д., Рогушина Ю.В. Сучасні інформаційні системи і технології: управління знаннями: навчальний посібник. – Ірпінь : Національний університет ДПС України, 2016. – 212 с.
9. Глибовець М.М., Отецький О.В. Штучний інтелект. – К.: Вид. дім «КМ Академія», 2002. – 366 с.
10. Ямпольський Л. С. Системи штучного інтелекту в плануванні, моделюванні та управлінні : підруч. для студ. вищ. навч. закл. / Л. С. Ямпольський, Б. П. Ткач, О. І. Лісовиченко. - К. : ДП «Вид. дім «Персонал», 2011. - 544 с.
11. Методи та системи штучного інтелекту: Теорія і практика: Навчальний посібник / О.С. Булгаков, В.В. Зосімов, В.О. Поздєєв. – Одеса. : Олді плюс, 2020. – 356с.
12. Субботін С. О. Нейронні мережі : теорія та практика: навч. посіб. / С. О. Субботін. – Житомир : Вид. О. О. Євенок, 2020. – 184 с
13. Тимошук П.В. Штучні нейронні мережі; Навч. посібн. - Львів: Львівська політехніка, 2011. -444 с.
14. Кононюк, А. Ю. Нейронні мережі і генетичні алгоритми. - Київ: Корнійчук, 2008. - 468 с.
15. Методи та системи штучного інтелекту: Навчальний посібник–/ Уклад. : А.С. Савченко, О. О. Синельников. – К. : НАУ, 2017. – 190 с.
16. Методи та системи штучного інтелекту: Навчальний посібник / Уклад.: І.М. Удовик, Г.М. Коротенко, Л.М. Коротенко, В.О. Трусів, А.Т. Харь. – Д.: Державний ВНЗ «Національний гірничий університет», 2017. – 105 с.
17. Методи та системи штучного інтелекту: навч. посіб. / укл.Д.В. Лубко, С.В. Шаров. – Мелітополь: ФОП Однорог Т.В., 2019. – 264 с.

- 18.Паламар М. І. Комп'ютерні технології штучного інтелекту для прецизійного управління у мехатронних системах : навч. посіб. / Михайло Паламар, Михайло Стрембіцький ; Тернопіл. нац. техн. ун-т ім. Івана Пулюя. Тернопіль : Тернопіл. нац. техн. ун-т ім. Івана Пулюя, 2018. 127 с.
- 19.Литвин В.В. Інтелектуальні системи : підручник / В.В. Литвин, В.В. Пасічник, Ю.В. Яцишен. – Львів: Новий світ, 2009. – 405 с.
- 20.Троцько В.В., Методи штучного інтелекту: навчально-методичний і практичний посібник. / Троцько В.В. – Київ: Університет економіки та права «КРОК», 2020. – 86 с.
- 21.Куклін В. М. К Подання знань і операції над ними; навчальний посібник. / В. М. Куклін. Харків: ХНУ імені В. Н. Каразіна, 2019 – 164 с.
- 22.Снитюк В. Є. Прогнозування. Моделі, методи, алгоритми / В. Є. Снитюк. – Київ : Маклаут, 2008. – 364 с.
- 23.Івахів, Орест Васильович. Основи побудови систем керування з нечіткою логікою : навчальний посібник /О. Івахів, М. Наконечний.– Львів :Растр-7,2017. – 129 с.
- 24.Коротка, Лариса Іванівна. Обчислювальний інтелект : теорія нечітких множин: навчальний посібник / Коротка Л.І., Зеленцов Д.Г., Науменко Н.Ю.,Ляшенко О.А., Солодка Н.О.– Дніпро :ДВНЗ УДХТУ,2020. – 161 с.
- 25.Шушура О.М. Методологічні основи побудови інформаційних технологій для автоматизації управління складними системами на принципах нечіткої логіки : дис. ... доктора технічних наук : 05.13.06 / Шушура Олексій Миколайович. – К., 2018. – 332 с.
- 26.Beale M. H., Carson E., Demuth H. B. *Deep Learning Toolbox: MATLAB R2021a*. The MathWorks, 2021. <https://www.mathworks.com/help/deeplearning/>

ДОДАТОК А

ВАРІАНТИ ЗАВДАНЬ ДЛЯ САМОСТІЙНОЇ РОБОТИ

Варіант 1

Файл вихідних даних

```
clear;clc;echo on;

% Система управління електроприводом
% механізму підйому шахтної підйімальної установки
% Система ТП-Д, двомасова

% Вихідні дані
ktp=44;
Tm=0.005;
Tms=2*Tm;
Re=0.0335;
Te=0.06325;
CF=118;
kdc=0.032;
kdv=4.72;
In=3720;
Mn=In*CF;
Jd=194892;
Jm=43508;
Js=Jd+Jm;
w0=3;
c12=(w0^2)*(Jd*Jm)/(Jd+Jm);
b12=0;
Uz=24;

kic=Re/(2*Tm*kdc*ktp);
kpc=kic*Te;
kiv=(kdc*Js)/(32*Tm^2*CF*kdv);
kpv=8*Tm*kiv;

% Матриці
A=[-b12/Jm 1/Jm b12/Jm zeros(1,4);
-c12 0 c12 zeros(1,4);
b12/Jd -1/Jd -b12/Jd CF/Jd zeros(1,3);
0 0 -CF/(Re*Te) -1/Te 1/(Re*Te) 0 0;
0 0 (-kpv*kpc*ktp*kdv)/Tm -(kdc*kpc*ktp)/Tm -1/Tm ktp/Tm ktp*kpc/Tm;
0 0 -kdv*kpv*kic -kic*kdc 0 0 kic;
0 0 -kdv*kiv zeros(1,4)];
B=[0 -1/Jm;
zeros(3,2);
(kpv*kpc*ktp)/Tm 0;
kpv*kic 0;
kiv 0];
C=zeros(6,7);
C(1,1)=1;C(2,2)=1;C(3,3)=1;C(4,4)=1;C(5,6)=1;C(6,7)=1;
D=zeros(6,2);
```

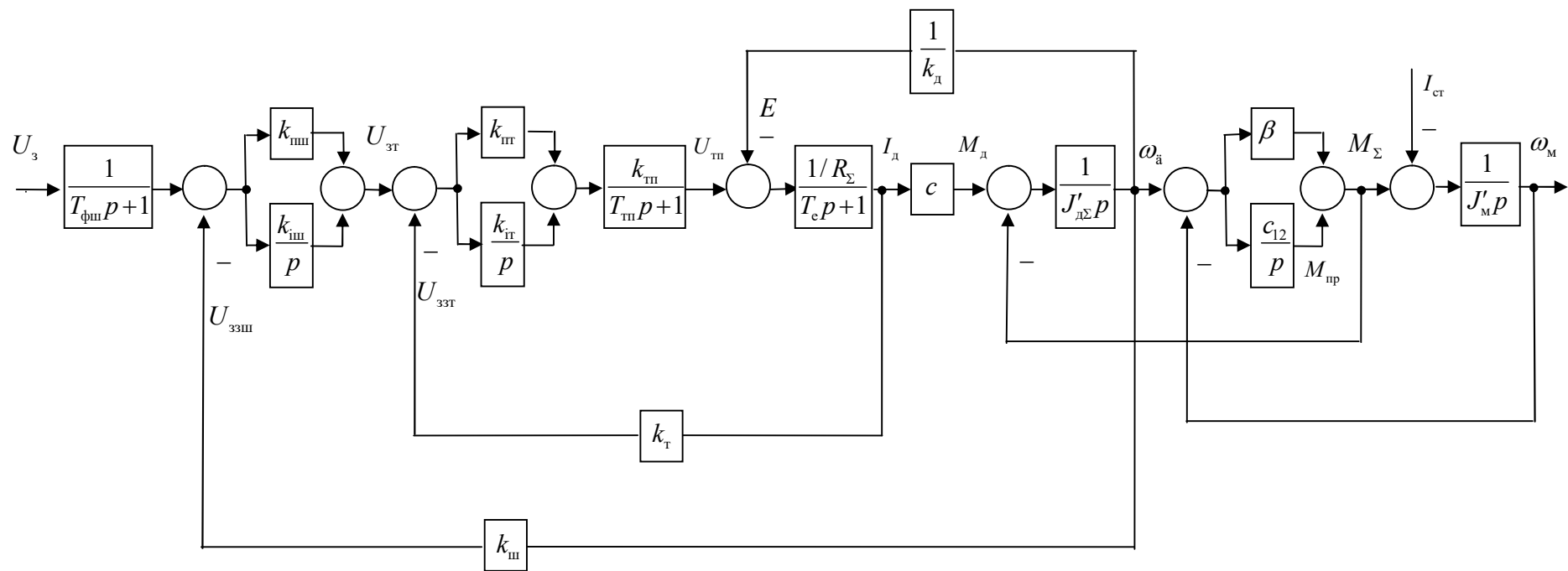


Рис. А.1. Алгоритмічна схема двомасової системи управління електроприводом ТП-Д механізму підйому шахтної підйимальної установки

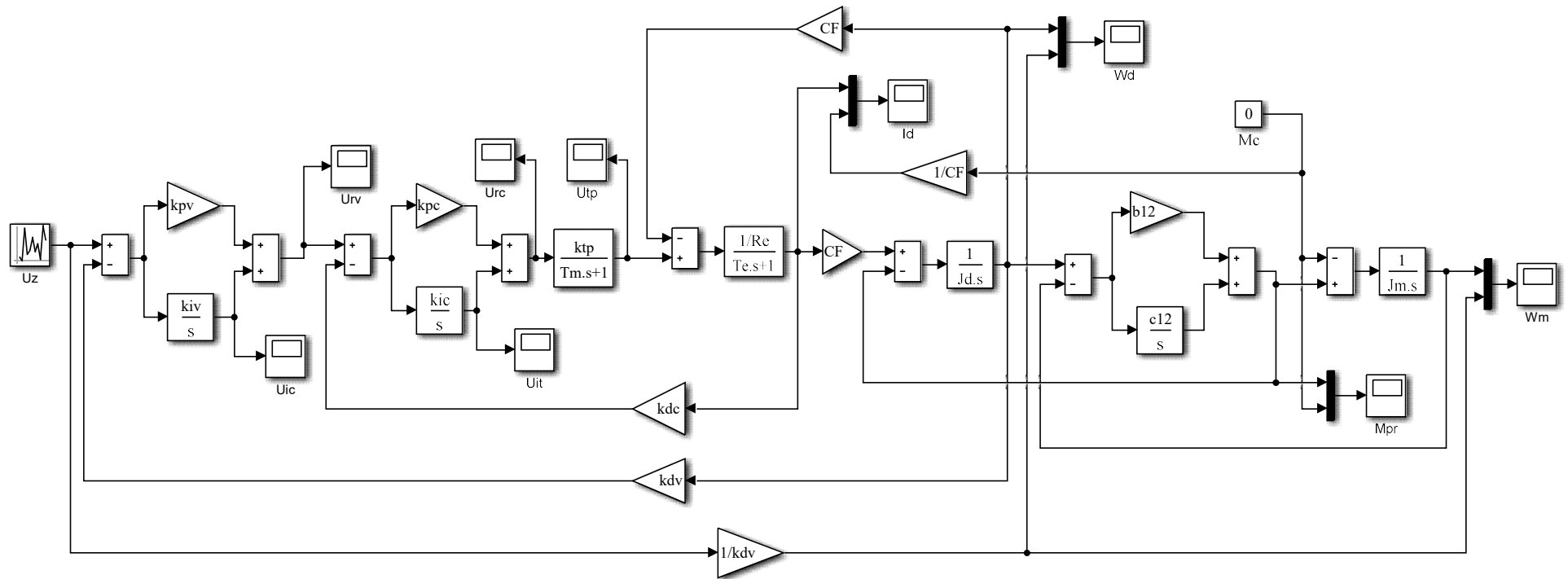


Рис. А.2. Схема моделі двомасової системи управління електроприводом ТП-Д механізму підйому шахтної підйомальної установки, розроблена в Simulink системи MATLAB

Варіант 2

Файл вихідних даних

```
clear;clc;echo on;

% Система управління електроприводом
% механізму підйому шахтної підйимальної установки
% Система ТП-Д, трьохохмасова

% Вихідні дані
ktp=66;
Tm=0.005;
Tms=2*Tm;
Re=0.0228;
Te=0.069;
CF=165;
kdc=0.0038;
kdv=6.7;
In=3180;;
Mn=In*CF;
Jd=532864;
Jk=51097;
Jm=145990;
Js=Jd+Jm+Jk;
Js=729823;
w0=2;
c=(w0^2)*((Jd+Jk/2)*(Jm+Jk/2))/((Jd+Jk/2)+(Jm+Jk/2));
c1=2*c;
c2=2*c1;
b1=0.01*c1;
b2=0.01*c2;

kic=Re/(2*Tm*kdc*ktp);
kpc=kic*Te;
kiv=(kdc*Js)/(32*Tm^2*CF*kdv);
kpv=8*Tm*kiv;
```

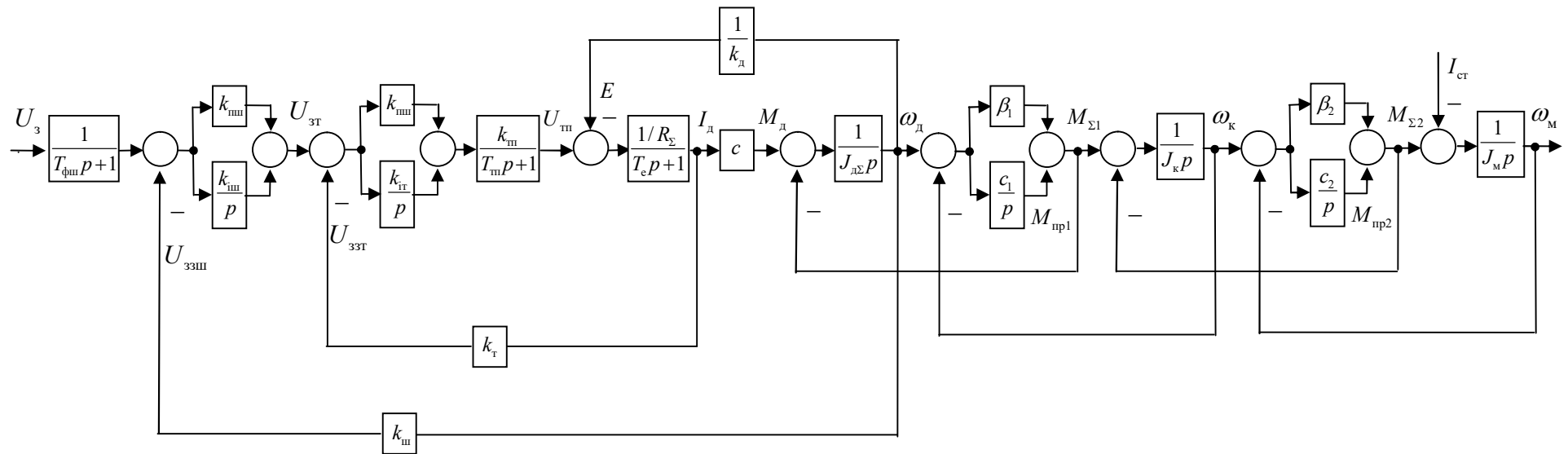


Рис. А.3. Алгоритмічна схема трьохмасової системи управління електроприводом ТП-Д механізму підйому шахтної підйимальної установки

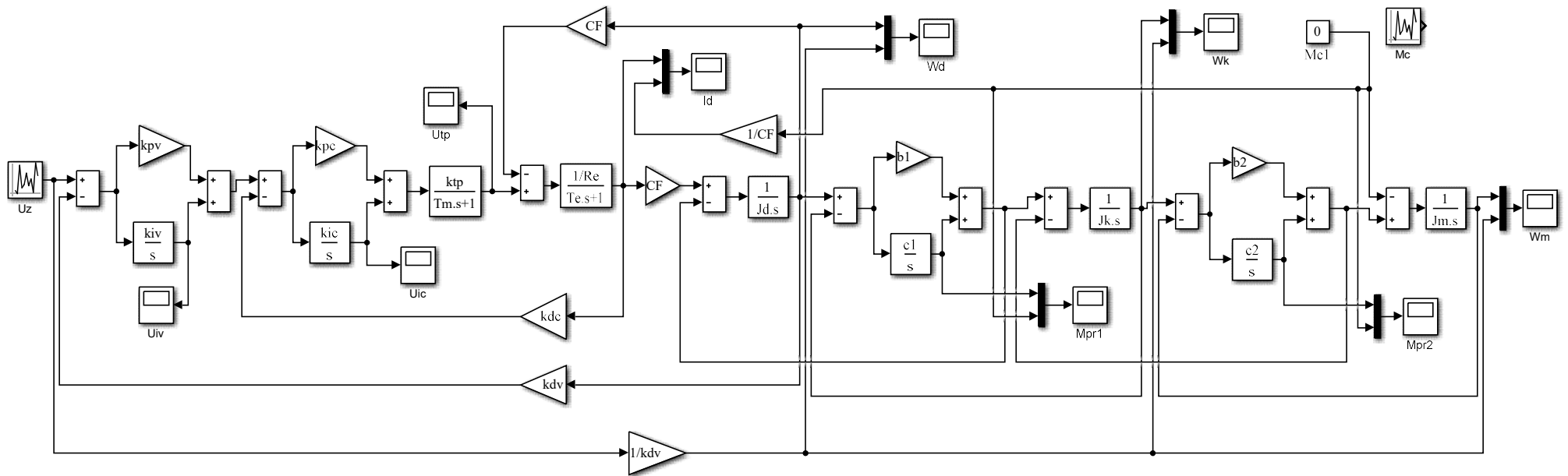


Рис. А.4. Схема моделі трьохасової системи управління електроприводом ТП-Д механізму підйому шахтної підіймальної установки, розроблена в Simulink системи MATLAB

Варіант 3

Файл вихідних даних

```
clear;clc;echo on;

% Система управління електроприводом
% механізму підйому шахтної підйімальної установки
% Система Г-Д, двомасова

% Вихідні дані
kg=3.34;
Tg=4.08;
ktp=22;
Tm=5e-3;
Tms=0.02;
Re=0.0254;
Te=0.036;
CF=170;
In=2624;
Mn=In*CF;
kdn=0.035;
kdc=3.66e-3;
kdv=5.44;
Jd=172098;
Jm=59764;
Js=Jd+Jm;
w0=1.5;
c12=(w0^2)*(Jd*Jm)/(Jd+Jm);
b12=0.005*c12;
kin=1/(2*Tm*ktp*kg*kdn);
kpn=Tg*kin;
kic=kdn*Re/(4*Tm*kdc);
kpc=Te*kic;
kiv=kdc*Js/(128*Tm^2*CF*kdv);
kpv=16*Tm*kiv;
Uz=24;
```

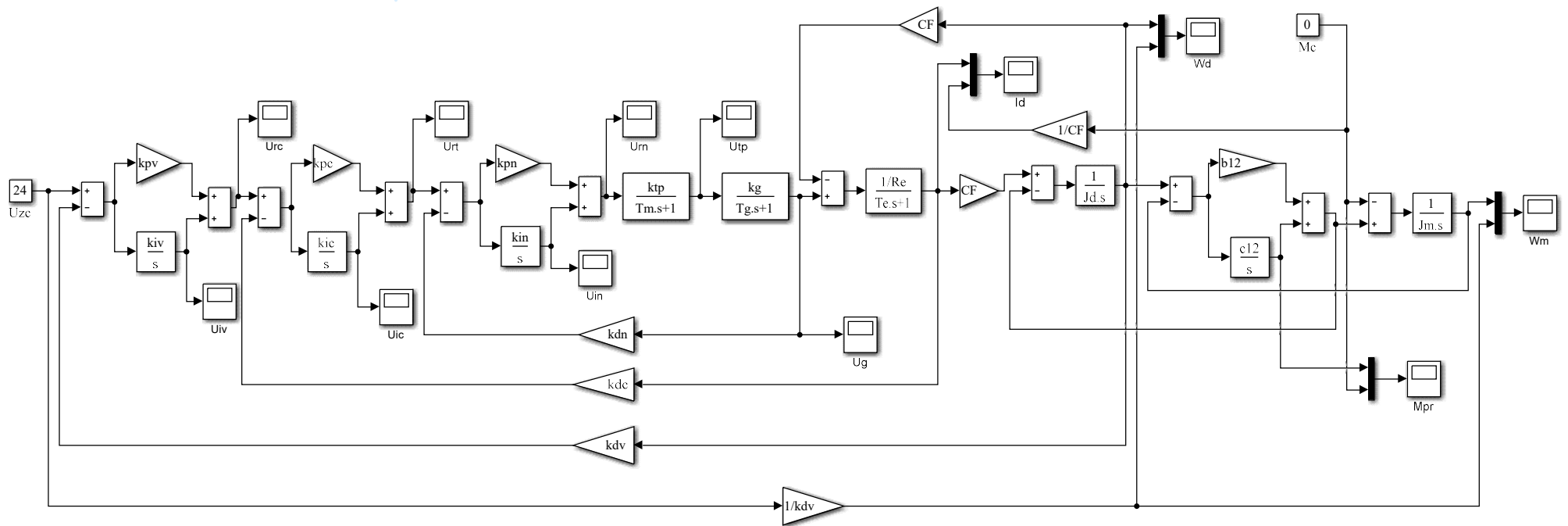



Рис. А.6. Схема моделі двоасової системи управління електроприводом Г-Д механізму підйому шахтної підйимальної установки, розроблена в Simulink ситеми MATLAB

Варіант 4

Файл вихідних даних

```
clear;clc;echo on;

% Система управління електроприводом
% механізму підйому шахтної підйимальної установки
% Система Г-Д, трьохмасова

% Вихідні дані
kg=2.845;
Tg=2.9;
ktp=22;
Tm=5e-3;
Tms=0.02;
Re=0.0269;
Te=0.056;
CF=118;
In=3720;
Mn=In*CF;
kdn=0.04;
kdt=2.58e-3;
kds=4.72;
Jd=163917;
Jk=15718;
Jm=54400;
Js=Jd+Jm+Jk;
w0=3;
c=(w0^2)*((Jd+Jk/2)*(Jm+Jk/2))/(Jd+Jm+Jk);
c1=c*2;
c2=c1;
b1=0.005*c1;
b2=0.005*c2;
kin=1/(2*Tm*ktp*kg*kdn);
kpn=Tg*kin;
kit=kdn*Re/(4*Tm*kdt);
kpt=Te*kit;
kis=kdt*Js/(128*Tm^2*CF*kds);
kps=16*Tm*kis;
```

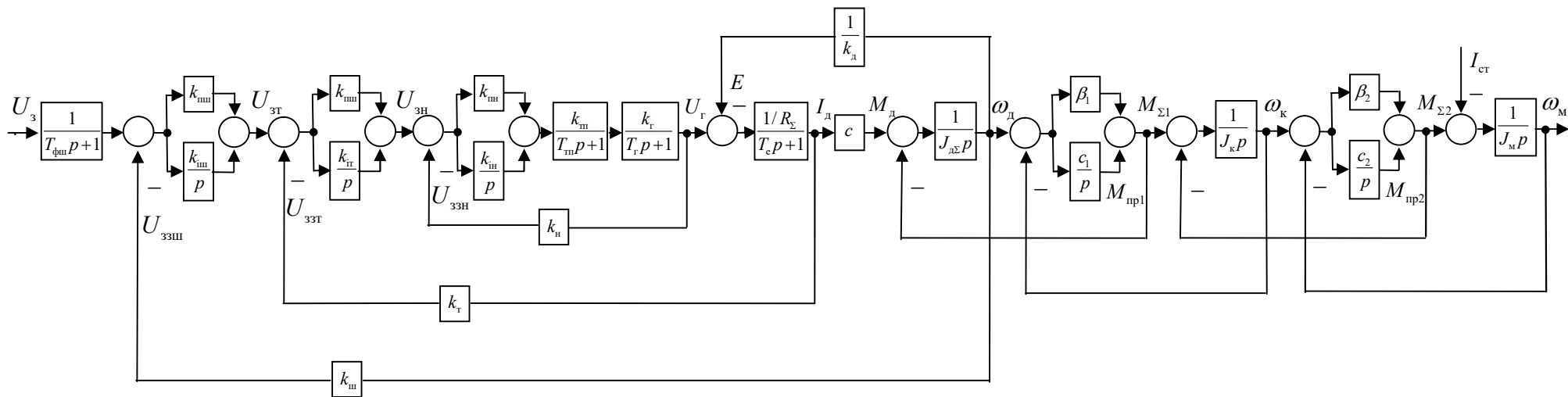


Рис. А.7. Алгоритмічна схема трьохмасової системи управління електроприводом Г-Д механізму підйому шахтної підйімальної установки

Варіант 4

Файл вихідних даних

```
% Система управління електроприводом
% повороту стріли роторного екскаватора

% Вихідні дані
kzv=1.84;
ks=4.9;
Tg=0.5;
Tzc=0.01;
Te=0.054;
Re=0.11;
CF=2*3.58;
In=338;
Mn=CF*In;
Jd=2*8.25;
Jm=41.5;
b12=0.02;
b12=0;
c12=50;
Js=Jd+Jm;
Uz=202.3;
ksx=ks / (CF+ks*kzv);
```

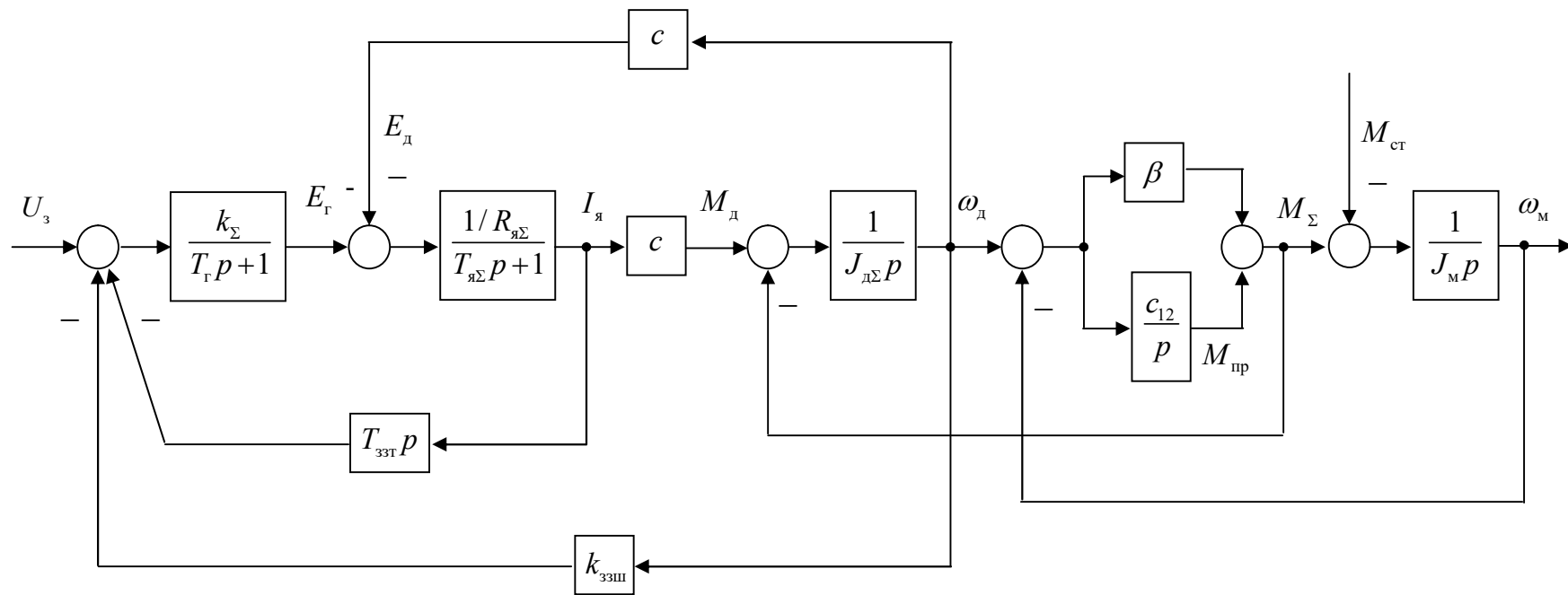


Рис. А.9. Алгоритмічна схема двомасової системи управління електроприводом Г-Д механізму повороту стріли роторного екскаватора

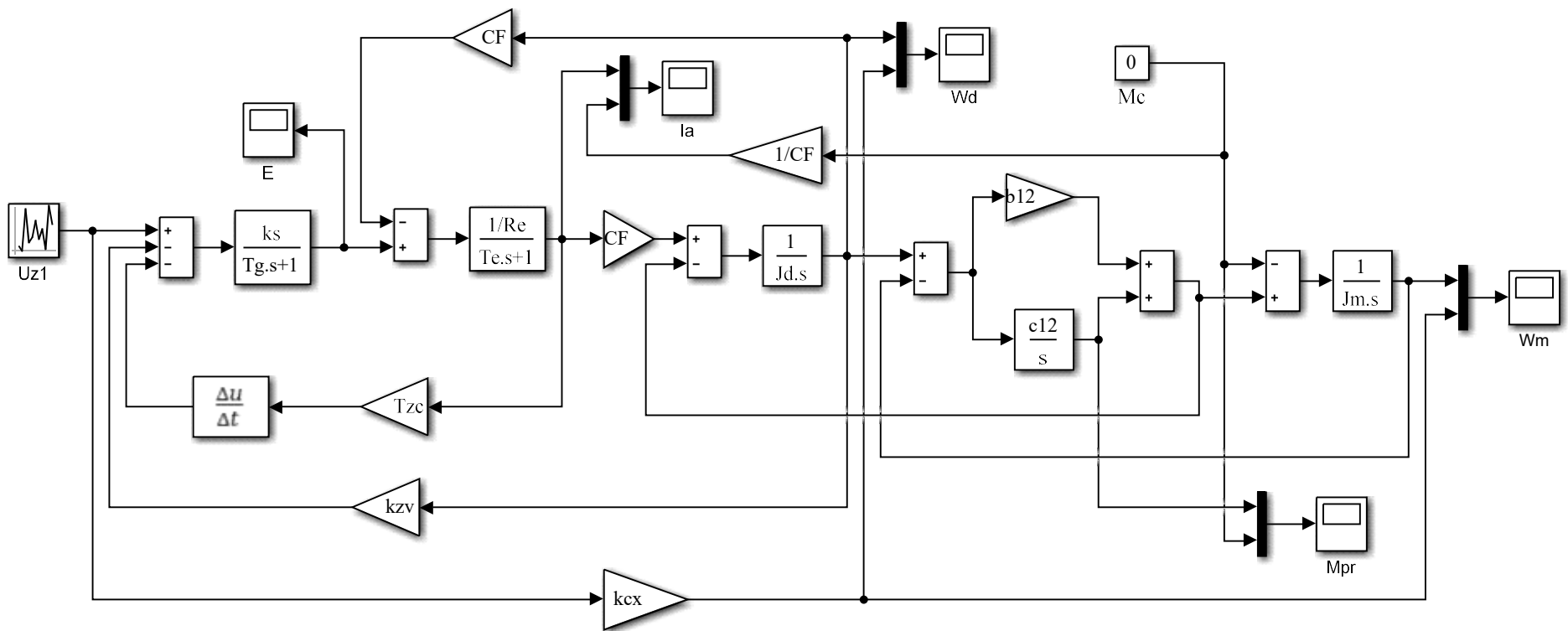


Рис. А10. Схема моделі двомасової системи управління електроприводом Г-Д механізму повороту стріли роторного екскаватора, розроблена в Simulink системи MATLAB

Варіант 5

Файл вихідних даних

```
clear;clc;echo on;

% Система управління електроприводом
% розвороту роторного екскаватора

% Вихідні дані
ks=3.43;
Re=0.307;
Rd=2*0.073;
Rcp=0;
Rs=Rd+Rcp;
Ld=2*0.003;
Lcp=0;
Ls=Ld+Lcp;
kzn=0.71;
CF=2*3.42;
Tzc=kzn*Ls;
Ttp=0.005;
Tg=Ttp;
Te=0.03;
kzv=kzn*CF;
In=192;
Mn=CF*In;
Jd=2*2;
Jm=39.4;
b12=0;
c12=25;
Js=Jd+Jm;
Uz=440;
ksx=ks/(CF*(1+ks*kzn));
k1=1.0;
k2=1.6;
k3=3.0;
```

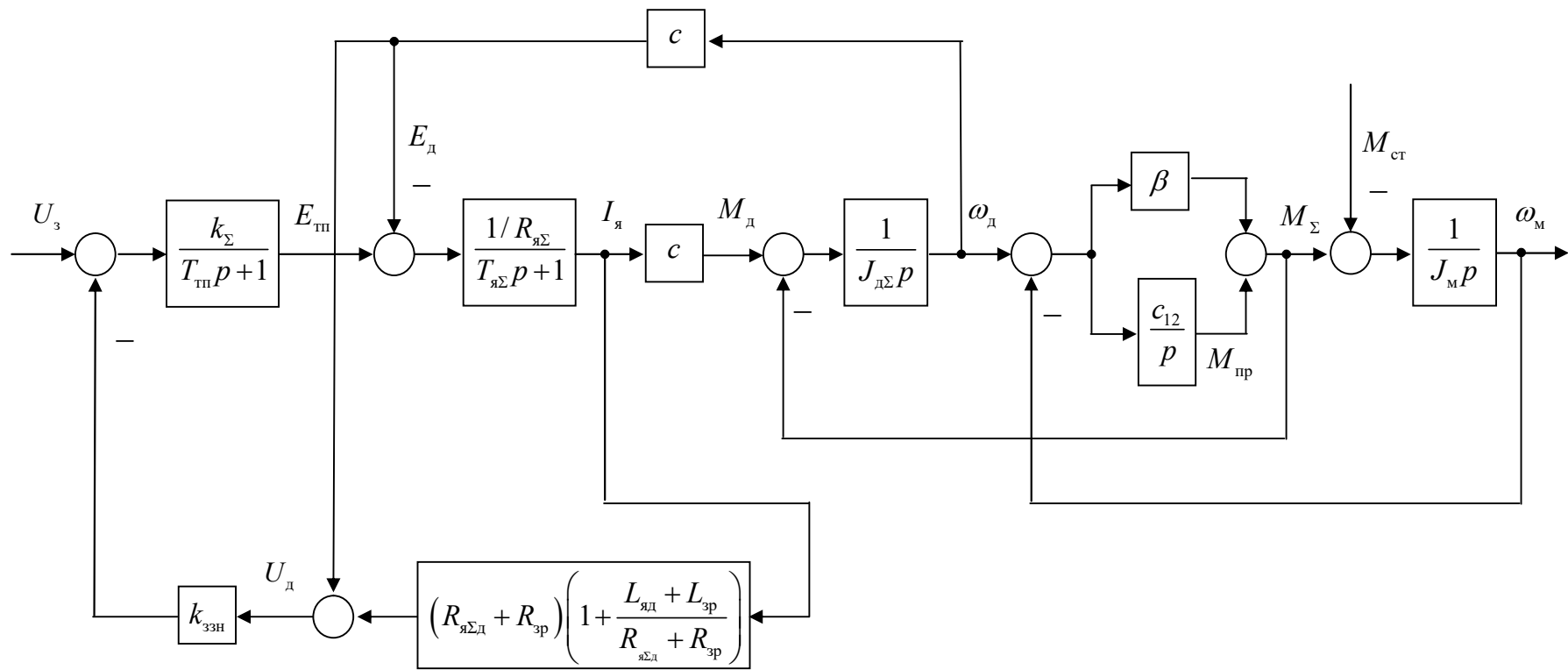


Рис. А.11. Алгоритмічна схема двомасової системи управління електроприводом ТП-Д механізму розвороту роторного екскаватора

Електронне навчальне видання комбінованого використання
Можна використовувати в локальному мережному режимі

Канюк Геннадій Іванович
Василюк Тетяна Юхимівна

ТЕХНОЛОГІЇ НЕЙРОННИХ МЕРЕЖ І НЕЧІТКОГО МОДЕЛЮВАННЯ В СИСТЕМАХ УПРАВЛІННЯ

Методичні вказівки
до проведення практичних занять для здобувачів вищої освіти
другого (магістерського) рівня за спеціальністю 174 «Автоматизація,
комп'ютерно-інтегровані технології та робототехніка»

У двох частинах

Частина 2

В авторській редакції

Підписано до розміщення 25.06.2025. Гарнітура Times New Roman.
Ум. друк. арк. 3,73. Обсяг. 4,537. Мб. Зам. 298/25.

Харківський національний університет імені В. Н. Каразіна,
61022, м. Харків, майдан Свободи, 4.
Свідоцтво суб'єкта видавничої справи ДК № 3367 від 13.01.2009
Видавництво ХНУ імені В. Н. Каразіна