

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
«ДНІПРОВСЬКА ПОЛІТЕХНІКА»



ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
Кафедра програмного забезпечення комп'ютерних систем
Кафедра інформаційних технологій та комп'ютерної інженерії

І.С. Лактіонов, В.В. Гнатушенко, В.Ю. Каштан

МЕТОДИ ТА ЗАСОБИ ПОБУДОВИ СИСТЕМ INDUSTRY 4.0

Методичні рекомендації до виконання практичних робіт
для здобувачів ступеня доктора філософії
спеціальності 123 Комп'ютерна інженерія

Дніпро
НТУ «ДП»
2024

Методи та засоби побудови систем Industry 4.0 [Електронний ресурс] : методичні рекомендації до виконання практичних робіт для здобувачів ступеня доктора філософії спеціальності 123 Комп'ютерна інженерія / уклад.: І.С. Лактіонов, В.В. Гнатушенко, В.Ю. Каштан ; М-во освіти і науки України, Нац. техн. ун-т «Дніпровська політехніка». – Дніпро : НТУ «ДП», 2024. – 80 с.

Укладачі:

І.С. Лактіонов, д-р техн. наук, доц., проф. каф. ПЗКС

В.В. Гнатушенко , д-р техн. наук, проф., зав. каф. ІТКІ

В.Ю. Каштан В.Ю., канд. техн. наук, доц., доц. каф. ІТКІ

Затверджено науково-методичною комісією спеціальності 123 Комп'ютерна інженерія (протокол № 3 від 11.03.2024) за поданням кафедри інформаційних технологій та комп'ютерної інженерії (протокол № 11 від 12.02.2024).

Уміщено загальні теоретичні відомості щодо сучасних методів і засобів побудови апаратно-програмних рішень комп'ютеризованих та кіберфізичних систем відповідно до концептуальних положень Industry 4.0; опис функціональних можливостей і особливостей використання сучасних програмних засобів для розробки, моделювання та дослідження таких систем; методики та типові приклади виконання практичних робіт; рекомендації щодо порядку виконання завдань і оформлення звітів; контрольні питання та перелік рекомендованих літературних джерел.

Відповідальний за випуск завідувач кафедри інформаційних технологій та комп'ютерної інженерії В.В. Гнатушенко, д-р техн. наук, проф.

ЗМІСТ

ВСТУП.....	4
ПРАКТИЧНА РОБОТА № 1. ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ АЛГОРИТМІВ ФУНКЦІОНУВАННЯ МІКРОКОНТРОЛЕРНИХ ПРИСТРОЇВ У СИСТЕМАХ INDUSTRY 4.0.....	6
ПРАКТИЧНА РОБОТА № 2. ДОСЛІДЖЕННЯ КОМП'ЮТЕРИЗОВАНИХ ЗАСОБІВ ЗОНАЛЬНОГО МОНІТОРИНГУ Й АДАПТИВНОГО КЕРУВАННЯ ТЕХНОЛОГІЧНИМИ ПРОЦЕСАМИ В СИСТЕМАХ INDUSTRY 4.0.....	11
ПРАКТИЧНА РОБОТА № 3. ДОСЛІДЖЕННЯ АЛГОРИТМІВ МЕРЕЖЕВОЇ ВЗАЄМОДІЇ ІНТЕЛЕКТУАЛЬНИХ ДАВАЧІВ У СКЛАДІ СИСТЕМ INDUSTRY 4.0.....	17
ПРАКТИЧНА РОБОТА № 4. ДОСЛІДЖЕННЯ ТА ОЦІНЮВАННЯ ЕФЕКТИВНОСТІ ІНТЕЛЕКТУАЛІЗОВАНИХ МЕТОДІВ ДЕТЕКТУВАННЯ РУХОМИХ ОБ'ЄКТІВ.....	22
ПРАКТИЧНА РОБОТА № 5. ДОСЛІДЖЕННЯ МЕТОДІВ ТА АЛГОРИТМІВ ОЦІНКИ ФУНКЦІОНАЛЬНО-МЕТРОЛОГІЧНИХ ХАРАКТЕРИСТИК СИСТЕМ INDUSTRY 4.0.....	27
ПРАКТИЧНА РОБОТА № 6. РОЗРОБКА ТА ДОСЛІДЖЕННЯ МОДЕЛЕЙ АПРОКСИМАЦІЇ ЧАСОВИХ РЯДІВ РЕЗУЛЬТАТІВ КОМП'ЮТЕРИЗОВАНОГО МОНІТОРИНГУ ПАРАМЕТРІВ ТЕХНОЛОГІЧНИХ ПРОЦЕСІВ.....	46
ПРАКТИЧНА РОБОТА № 7. РОЗРОБКА ТА ДОСЛІДЖЕННЯ АЛГОРИТМІВ НЕЧІТКОЇ ТРАНСФОРМАЦІЇ ДАНИХ У КОМП'ЮТЕРИЗОВАНИХ СИСТЕМАХ МОНІТОРИНГУ ТА КЕРУВАННЯ ТЕХНОЛОГІЧНИМИ ПРОЦЕСАМИ.....	61
ПРАКТИЧНА РОБОТА № 8. РОЗРОБКА ТА ДОСЛІДЖЕННЯ ГІБРИДНИХ АЛГОРИТМІВ НЕЙРОНЕЧІТКОЇ ТРАНСФОРМАЦІЇ ДАНИХ У КОМП'ЮТЕРИЗОВАНИХ СИСТЕМАХ МОНІТОРИНГУ ТА КЕРУВАННЯ ТЕХНОЛОГІЧНИМИ ПРОЦЕСАМИ.....	71
КРИТЕРІЇ ОЦІНЮВАННЯ ПРАКТИЧНИХ РОБІТ.....	78

ВСТУП

Практичні роботи з дисципліни «Методи та засоби побудови систем Industry 4.0» є обов'язковою складовою підготовки здобувачів третього (освітньо-наукового) рівня вищої освіти за ОНП «Комп'ютерна інженерія». Їх виконання спрямоване на закріплення теоретичних положень дисципліни та формування практичних навичок проєктування, моделювання, апаратно-програмної реалізації, тестування й дослідження сучасних комп'ютеризованих та кіберфізичних систем, що функціонують відповідно до концептуальних засад Industry 4.0.

Сучасні системи Industry 4.0 характеризуються широким використанням мікроконтролерних пристроїв, інтелектуальних давачів, бездротових сенсорних мереж, технологічних рішень на основі Інтернету речей, комп'ютеризованих систем моніторингу та керування, а також методів інтелектуального опрацювання даних. Поєднання апаратних, програмних та інфокомунікаційних компонент забезпечує можливість отримання даних і інформації щодо стану технологічних процесів, їх передавання та оброблення, виявлення змін параметрів досліджуваних об'єктів і формування керуючих впливів. Саме тому практичне опанування методів побудови та дослідження таких систем є необхідною складовою науково-прикладної підготовки фахівців у галузі комп'ютерної інженерії.

Особливістю практикуму є поєднання апаратно-програмного моделювання та дослідження алгоритмів функціонування систем Industry 4.0. У практичних роботах передбачено створення та тестування моделей мікроконтролерних систем моніторингу й керування, дослідження засобів зонального моніторингу та адаптивного керування, а також моделювання мережевої взаємодії інтелектуальних давачів і бездротових сенсорних вузлів.

Окремий напрям практикуму становлять методи статистичного опрацювання вимірювальної інформації. Здобувачі досліджують методи оцінювання функціонально-метрологічних характеристик інформаційно-

вимірювальних компонент систем Industry 4.0, виконують опрацювання результатів багаторазових вимірювань, оцінюють невизначеність результатів та досліджують вплив дестабілізуючих факторів. Це забезпечує формування навичок кількісного оцінювання достовірності отриманої інформації, що є важливим під час побудови надійних систем комп'ютеризованого моніторингу й керування.

Наступний рівень досліджень пов'язаний із застосуванням інтелектуальних методів обробки даних. У практичних роботах передбачено розроблення та дослідження моделей апроксимації часових рядів результатів комп'ютеризованого моніторингу параметрів технологічних процесів, а також алгоритмів нечіткої та нейронечіткої трансформації даних. Такий підхід дає змогу досліджувати можливості інтелектуалізації комп'ютеризованих систем, зокрема під час роботи з нелінійними залежностями, невизначеністю та складними взаємозв'язками між параметрами досліджуваних процесів.

Практичні роботи мають дослідницьку спрямованість. Під час їх виконання здобувачі не лише реалізують задані структурні та алгоритмічні рішення, а й проводять комп'ютерні експерименти, тестують базові та модифіковані версії моделей, здійснюють валідацію отриманих результатів, будують графічні залежності та виконують їх якісний і кількісний аналіз. У звітах передбачається представлення результатів моделювання, програмних або скриптових реалізацій, графічної візуалізації та критичного аналізу отриманих результатів із формулюванням рекомендацій щодо вдосконалення досліджуваних систем.

Виконання практикуму сприятиме формуванню у здобувачів компетентностей, необхідних для самостійного проєктування та експериментального дослідження комп'ютеризованих і кіберфізичних систем, розроблення алгоритмів моніторингу, керування та мережевої взаємодії, оцінювання достовірності вимірювальної інформації, а також застосування методів інтелектуального опрацювання даних у сучасних системах Industry 4.0.

ПРАКТИЧНА РОБОТА № 1
ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ АЛГОРИТМІВ
ФУНКЦІОНУВАННЯ МІКРОКОНТРОЛЕРНИХ ПРИСТРОЇВ У
СИСТЕМАХ INDUSTRY 4.0

Мета роботи: закріпити теоретичні знання та розвинути практичні навички з розробки, тестування й аналізу алгоритмів функціонування мікроконтролерних пристроїв моніторингу й керування технологічними процесами у складі комп'ютеризованих комплексів Industry 4.0.

Обладнання та програмне забезпечення: персональний комп'ютер, безкоштовний онлайн-сервіс Tinkercad (доступний за посиланням <https://www.tinkercad.com/dashboard>), MS Office; опціонально: інструментальні засоби, контрольно-вимірювальні прилади, інфокомунікаційні програмно-технічні пристрої, засоби автоматизації контролю і моніторингу, мікроконтролерна та мікропроцесорна техніка.

Завдання до виконання: розробити та дослідити різні за функціональним призначенням сценарії роботи мікроконтролерних пристроїв моніторингу й керування режимами крапельного зрошення ґрунту методами комп'ютерного моделювання в онлайн-середовищі розробки Tinkercad.

Хід виконання роботи:

1. Зареєструватися у безкоштовному онлайн-сервісі Tinkercad (доступний за посиланням <https://www.tinkercad.com/dashboard>) та створити новий проєкт (New Design – Circuit).

2. Розробити базову версію апаратного забезпечення комп'ютерної моделі мікроконтролерної системи моніторингу та керування крапельним зрошенням ґрунту, до складу якої входять такі компоненти: мікроконтролерна платформа Arduino Uno, плата для макетування, аналоговий давач вологості ґрунту, двигун

постійного струму та рідкокристалічний дисплей, а також дискретні аналогові компоненти. Типовий вигляд досліджуваної комп'ютерної моделі наведено на рисунку 1.1.

3. Розробити вбудоване програмне забезпечення досліджуваного мікроконтролерного пристрою моніторингу й керування зрошенням ґрунтів. Алгоритм роботи базової версії пристрою має реалізовувати наступне: онлайн детектування відносної вологості ґрунту в діапазоні від 0 % до 100 %; керування потужністю поливного насоса в трьох режимах: імітується двигуном постійного струму на основі принципу широтно-імпульсної модуляції сигналів: вимкнено (0 В), середня потужність (2,5 В) і максимальна потужність (5 В) в залежності від результатів вимірювання відносної вологості; візуалізація результатів вимірювання вологості та напруги на двигуні за допомогою рідкокристалічного дисплею. Типове вбудоване програмне забезпечення досліджуваного мікроконтролерного пристрою, яке реалізує вищенаведені обов'язкові функції, наведено на рисунку 1.2.

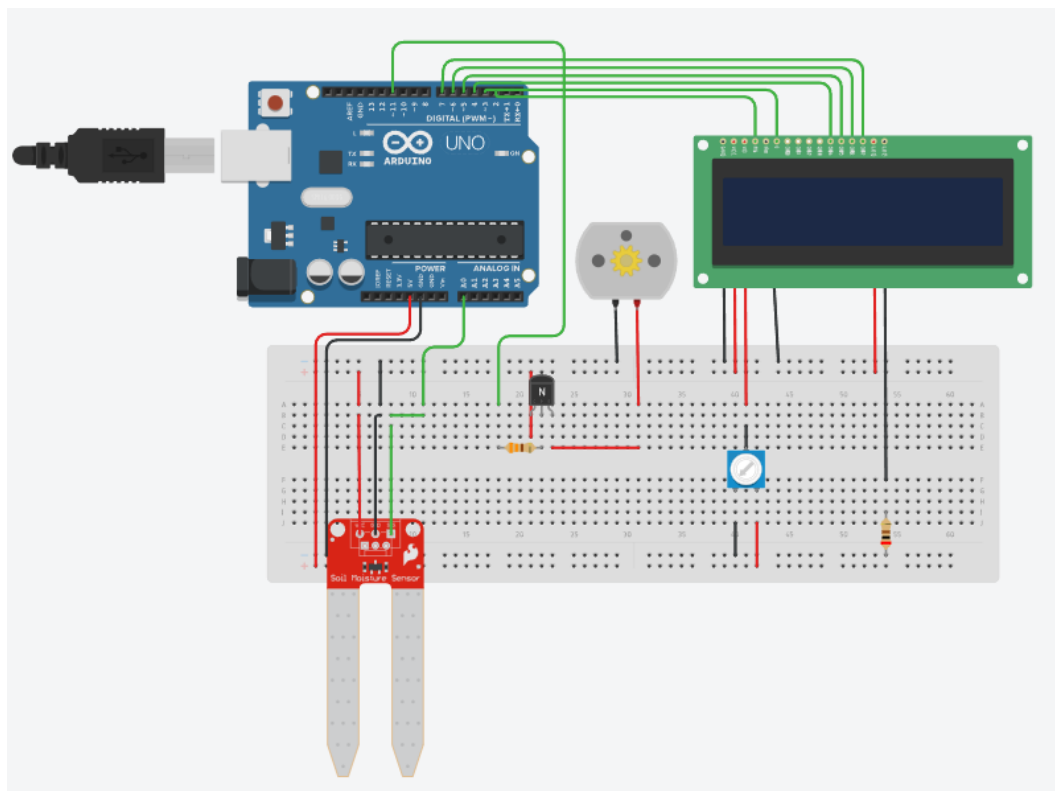


Рисунок 1.1 – Базова версія апаратного забезпечення комп'ютерної моделі досліджуваної мікроконтролерної системи

```

1  #include <LiquidCrystal.h>
2
3  const int Hum_Sensor = A0;
4  const int DC_Motor = 11;
5
6  LiquidCrystal lcd(2, 3, 4, 5, 6, 7);
7
8  void setup() {
9      Serial.begin(9600);
10     lcd.begin(16, 2);
11     lcd.print("Lets get started!");
12     pinMode(DC_Motor, OUTPUT);
13     delay(2000);
14     lcd.clear();
15     lcd.setCursor(0,0);
16     lcd.print("Humidity= ");
17     lcd.setCursor(0,1);
18     lcd.print("Motor volts= ");
19 }
20
21 void loop() {
22     int Kod_ADC = analogRead(Hum_Sensor);
23     float Humidity = map(Kod_ADC,0,1023,0,100);
24     lcd.setCursor(9,0);
25     lcd.print(Humidity);
26     lcd.print(" %");
27     lcd.setCursor(12,1);
28
29     if (Humidity < 50)
30     {
31         analogWrite(DC_Motor, 255);
32         lcd.print("5 V");
33     }
34
35     if ((Humidity >= 50) and (Humidity < 75))
36     {
37         analogWrite(DC_Motor, 127);
38         lcd.print("2.5 V");
39     }
40
41     if (Humidity >= 75)
42     {
43         analogWrite(DC_Motor, 0);
44         lcd.print("0 V");
45     }
46     delay(1000);
47 }

```

Рисунок 1.2 – Базова версія вбудованого програмного забезпечення комп'ютерної моделі досліджуваного мікроконтролерного пристрою

4. Протестувати розроблену комп'ютерну модель (Start Simulation) і оцінити коректність її роботи. Типовий вигляд результатів роботи досліджуваної моделі наведено на рисунку 1.3.

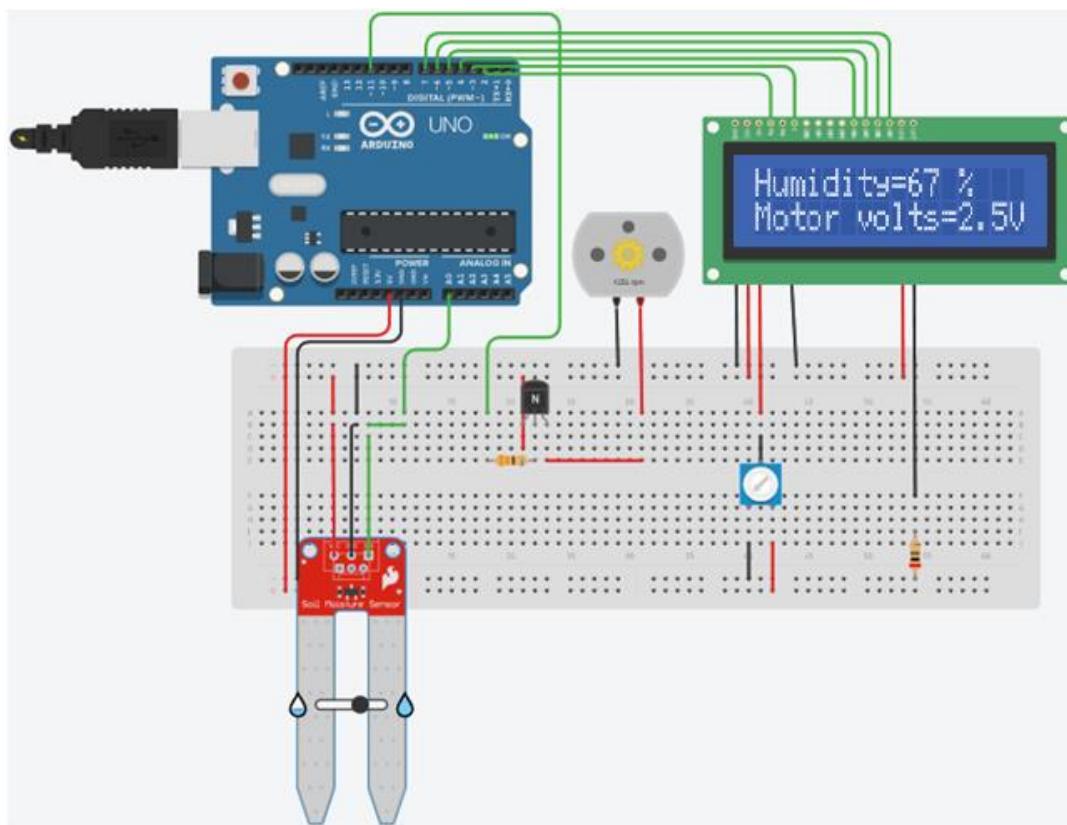


Рисунок 1.3 – Результат тестування базової версії комп’ютерної моделі досліджуваного мікроконтролерного пристрою

5. Модифікувати базову версію досліджуваного мікроконтролерного пристрою шляхом обліку впливу температури ґрунту на режими керування поливним насосом (імітується двигуном постійного струму). Модифікована система повинна вимірювати температуру (рекомендовано використовувати датчик Temperature Sensor) у діапазоні від 10 °C до 50 °C. Результати вимірювань температури (Т) та вологості (Н) повинні бути розподілені на три якісні діапазони: High, Medium та Low (числові межі обираються довільно). Режим керування поливним насосом обираються відповідно до таблиці 1.1. Результати вимірювання температури і вологості мають бути візуалізовані за допомогою рідкокристалічного дисплея. Також поточна напруга на двигуні, температура і вологість ґрунту мають відображатися за допомогою монітора послідовного порту. Додатково в системі має бути передбачена світлодіодна індикація (LED): червоний світлодіод (напруга на двигуні – 5 В); зелений світлодіод (напруга на двигуні – 2,5 В); синій світлодіод (напруга на двигуні – 0 В).

Таблиця 1.1 – Режими керування поливним насосом

T H	LOW	MEDIUM	HIGH
LOW	5 B	2,5 B	0 B
MEDIUM	5 B	2,5 B	0 B
HIGH	5 B	5 B	0 B

6. Протестувати модифіковану версію комп'ютерної моделі досліджуваного мікроконтролерного пристрою моніторингу й керування поливом ґрунту та валідувати коректність її роботи.

7. Побудувати графічні залежності напруги на двигуні постійного струму, що імітує роботу поливного насосу, в залежності від температури й вологості ґрунту. Графічні залежності мають бути побудовані за результатами тестування модифікованої версії комп'ютерної моделі досліджуваного мікроконтролерного пристрою (на основі п. 5 і 6 практичної роботи).

8. *Опціонально:* перевірити працездатність розробленого апаратно-програмного рішення в лабораторних умовах.

Типовий зміст звіту до практичної роботи:

1. Титульний лист.
2. Мета та задачі практичної роботи.
3. Основні результати, що були отримані під час створення та тестування базової та модифікованої версій комп'ютерної моделі мікроконтролерного пристрою моніторингу та керування режимами крапельного зрошення ґрунту за відповідними пунктами досліджень (скріншоти апаратної та програмних компонент, коментарі до коду програми, текстовий опис і графічна візуалізація основних результатів досліджень).
4. Загальні висновки по роботі, що містять результати критичного аналізу на якісному й кількісному рівнях, а також рекомендації з оптимізації та сфер практичного використання досліджуваного пристрою.

Контрольні питання:

1. Які типові компоненти мікроконтролерних систем моніторингу й керування технологічними процесами були використані та протестовані в цій практичній роботі?

2. Дайте визначення термінам «давач» та «мікроконтролер». Яке їх функціональне призначення в складі комп'ютеризованих комплексів моніторингу й керування технологічними процесами?

3. Розкрийте фізичні засади принципу широтно-імпульсної модуляції сигналів під час керування електричним навантаженням.

4. Які функції програмування були використані для моніторингу вхідних параметрів системи та керування вихідною величиною системи?

5. Які функції роботи з рідкористалічним дисплеєм були використані в цій практичній роботі? Яке їх призначення та синтаксис?

Перелік рекомендованих джерел інформації:

1. Вовна О.В., Лактіонов І.С., Лебедєв В.А. Комп'ютерно-інтегрований моніторинг та керування в промислових теплицях: поточні результати і перспективи досліджень: монографія. Покровськ: ДонНТУ, 2020. 255 с.

2. Pascual D.G., Daponte P., Kumar U. The Industry 4.0 Architecture and Cyber-Physical Systems from: Handbook of Industry 4.0 and SMART Systems CRC. London: CRC Press, 2022. 41 p.

3. Лактіонов І.С. Інформаційно-вимірювальне забезпечення та апаратно-програмні засоби побудови комп'ютеризованих систем моніторингу стану мікроклімату теплиць: дис. ... д-р. техн. наук: 05.13.05 / ДВНЗ «Донецький національний технічний університет»: Д 11.052.03. Покровськ, 2021. 518 с.

4. Kharchenko V.S. Internet of Things for Industry and Human Application. In Volumes 1–3. Vol. 2. Modelling and Development. Kyiv: Ministry of Education and Science of Ukraine, National Aerospace University KhAI, 2019. 547 p.

5. Laktionov I.S. et al. Results of simulation and physical modeling of the computerized monitoring and control system for greenhouse microclimate parameters. Int. J. on Smart Sensing and Intell. Systems. 2018. Vol. 11 (1). P. 1–15.

ПРАКТИЧНА РОБОТА № 2

ДОСЛІДЖЕННЯ КОМП'ЮТЕРИЗОВАНИХ ЗАСОБІВ ЗОНАЛЬНОГО МОНІТОРИНГУ Й АДАПТИВНОГО КЕРУВАННЯ ТЕХНОЛОГІЧНИМИ ПРОЦЕСАМИ В СИСТЕМАХ INDUSTRY 4.0

Мета роботи: закріпити теоретичні знання і розвинути практичні навички з методів розробки та дослідження архітектурних рішень комп'ютеризованих і кіберфізичних систем Industry 4.0 під час зонального моніторингу й адаптивного керування технологічними процесами.

Обладнання та програмне забезпечення: персональний комп'ютер, безкоштовний онлайн-сервіс Tinkercad (доступний за посиланням <https://www.tinkercad.com/dashboard>), MS Office; опціонально: інструментальні засоби, контрольно-вимірювальні прилади, інфокомунікаційні програмно-технічні пристрої, засоби автоматизації контролю і моніторингу, мікроконтролерна та мікропроцесорна техніка.

Завдання до виконання: розробити та дослідити різні за функціональним призначенням адаптивні сценарії роботи мікроконтролерних систем зонального моніторингу й керування інфокомунікаційними процесами Smart Home методами комп'ютерного моделювання в онлайн-середовищі Tinkercad.

Хід виконання роботи:

1. Запустити онлайн-сервіс Tinkercad (за посиланням <https://www.tinkercad.com/dashboard>) та створити новий проєкт (New Design – Circuit).

2. Розробити базову версію апаратного забезпечення комп'ютерної моделі мікроконтролерної системи адаптивного моніторингу та керування процесами, що протікають у інфокомунікаційному середовищі досліджуваного об'єкта

Smart Home. До складу базової версії системи мають входити такі апаратні компоненти: мікроконтролерна платформа Arduino Uno, плата для макетування, давач концентрації газу, давач температури, давач інтенсивності освітлення, резистори номіналом 4,7 кОм (2 шт.) і 200 Ом (1 шт.), аналоговий потенціометр та рідкокристалічний дисплей. Типовий вигляд базової версії досліджуваної комп'ютерної моделі наведено на рисунку 2.1.

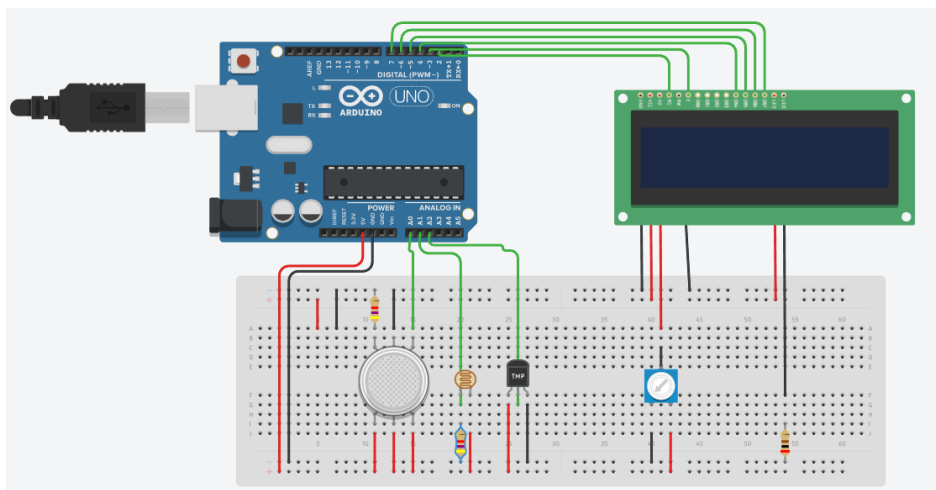


Рисунок 2.1 – Базова версія апаратного забезпечення комп'ютерної моделі досліджуваної мікроконтролерної системи

3. Розробити вбудоване програмне забезпечення базової версії досліджуваної мікроконтролерної системи зонального моніторингу й адаптивного керування інфокомунікаційними процесами Smart Home. Типовий алгоритм роботи базової версії комп'ютерної моделі системи має реалізовувати наступне: онлайн детектування температури, інтенсивності освітлення й концентрації газу; візуалізація результатів вимірювання за допомогою рідкористалічного дисплею. Типовий скетч до вбудованого програмного забезпечення базової версії досліджуваної мікроконтролерної системи, яке реалізує вищенаведені обов'язкові функції, наведено на рисунку 2.2.

4. Протестувати розроблену в п. 3 практичної роботи комп'ютерну модель (Start Simulation) і оцінити коректність її роботи. Типовий вигляд результатів роботи досліджуваної комп'ютерної моделі базової версії

мікроконтролерної системи наведено на рисунку 2.3.

```
1 #include <LiquidCrystal.h>
2
3 const int Gas_Sensor = A0;
4 const int Light_Sensor = A1;
5 const int Temp_Sensor = A2;
6
7 LiquidCrystal lcd(2, 3, 4, 5, 6, 7);
8
9 void setup() {
10   Serial.begin(9600);
11   lcd.begin(16, 2);
12   lcd.print("Lets get started!");
13   delay(2000);
14   lcd.clear();
15   lcd.setCursor(0,0);
16   lcd.print("G= ");
17   lcd.setCursor(9,0);
18   lcd.print("T= ");
19   lcd.setCursor(0,1);
20   lcd.print("L= ");
21 }
22
23 void loop() {
24   int Kod_ADC_Gas = analogRead(Gas_Sensor);
25   float Gas = map(Kod_ADC_Gas,0,1023,0,500);
26   lcd.setCursor(2,0);
27   lcd.print(Gas,0);
28   lcd.print(" un");
29   lcd.setCursor(11,0);
30
31   int Kod_ADC_Temp = analogRead(Temp_Sensor);
32   float Temp = Kod_ADC_Temp*150.0/1023;
33   lcd.print(Temp,0);
34   lcd.print(" C");
35   lcd.setCursor(2,1);
36
37   int Kod_ADC_Light = analogRead(Light_Sensor);
38   float Light = map(Kod_ADC_Light,0,1023,0,10000);
39   lcd.print(Light,0);
40   lcd.print(" lx");
41
42   delay(1000);
43 }
```

Рисунок 2.2 – Базова версія вбудованого програмного забезпечення моделі

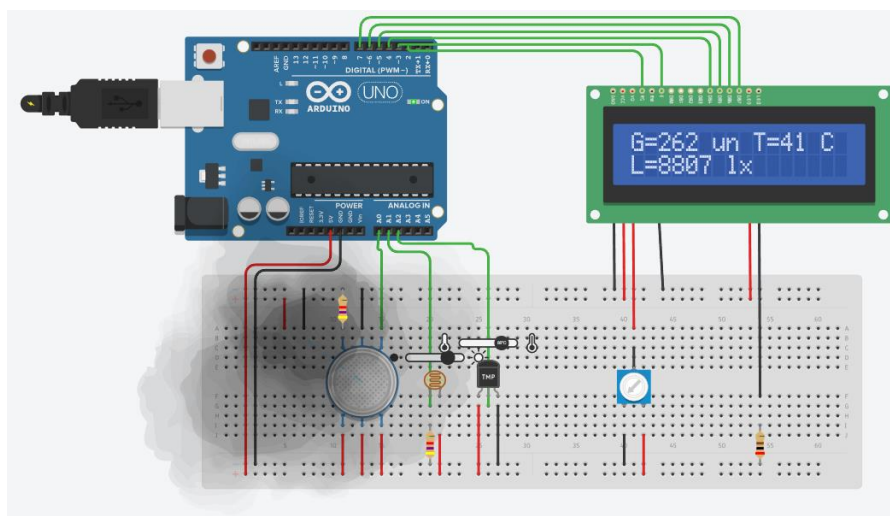


Рисунок 2.3 – Результати тестування базової версії досліджуваної комп'ютерної моделі

5. Модифікувати базову версію досліджуваної системи наступним чином:

5.1. Реалізувати зональний контроль детектованих фізичних параметрів Smart Home: кількість датчиків температури, концентрації газу та інтенсивності освітлення – по 2 шт.; виконувати розрахунки і візуалізацію середніх, максимальних і мінімальних значень детектованих фізичних параметрів за допомогою монітору послідовного порту (середні значення вимірюваних величин мають додатково відображатись на рідкристалічному дисплеї).

5.2. Додати до моделі три світлодіоди та реалізувати наступний алгоритм. Засвітити червоний світлодіод, якщо середнє значення будь-якої вимірюваної фізичної величини менше на 25 % від максимального значення відповідних результатів вимірювань. Засвітити синій світлодіод, якщо середнє значення будь-якої вимірюваної фізичної величини перевищує мінімальне значення відповідних результатів вимірювань на 25 %. Засвітити зелений світлодіод, якщо жодна з перелічених вище умов не виконується.

6. Протестувати модифіковану версію комп'ютерної моделі досліджуваної мікроконтролерної системи зонального моніторингу й адаптивного керування інфокомунікаційними процесами Smart Home та валідувати коректність її роботи.

7. Побудувати графічні залежності станів відповідних світлодіодів (Low / High) залежно від результатів детектування фізичних параметрів мікроклімату (температури повітря, інтенсивності освітлення та концентрації газу).

8. *Опціонально:* перевірити працездатність розробленого апаратно-програмного рішення в лабораторних умовах.

Типовий зміст звіту до практичної роботи:

1. Титульний лист.

2. Мета та задачі практичної роботи.

3. Основні результати, що були отримані під час створення та тестування базової та модифікованої версій комп'ютерної моделі мікроконтролерної системи зонального моніторингу та адаптивного керування процесами Smart

Номе за відповідними пунктами досліджень (скріншоти апаратної та програмних компонент, коментарі до коду програми, текстовий опис і графічна візуалізація основних результатів досліджень).

4. Загальні висновки по роботі, що містять результати критичного аналізу на якісному й кількісному рівнях, а також рекомендації з оптимізації та практичного використання досліджуваної системи.

Контрольні питання:

1. Які типові компоненти мікроконтролерних систем моніторингу й керування інфокомунікаційними процесами Smart Home були використані та протестовані в цій практичній роботі?

2. Яким є основне функціональне призначення використаних компонент у складі комп'ютеризованих комплексів моніторингу й керування?

3. Розкрийте фізичні засади щодо зонального моніторингу та адаптивного керування?

4. Які функції програмування були використані для моніторингу вхідних параметрів системи та керування її вихідними сигналами?

5. Які функції роботи з рідкокристалічним дисплеєм та монітором послідовного порту були використані в цій практичній роботі? Яке їх призначення та синтаксис?

Перелік рекомендованих джерел інформації:

1. Laktionov I.S. et al. Results of simulation and physical modeling of the computerized monitoring and control system for greenhouse microclimate parameters. Int. Journal on Smart Sensing and Intelligent Systems. 2018. Vol. 11 (1). P. 1–15.

2. Kharchenko V.S. Internet of Things for Industry and Human Application. In Volumes 1–3. Vol. 2. Modelling and Development. Kyiv: Ministry of Education and Science of Ukraine, National Aerospace University KhAI, 2019. 547 p.

ПРАКТИЧНА РОБОТА № 3

ДОСЛІДЖЕННЯ АЛГОРИТМІВ МЕРЕЖЕВОЇ ВЗАЄМОДІЇ

ІНТЕЛЕКТУАЛЬНИХ ДАВАЧІВ У СКЛАДІ СИСТЕМ INDUSTRY 4.0

Мета роботи: закріпити теоретичні знання та розвинути практичні навички з розробки, тестування й аналізу типових алгоритмів функціонування мережеских організацій інтелектуальних давачів у складі комп'ютеризованих систем Industry 4.0.

Обладнання та програмне забезпечення: персональний комп'ютер, безкоштовний онлайн-сервіс CupCarbon (доступний за посиланням <https://cupcarbon.com/>), MS Office; опціонально: інструментальні засоби, контрольно-вимірювальні прилади, інфокомунікаційні програмно-технічні пристрої, засоби автоматизації контролю і моніторингу, мікроконтролерна та мікропроцесорна техніка.

Завдання до виконання: розробити та дослідити різні за функціональним призначенням сценарії мережескої взаємодії інтелектуальних давачів під час контролю й діагностування параметрів і характеристик технологічних процесів із урахуванням концептуальних засад Industry 4.0.

Хід виконання роботи:

1. Створити новий проєкт в середовищі CupCarbon: Project – New project.
2. Розробити та протестувати комп'ютерну модель мережі інтелектуальних давачів на предмет відпрацювання алгоритму визначення координат інтелектуальних бездротових сенсорів та базової станції, як це показано на рисунку 3.1. Під час розробки скриптів сенсорних вузлів рекомендовано протестувати коректність відпрацювання двох варіантів із використанням функцій: `getpos` та `getpos2`. Реалізувати відправку інформаційних повідомлень із координатами від одного сенсорного вузла до

базової станції. На стороні приймача (базової станції) передбачити розділення інформаційного повідомлення, що отримане від передавачів сигналів під час використання функції `getpos`, на окремі змінні і вивести отриманий результат на індикацію у вікні консолі. Типові скрипти щодо відпрацювання алгоритмів визначення координат функціональних компонент моделі з використанням функцій `getpos` і `getpos2`, а також відправлення й приймання інформаційних повідомлень, наведено на рисунку 3.2.

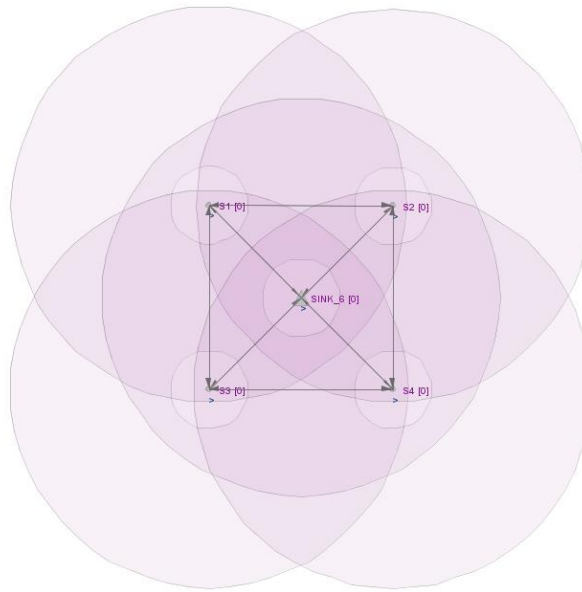


Рисунок 3.1 – Досліджувана комп’ютерна модель інфокомунікаційної мережевої організації інтелектуальних давачів

<pre> loop getpos a print a stop a) getpos </pre>	<pre> loop getpos2 a b print a delay 2000 print b delay 2000 б) getpos2 </pre>
<pre> loop getpos a print a send a 0 1 delay 1000 в) передавач </pre>	<pre> loop wait read x rdata x m n cprint m cprint n г) приймач (базова станція) </pre>

Рисунок 3.2 – Скрипти до досліджуваної комп’ютерної моделі інфокомунікаційної мережевої організації інтелектуальних давачів

3. Реалізувати алгоритм визначення базовою станцією кількості підключених до неї інтелектуальних сенсорних вузлів. Під час розробки відповідного скрипта рекомендовано використовувати стандартну функцію `atnd`. Отриманий результат вивести на індикацію на вузлі базової станції. Типовий результат тестування за цим пунктом досліджень наведено на рисунку 3.3.

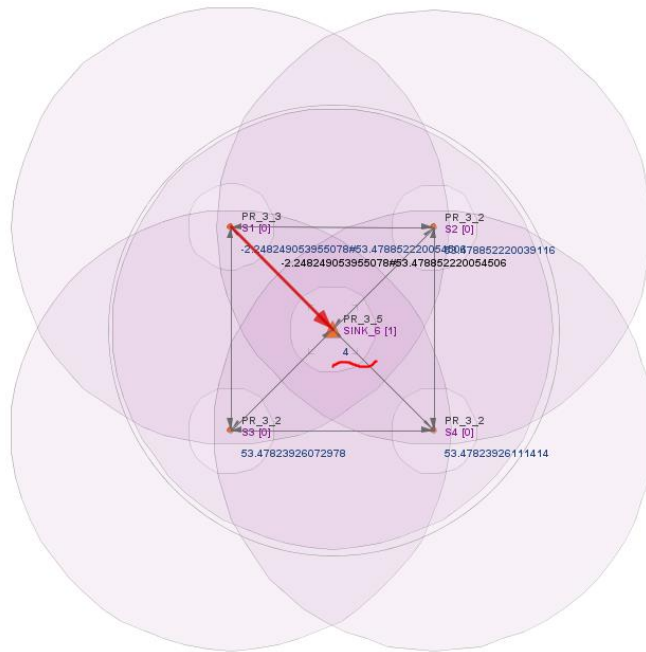


Рисунок 3.3 – Результат тестування комп’ютерної моделі за п. 3 практичної роботи

4. Модифікувати розроблену в п. 2 і 3 комп’ютерну модель наступним чином:

4.1. Додати до вузла базової станції можливість збереження отриманих від сенсорного вузла координат у файл.

4.2. Реалізувати відправлення значення кількості підключених до базової станції інтелектуальних сенсорів до бездротового сенсорного вузла (попередньо необхідно додати до моделі ще один елемент типу `Sensor node`) за технологією LoRa.

4.3. Визначити на доданому сенсорному вузлі (приймач сигналів за

технологією LoRa) його мережеві параметри: Network ID, Channel та MY address. Результати визначення цих параметрів вивести на індикацію у вікні консолі. Рекомендовано під час цього користуватись стандартною функцією atget.

5. Протестувати модифіковану версію комп'ютерної моделі досліджуваної мережевої організації бездротових інтелектуальних давачів та валідувати коректність її роботи.

6. Навести приклади можливостей і сфер прикладного застосування результатів моделювання, отриманих у цій практичній роботі, під час проектування та використання систем Industry 4.0.

7. *Опціонально:* перевірити працездатність розробленого апаратно-програмного рішення в лабораторних умовах.

Типовий зміст звіту до практичної роботи:

1. Титульний лист.

2. Мета та задачі практичної роботи.

3. Основні результати, що були отримані під час створення та тестування базової та модифікованої версій комп'ютерної моделі мережевої взаємодії інтелектуальних давачів у складі систем Industry 4.0, за відповідними пунктами досліджень (скріншоти моделей, скрипти із відповідними коментарями, текстовий опис і графічна візуалізація основних результатів досліджень валідованих моделей).

4. Загальні висновки по роботі, що містять результати критичного аналізу на якісному й кількісному рівнях, а також рекомендації з оптимізації досліджуваної системи.

Контрольні питання:

1. Які типові функціональні компоненти середовища моделювання CupCarbon були використані та протестовані в цій практичній роботі? Яким є їхнє функціональне призначення в реалізованій комп'ютерній моделі?

2. Які бездротові технології обміну інформаційними повідомленнями були застосовані під час комп'ютерного експерименту в цій практичній роботі? Якими є їхні основні характеристики?

3. Які стандартні функції CupCarbon були використані під час визначення координат функціональних компонент інфокомунікаційної мережі? Яким є їх синтаксис? У чому полягає різниця між використаними функціями?

4. Надайте характеристику типовому алгоритму визначення кількості бездротових сенсорних вузлів, підключених до базової станції мережі. Які функції CupCarbon були використані під час реалізації цього алгоритму?

5. Яким є функціональне призначення функцій wait, print та sprint під час розробки скриптів досліджуваної моделі?

6. Які основні мережеві ідентифікатори бездротових сенсорних вузлів визначалися в цій практичній роботі? Яким чином було реалізовано цю ідентифікацію?

7. У чому полягає функціональне призначення бездротових сенсорних мереж під час проєктування та використання систем Industry 4.0 в умовах промислових підприємств?

Перелік рекомендованих джерел інформації:

1. CupCarbon IoT. URL: <https://cupcarbon.com/> (дата звернення 14.11.2023).
2. Forster A. Introduction to Wireless Sensor Networks / A. Forster. New York: Wiley, 2016. 186 p.
3. Жураковський Б.Ю., Зенів І.О. Технології інтернету речей. Навчальний посібник [Ел. ресурс]: навч. посіб. для студ. спец. 126 «Інформаційні системи та технології», спеціаліз. «Інформаційне забезпечення робототехнічних систем» / Б.Ю. Жураковський, І.О. Зенів. Київ: КПІ ім. Ігоря Сікорського, 2021. 271 с.
4. Laktionov I., Diachenko G., Koval V., Yevstratiev M. Computer-Oriented Model for Network Aggregation of Measurement Data in IoT Monitoring of Soil and Climatic Parameters of Agricultural Crop Production Enterprises. Baltic J. Modern Computing, 2023, Vol. 11 (3). P. 500–522.

ПРАКТИЧНА РОБОТА № 4
ДОСЛІДЖЕННЯ ТА ОЦІНЮВАННЯ ЕФЕКТИВНОСТІ
ІНТЕЛЕКТУАЛІЗОВАНИХ МЕТОДІВ ДЕТЕКТУВАННЯ РУХОМИХ
ОБ'ЄКТІВ

Мета роботи: закріпити теоретичні знання та розвинути практичні навички з розробки, тестування й аналізу структурно-алгоритмічних організацій цифрових інтелектуалізованих систем детектування рухомих об'єктів.

Обладнання та програмне забезпечення: персональний комп'ютер, безкоштовний онлайн-сервіс CupCarbon (доступний за посиланням <https://cupcarbon.com/>), MS Office; опціонально: інструментальні засоби, контрольно-вимірювальні прилади, інфокомунікаційні програмно-технічні пристрої, засоби автоматизації контролю і моніторингу, мікроконтролерна та мікропроцесорна техніка.

Завдання до виконання: розробити та дослідити комп'ютерну модель інтелектуалізованої цифрової системи детектування рухомих (мобільних) об'єктів із можливістю мережевого обміну вимірювальною інформацією.

Хід виконання роботи:

1. Створити новий проєкт в середовищі CupCarbon: Project – New project.
2. Розробити та протестувати комп'ютерну модель базової структурно-алгоритмічної організації цифрової системи детектування рухомих (мобільних) об'єктів із можливістю мережевого обміну вимірювальними даними. Під час розробки комп'ютерної моделі рекомендовано використовувати наступні стандартні компоненти: маркери (markers) – для побудови траєкторії руху мобільного об'єкту; рухомий об'єкт (mobile); вузли бездротових цифрових давачів (sensor nodes) – для детектування сигналів щодо положення рухомого

об'єкту; базова станція (base station) – для агрегування сигналів від цифрових давачів. Під час розробки моделі необхідно збільшити радіус зони покриття базової станції та сенсорних вузлів таким чином, щоб була можливою інфокомунікація між компонентами моделі. Візуалізацію результатів приймання інформаційних повідомлень необхідно виконувати на вузлі базової станції. Типовий вигляд базової версії досліджуваної моделі наведено на рисунку 4.1. Скрипти до функціональних компонент досліджуваної комп'ютерної моделі системи наведено на рисунку 4.2. Типовий вигляд результату тестування досліджуваної комп'ютерної моделі наведено на рисунку 4.3.

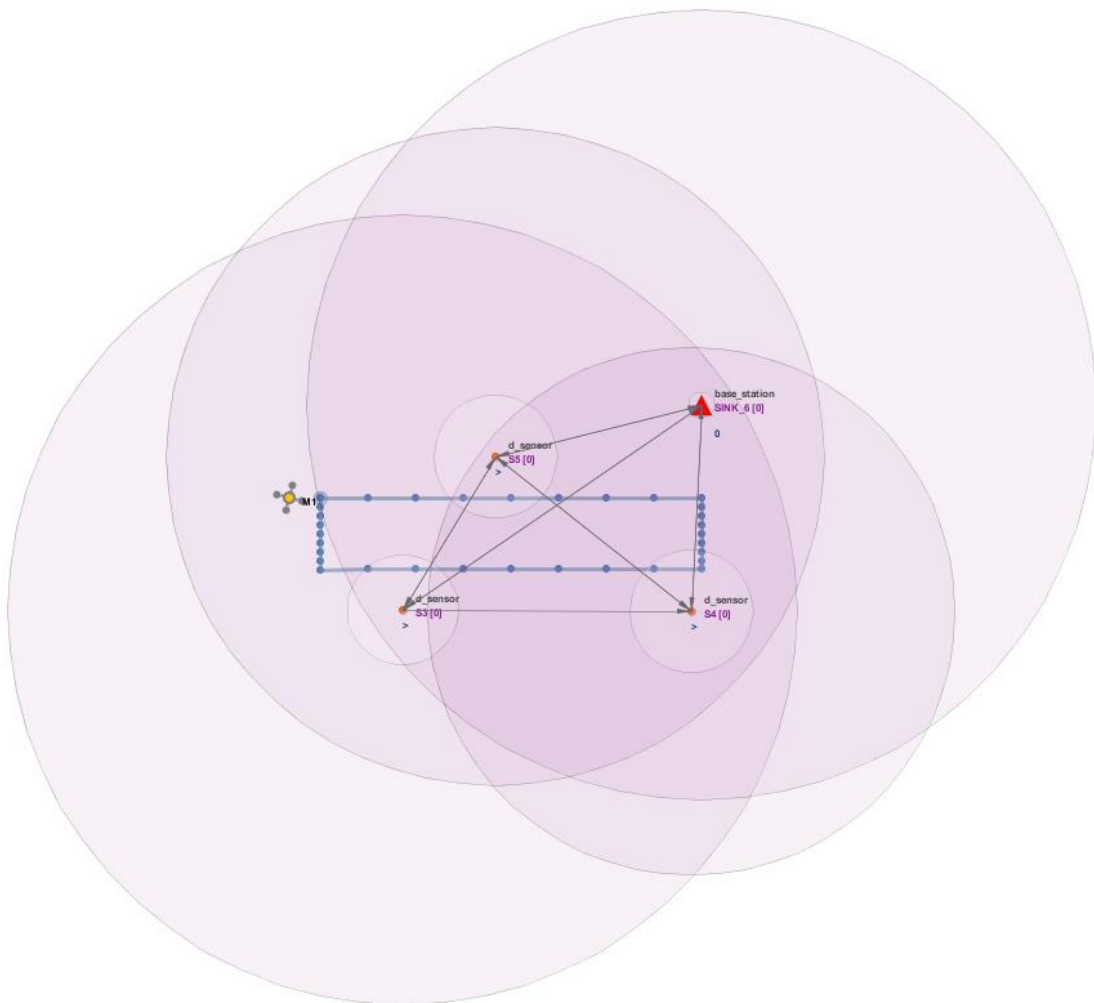


Рисунок 4.1 – Зовнішній вигляд досліджуваної комп'ютерної моделі системи детектування рухомих об'єктів

```

loop
dreadsensor a
send a
delay 500

```

а) сенсорні вузли

```

loop
wait
read a
print a

```

б) базова станція

Рисунок 4.2 – Типові скрипти функціональних компонент досліджуваної комп'ютерної моделі

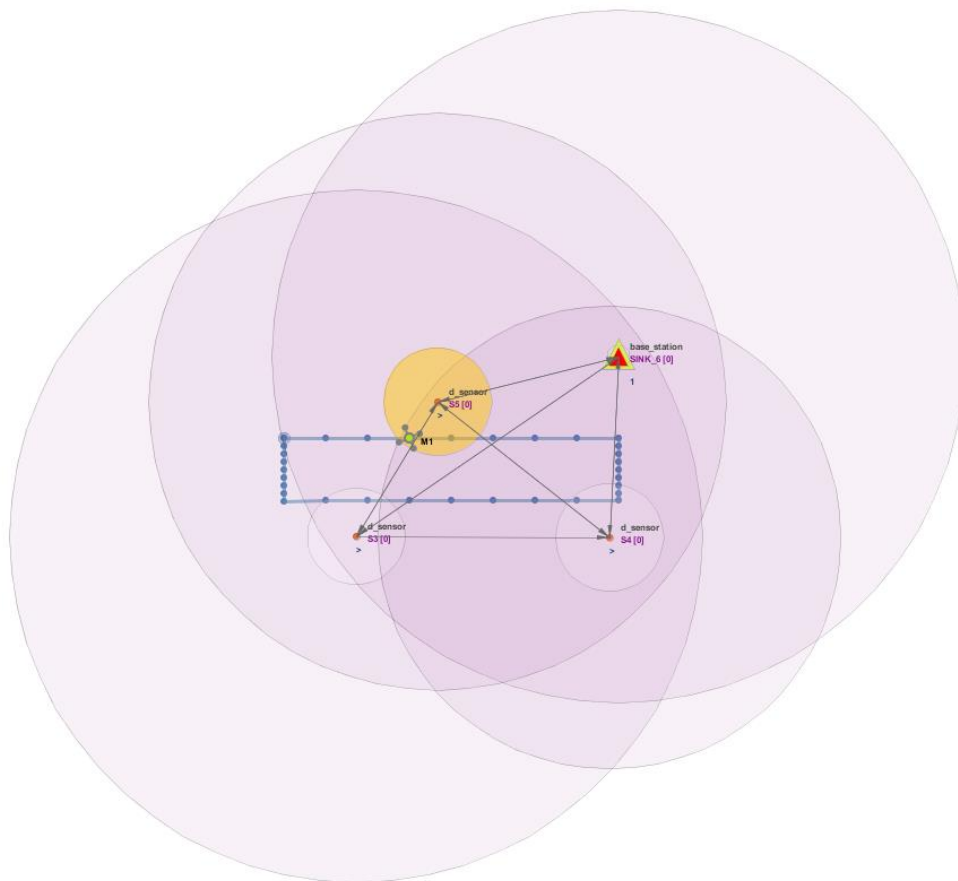


Рисунок 4.3 – Типовий результат тестування базової версії досліджуваної комп'ютерної моделі системи детектування рухомих об'єктів

3. Модифікувати розроблену у п. 2 комп'ютерну модель наступним чином:

3.1. Забезпечити відпрацювання моделлю наступного алгоритму. В якості інформаційних повідомлень, що відправляються цифровими бездротовими сенсорними вузлами, мають виступати координати того сенсорного вузла, що у відповідний момент часу детектує рухомий об'єкт. Відправка інформаційних повідомлень має здійснюватися лише тим вузлом, який детектує наявність

рухомого об'єкта в конкретний момент часу.

3.2. Додати до моделі другий мобільний об'єкт і прив'язати його до модифікованого маршруту (додати до проєкту новий маршрут на основі існуючого з меншою кількістю кроків: від 40 % до 60 % від початкової кількості) і перевірити коректність роботи моделі.

3.3. Додати до кожного сенсорного вузла маршрутизатори інформаційних повідомлень та реалізувати відправлення інформаційних повідомлень від сенсорних вузлів до маршрутизаторів за технологією Wi-Fi, а від маршрутизаторів до базової станції за технологією LoRa.

4. Протестувати модифіковану версію комп'ютерної моделі досліджуваної системи мережевого детектування рухомих об'єктів та валідувати коректність її роботи.

5. Навести приклади можливостей і сфер прикладного застосування результатів моделювання, отриманих у цій практичній роботі, під час проєктування та використання систем Industry 4.0.

6. *Опціонально:* перевірити працездатність розробленого апаратно-програмного рішення в лабораторних умовах.

Типовий зміст звіту до практичної роботи:

1. Титульний лист.

2. Мета та задачі практичної роботи.

3. Основні результати, що були отримані під час створення та тестування базової та модифікованої версій комп'ютерної моделі системи детектування рухомих об'єктів у складі систем Industry 4.0 за відповідними пунктами досліджень (скріншоти моделей, скрипти із відповідними коментарями, текстовий опис і графічна візуалізація основних результатів досліджень валідованих моделей).

4. Загальні висновки по роботі, що містять результати критичного аналізу на якісному й кількісному рівнях, а також рекомендації з оптимізації досліджуваної системи.

Контрольні питання:

1. У чому полягає функціональне призначення використаних у цій практичній роботі стандартних компонент середовища імітаційного моделювання CupCarbon?
2. Які стандартні функції створення скриптів CupCarbon були використані в цій практичній роботі? Яким є їх синтаксис? У чому полягає їх призначення?
3. Які бездротові технології обміну інформаційними повідомленнями були застосовані під час комп'ютерного експерименту в цій практичній роботі? Якими є їх основні характеристики?
4. Надайте характеристику типовому алгоритму одночасного використання декількох мережевих технологій на одному сенсорному вузлі? Яким чином це реалізується в середовищі моделювання CupCarbon?
5. Якими є функціональне призначення та синтаксис функцій radio та dreadsensor?
6. У чому полягає функціональне призначення інтелектуалізованих алгоритмів детектування місця положення рухомих об'єктів під час проєктування та впровадження систем Industry 4.0 на промислових підприємствах?

Перелік рекомендованих джерел інформації:

1. CupCarbon IoT. URL: <https://cupcarbon.com/> (дата звернення 15.11.2023).
2. Forster A. Introduction to Wireless Sensor Networks / A. Forster. New York: Wiley, 2016. 186 p.
3. Laktionov I.S. et al. Information model of the computer-integrated technology for wireless monitoring of the state of microclimate of industrial agricultural greenhouses. Instrumentation Measure Metrologie. 2021. Vol. 20 (6). P. 289 – 300.
4. Laktionov I. et al. Computer-Oriented Model for Network Aggregation of Measurement Data in IoT Monitoring of Soil and Climatic Parameters of Agricultural Crop Production Enterprises. BJMC, 2023, Vol. 11 (3). P. 500–522.

ПРАКТИЧНА РОБОТА № 5
ДОСЛІДЖЕННЯ МЕТОДІВ ТА АЛГОРИТМІВ ОЦІНКИ
ФУНКЦІОНАЛЬНО-МЕТРОЛОГІЧНИХ ХАРАКТЕРИСТИК СИСТЕМ
INDUSTRY 4.0

Мета роботи: закріпити теоретичні знання та розвинути практичні навички з методів і технік оцінки основних функціонально-метрологічних характеристик і параметрів інформаційно-вимірювальних компонент систем Industry 4.0.

Обладнання та програмне забезпечення: персональний комп'ютер, MS Office, безкоштовний сервіс MathCad Express (доступний за посиланням <https://www.mathcad.com/>).

Завдання до практичної роботи. Основне завдання практичної роботи полягає в побудові узагальненої структурної схеми інформаційно-вимірювального процесу та визначенні частинного виду функціональної залежності результату вимірювання фізичної величини U за результатами вимірювань інформативних величин X_1 , X_2 та X_3 та впливного параметру T з обліком ряду дестабілізуючих факторів. У, свою чергу, на першому етапі реалізації інформаційно-вимірювальної процедури оцінка інформативної величини U була отримана шляхом виконання опосередкованих вимірювань на базі прямих вимірювань параметрів X_1 , X_2 та X_3 . Оцінки вимірюваних величин X_1 , X_2 та X_3 були отримані за методикою накопичення та опрацювання груп прямих вимірювань із багаторазовими спостереженнями. Під час виконання опосередкованого вимірювання фізичної величини U на результат чинили негативний вплив наступні дестабілізуючі фактори: нестабільність джерела живлення (НДЖ) та вплив електромагнітних полів (ВЕР). Також у процесі вимірювань було встановлено, що результат сумісного вимірювання залежить від впливного фактору температури T у певному діапазоні.

Основні задачі до виконання є наступними:

1. На підставі лінгвістичного формулювання основного завдання практичної роботи необхідно розробити структурну схему інформаційно-вимірювальної процедури отримання результату сумісного вимірювання фізичної величини U . На схемі обов'язково мають бути відображені всі задіяні вимірювальні канали та етапи перетворень фізичних величин, що супроводжуються відповідними невизначеностями вимірювань.

2. У відповідності до індивідуальних завдань та методики, яку наведено в нижче, необхідно виконати опрацювання результатів прямих вимірювань із багаторазовими спостереженнями X_1 , X_2 та X_3 . У процесі оцінки результатів вимірювань X_1 , X_2 та X_3 для кожної величини мають бути використані по три групи результатів спостережень. Під час опрацювання цих груп обов'язково мають бути виконані наступні етапи: виявлені та виключені всі грубі невизначеності; встановлені математичні очікування та стандартні відхилення вимірюваних величин в абсолютній та відносній формі представлення; виконана оцінка середнього арифметичного математичних очікувань результатів вимірювань; кінцеві результати обчислень мають бути записані у відповідності до правил, які наведено нижче.

3. Виконати оцінку результату опосередкованого вимірювання величини $U=f(X_1, X_2, X_3)$ та сумарної граничної невизначеності з обліком негативного впливу дестабілізуючих факторів у відповідності до методики, яку наведено в нижче. Кінцевий результат обчислень опосередкованого вимірювання має бути записаний у відповідності до правил, які наведено нижче.

4. Визначити частинний вид функціональної залежності результату сумісного вимірювання $U=\varphi(T, X_1, X_2, X_3)$ та оцінити відповідні границі невизначеності апроксимації за методикою, яку наведено нижче. Під час виконання завдання рекомендується використання пакетів прикладних програм.

5. На підставі проведених розрахунків та аналізу отриманих результатів необхідно сформулювати рекомендації зі зменшення сумарної граничної невизначеності результату вимірювання фізичної величини U .

Таблиця 1.1 – Варіанти індивідуальних завдань до пп. 2 (результати спостережень фізичної величини X_1)

a	b												
	1	2	3	4	5	6	7	8	9	0			
1	3,251	3,232	3,233	3,244	3,235	3,236	3,247	3,248	3,247	3,019	3,228	3,657	3,251
2	3,231	3,254	3,229	3,244	3,239	3,257	3,221	3,224	3,019	3,221	3,629	3,248	3,242
3	3,232	3,223	3,228	3,263	3,238	3,248	3,232	3,213	3,008	3,212	3,238	3,647	3,303
4	3,223	3,252	3,267	3,262	3,237	3,239	3,253	3,232	3,217	3,002	3,627	3,266	3,214
5	3,234	3,241	3,256	3,221	3,246	3,238	3,264	3,251	3,246	3,232	3,016	3,645	3,265
6	3,265	3,262	3,255	3,232	3,245	3,258	3,245	3,002	3,255	3,253	3,655	3,244	3,266
7	3,256	3,243	3,264	3,223	3,234	3,247	3,226	3,013	3,254	3,244	3,254	3,622	3,227
8	3,247	3,254	3,273	3,234	3,023	3,246	3,247	3,234	3,243	3,255	3,623	3,233	3,248
9	3,248	3,265	3,242	3,245	3,012	3,255	3,268	3,225	3,232	3,246	3,222	3,631	3,239
0	3,239	3,006	3,231	3,226	3,211	3,234	3,279	3,236	3,251	3,625	3,331	3,209	3,213
	3,247	3,627	3,273	3,627	3,230	3,633	3,248	3,337	3,232	3,243			
	3,635	3,248	3,654	3,248	3,659	3,242	3,656	3,248	3,623	3,662			
	3,233	3,209	3,265	3,209	3,608	3,241	3,285	3,219	3,604	3,241			

Таблиця 1.2 – Варіанти індивідуальних завдань до пп. 2 (результати спостережень фізичної величини X_2)

a	b												
	1	2	3	4	5	6	7	8	9	0			
1	4,251	4,232	4,233	4,244	4,235	4,236	4,247	4,248	4,247	4,019	4,228	4,957	4,251
2	4,231	4,254	4,229	4,244	4,239	4,257	4,221	4,224	4,019	4,221	4,929	4,248	4,242
3	4,232	4,223	4,228	4,263	4,238	4,248	4,232	4,213	4,008	4,212	4,238	4,947	4,303
4	4,223	4,252	4,267	4,262	4,237	4,239	4,253	4,232	4,217	4,002	4,927	4,266	4,214
5	4,234	4,241	4,256	4,221	4,246	4,238	4,264	4,251	4,246	4,232	4,016	4,945	4,265
6	4,265	4,262	4,255	4,232	4,245	4,258	4,245	4,002	4,255	4,253	4,955	4,244	4,266
7	4,256	4,243	4,264	4,223	4,234	4,247	4,226	4,013	4,254	4,244	4,254	4,922	4,227
8	4,247	4,254	4,273	4,234	4,023	4,246	4,247	4,234	4,243	4,255	4,923	4,233	4,248
9	4,248	4,265	4,242	4,245	4,012	4,255	4,268	4,225	4,232	4,246	4,222	4,931	4,239
0	4,239	4,006	4,231	4,226	4,211	4,234	4,279	4,236	4,251	4,925	4,331	4,209	4,213
	4,247	4,927	4,273	4,927	4,230	4,933	4,248	4,337	4,232	4,243			
	4,935	4,248	4,954	4,248	4,959	4,242	4,956	4,248	4,923	4,962			
	4,233	4,209	4,265	4,209	4,908	4,241	4,285	4,919	4,404	4,241			

Таблиця 1.3 – Варіанти індивідуальних завдань до пп. 2 (результати спостережень фізичної величини X_3)

a	b												
	1	2	3	4	5	6	7	8	9	0			
1	2,251	2,232	2,233	2,244	2,235	2,236	2,247	2,248	2,247	2,019	2,228	2,657	2,251
2	2,231	2,254	2,229	2,244	2,239	2,257	2,221	2,224	2,019	2,221	2,629	2,248	2,242
3	2,232	2,223	2,228	2,263	2,238	2,248	2,232	2,213	2,008	2,212	2,238	2,647	2,303
4	2,223	2,252	2,267	2,262	2,237	2,239	2,253	2,232	2,217	2,002	2,627	2,266	2,214
5	2,234	2,241	2,256	2,221	2,246	2,238	2,264	2,251	2,246	2,232	2,016	2,645	2,265
6	2,265	2,262	2,255	2,232	2,245	2,258	2,245	2,002	2,255	2,253	2,655	2,244	2,266

7	2,256	2,243	2,264	2,223	2,234	2,247	2,226	2,013	2,254	2,244	2,254	2,622	2,227
8	2,247	2,254	2,273	2,234	2,023	2,246	2,247	2,234	2,243	2,255	2,623	2,233	2,248
9	2,248	2,265	2,242	2,245	2,012	2,255	2,268	2,225	2,232	2,246	2,222	2,631	2,239
0	2,239	2,006	2,231	2,226	2,211	2,234	2,279	2,236	2,251	2,625	2,331	2,209	2,213
	2,247	2,627	2,273	2,627	2,230	2,633	2,248	2,337	2,232	2,243			
	2,635	2,248	2,654	2,248	2,659	2,242	2,656	2,248	2,623	2,662			
	2,233	2,209	2,265	2,209	2,608	2,241	2,285	2,219	2,604	2,241			

Таблиця 1.4 – Варіанти індивідуальних завдань до пп. 3 (вид функціональної залежності)

a	0	1	2	3	4
Вид функції	$U = \frac{X_1}{X_2} \cdot \sqrt{X_3}$	$U = \frac{\sqrt{X_1}}{X_2} \cdot X_3$	$U = \frac{\sqrt{X_1 \cdot X_2}}{X_3}$	$U = \frac{X_2 \cdot X_3}{\sqrt{X_1}}$	$U = \sqrt{\frac{X_1}{X_2}} \cdot X_3$
a	5	6	7	8	9
Вид функції	$U = \frac{X_2}{X_1} \cdot \sqrt{X_3}$	$U = \frac{\sqrt{X_3}}{X_1} \cdot X_2$	$U = \frac{\sqrt{X_1 \cdot X_3}}{X_2}$	$U = \frac{X_1 \cdot X_2}{\sqrt{X_3}}$	$U = \sqrt{\frac{X_2}{X_3}} \cdot X_1$

Таблиця 1.5 – Варіанти індивідуальних завдань до пп. 3

b	0	1	2	3	4	5	6	7	8	9
НДЖ, %	6	5	4	7	5	3	8	5	4	7
ВЕР, %	5	6	7	4	5	7	3	4	5	5
Характеристики невизначеностей від нестабільності джерела живлення (НДЖ) та впливу електромагнітних полів (ВЕР) вказані у відсотках відносно основної стандартної невизначеності вимірювання фізичної величини U.										

Таблиця 1.6 – Варіанти індивідуальних завдань до пп. 4

b	0	1
Експериментальні точки	$U = \begin{bmatrix} 0,7 \cdot U_{ном.} \\ 0,9 \cdot U_{ном.} \\ U_{ном.} \\ 1,1 \cdot U_{ном.} \\ 1,3 \cdot U_{ном.} \end{bmatrix} \quad T = \begin{bmatrix} 5 \\ 10 \\ 13 \\ 15 \\ 19 \end{bmatrix}$	$U = \begin{bmatrix} 0,75 \cdot U_{ном.} \\ 0,85 \cdot U_{ном.} \\ U_{ном.} \\ 1,05 \cdot U_{ном.} \\ 1,1 \cdot U_{ном.} \end{bmatrix} \quad T = \begin{bmatrix} 3 \\ 7 \\ 11 \\ 15 \\ 17 \end{bmatrix}$
b	2	3
Експериментальні точки	$U = \begin{bmatrix} 0,75 \cdot U_{ном.} \\ 0,95 \cdot U_{ном.} \\ 1,05 \cdot U_{ном.} \\ 1,1 \cdot U_{ном.} \\ 1,15 \cdot U_{ном.} \end{bmatrix} \quad T = \begin{bmatrix} 3 \\ 9 \\ 14 \\ 15 \\ 21 \end{bmatrix}$	$U = \begin{bmatrix} 0,6 \cdot U_{ном.} \\ 0,8 \cdot U_{ном.} \\ U_{ном.} \\ 1,2 \cdot U_{ном.} \\ 1,3 \cdot U_{ном.} \end{bmatrix} \quad T = \begin{bmatrix} 4 \\ 9 \\ 11 \\ 15 \\ 19 \end{bmatrix}$
b	4	5

Експериментальні точки	$U = \begin{bmatrix} 0,65 \cdot U_{ном.} \\ 0,95 \cdot U_{ном.} \\ 1,1 \cdot U_{ном.} \\ 1,15 \cdot U_{ном.} \\ 1,2 \cdot U_{ном.} \end{bmatrix} \quad T = \begin{bmatrix} 3 \\ 8 \\ 11 \\ 15 \\ 21 \end{bmatrix}$	$U = \begin{bmatrix} 0,5 \cdot U_{ном.} \\ 0,75 \cdot U_{ном.} \\ U_{ном.} \\ 1,25 \cdot U_{ном.} \\ 1,35 \cdot U_{ном.} \end{bmatrix} \quad T = \begin{bmatrix} 4 \\ 9 \\ 15 \\ 17 \\ 19 \end{bmatrix}$
b	6	7
Експериментальні точки	$U = \begin{bmatrix} 0,85 \cdot U_{ном.} \\ 0,95 \cdot U_{ном.} \\ U_{ном.} \\ 1,1 \cdot U_{ном.} \\ 1,2 \cdot U_{ном.} \end{bmatrix} \quad T = \begin{bmatrix} 5 \\ 11 \\ 15 \\ 19 \\ 23 \end{bmatrix}$	$U = \begin{bmatrix} 0,55 \cdot U_{ном.} \\ 0,75 \cdot U_{ном.} \\ U_{ном.} \\ 1,15 \cdot U_{ном.} \\ 1,35 \cdot U_{ном.} \end{bmatrix} \quad T = \begin{bmatrix} 4 \\ 11 \\ 17 \\ 23 \\ 27 \end{bmatrix}$
b	8	9
Експериментальні точки	$U = \begin{bmatrix} 0,8 \cdot U_{ном.} \\ 0,9 \cdot U_{ном.} \\ U_{ном.} \\ 1,15 \cdot U_{ном.} \\ 1,2 \cdot U_{ном.} \end{bmatrix} \quad T = \begin{bmatrix} 5 \\ 14 \\ 16 \\ 18 \\ 21 \end{bmatrix}$	$U = \begin{bmatrix} 0,75 \cdot U_{ном.} \\ 0,95 \cdot U_{ном.} \\ U_{ном.} \\ 1,15 \cdot U_{ном.} \\ 1,45 \cdot U_{ном.} \end{bmatrix} \quad T = \begin{bmatrix} 6 \\ 13 \\ 19 \\ 24 \\ 28 \end{bmatrix}$
<i>В якості $U_{ном.}$ приймається результат опосередкованого вимірювання, який було знайдено в п. 3 індивідуальної роботи</i>		

У таблицях 1.1 – 1.6 використано такі умовні позначення: *a* – передостання цифра документа, що посвідчує особу; *b* – остання цифра документа, що посвідчує особу.

Хід виконання роботи:

Оцінювання невизначеності груп прямих багаторазових вимірювань.

Прямим називають таке вимірювання, в якому шукане значення фізичної величини знаходять безпосередньо на підставі дослідних даних: $Y=X$, де Y – це шукане значення вимірюваної фізичної величини, X – значення фізичної величини, яке знайдене шляхом реалізації процедури вимірювання та називається результатом вимірювання. З метою зменшення випадкової складової невизначеності на практиці використовують методику багаторазових прямих вимірювань фізичної величини за умов збіжності й єдності. Під час накопичення багаторазових спостережень фізичної величини за результат вимірювання приймають не результат окремого спостереження (одиничне значення з ряду результатів), а оброблений результат групи спостережень.

Задля обробки результатів спостережень та оцінки випадкової невизначеності використовується апарат теорії ймовірності та математичної статистики. Більшість випадкових невизначеностей найчастіше можуть бути описані нормальним законом розподілу. Отже, подальша методика оцінки результатів та невизначеності наведена саме для випадку нормального закону розподілу:

1. Обчислення математичних очікувань (найбільш імовірних значень) вимірюваних величин для кожного ряду результатів спостережень:

$$U_i = \frac{1}{N} \sum_{j=1}^N U_{i,j}, \quad (5.1)$$

де: $U_{i,j}$ – j -е спостереження фізичної величина в рамках i -ї серії результатів спостережень; N – кількість спостережень у відповідній серії.

2. Оцінка середнього арифметичного математичних очікувань результатів обробки за кожною серією результатів спостережень. Це значення приймається в якості найкращої оцінки вимірюваної фізичної величини та в подальших розрахунках виступає в якості результату вимірювання:

$$\bar{U} = \frac{1}{K} \sum_{i=1}^K U_i, \quad (5.2)$$

де K – кількість груп результатів спостережень, що були отримані за умов збіжності та єдності вимірювань.

3. Обчислення внутрішньогрупових дисперсій (D_{U_i}) для кожної серії дослідних даних за умов збіжності та єдності вимірювань:

$$D_{U_i} = \frac{1}{N-1} \sum_{j=1}^N (U_{i,j} - U_i)^2. \quad (5.3)$$

4. Оцінка експериментальної дисперсії ($D_{\bar{U}}$) математичних очікувань

груп результатів спостережень величин за умов збіжності та єдності вимірювань:

$$D_{\bar{U}} = \frac{1}{K-1} \sum_{i=1}^K (U_i - \bar{U})^2. \quad (5.4)$$

5. Перевірка статистичної гіпотези щодо значимості впливу міжгрупової дисперсії відносно внутрішньогрупової складової на підставі критерію Фішера (F -розподілу). Задля цього під час першого етапу виконується розрахунок значення критерію Фішера ($F_{експ.}$), що може бути отримане на підставі експериментальних даних (груп спостережень) за формулою:

$$F_{експ.} = \frac{S_I^2}{S_{II}^2} = \frac{\frac{N}{K-1} \sum_{i=1}^K (U_i - \bar{U})^2}{\frac{1}{K(N-1)} \sum_{i=1}^K \sum_{j=1}^N (U_{i,j} - \bar{U})^2}, \quad (5.5)$$

де S_I та S_{II} – незалежні оцінки випадкового розподілу отриманих результатів спостережень фізичної величини.

Другий етап полягає в перевірці наступної умови:

$$F_{експ.} < F_P, \quad (5.6)$$

де F_P – гранично допустиме значення F -критерію за довірчої ймовірності P .

Під час реалізації третього етапу перевірки статистичної гіпотези виконується розрахунок підсумкової випадкової невизначеності результату вимірювання фізичної величини.

У випадку, якщо умова (5.6) виконується, то впливом міжгрупової дисперсії можна знехтувати та сумарне стандартне відхилення результатів спостережень фізичної величини (σ_U) розраховується за формулою:

$$\sigma_U = \sqrt{\frac{(K-1)S_I^2 + K(N-1)S_{II}^2}{K \cdot N(K \cdot N - 1)}}. \quad (5.7)$$

У тому випадку, коли умова (5.6) не виконується, то вплив міжгрупової дисперсії є суттєвим та сумарне стандартне відхилення результатів спостережень фізичної величини (σ_U) розраховується за наступною формулою:

$$\sigma_U = \sqrt{\frac{1}{K(K-1)} \sum_{i=1}^K (U_i - \bar{U})^2}. \quad (5.8)$$

6. Оцінка відносної основної випадкової невизначеності результату вимірювання фізичної величини (δ_U), яка обумовлена неідеальністю компонент засобів інформаційно-вимірювальної техніки, виконується за формулою:

$$\delta_U = \frac{\sigma_U}{\bar{U}} \cdot 100 \%. \quad (5.9)$$

де σ_U – сумарне стандартне відхилення результатів спостережень фізичної величини; $\bar{U}$ – оцінка середнього арифметичного математичних очікувань результатів обробки за кожною серією результатів спостережень.

Вищеописана методика у вигляді блок-схеми алгоритму обчислень наведена на рисунку 5.1.

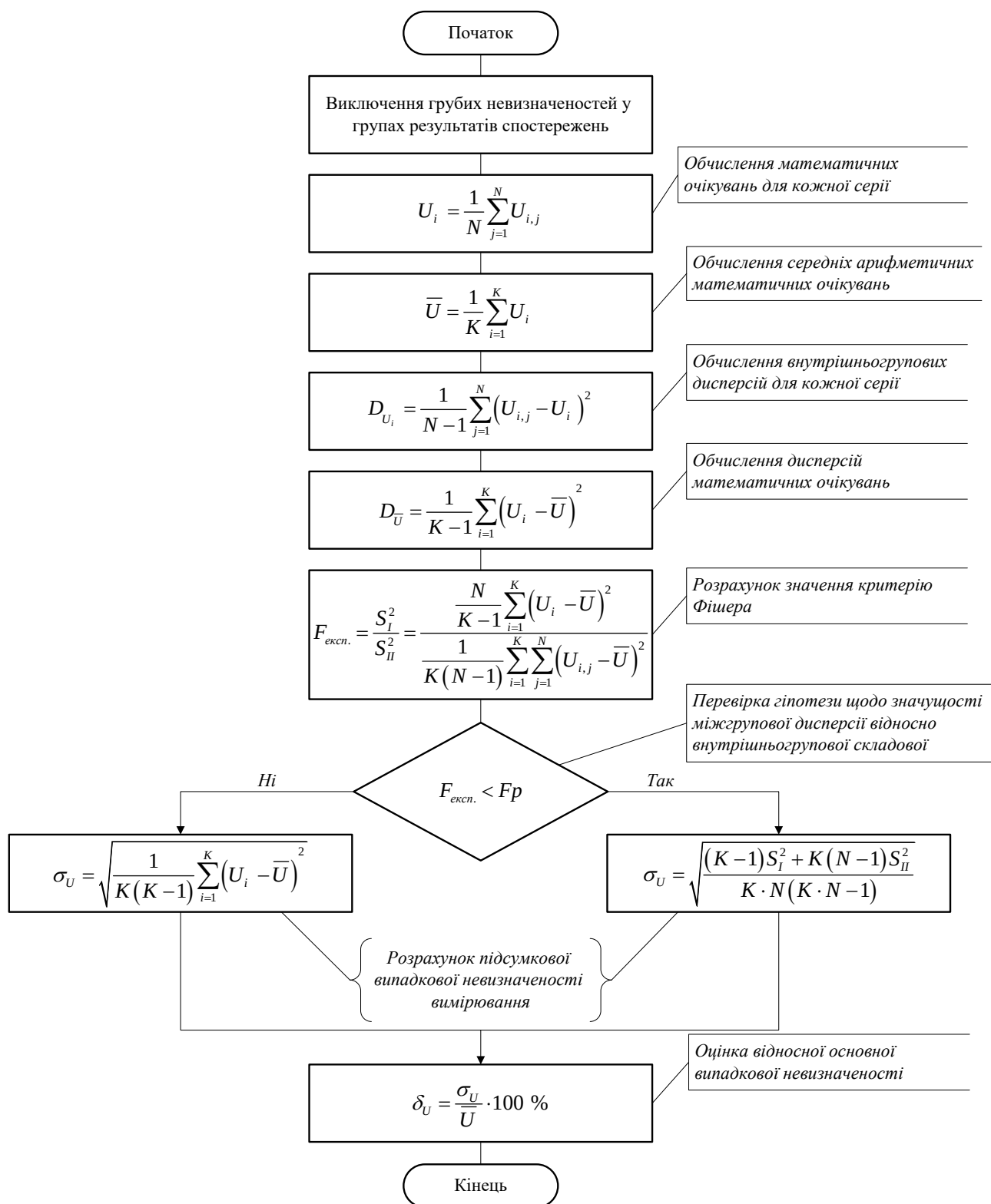


Рисунок 5.1 – Блок-схема алгоритму оцінювання невизначеності груп прямих багаторазових вимірювань

Таким чином, методика, яку у вигляді блок-схеми алгоритму наведено на рисунку 5.1, може бути використана в якості базової під час опрацювання груп

прямих багаторазових вимірювань фізичних величин на предмет оцінки основної невизначеності.

Виявлення та виключення грубих невизначеностей (промахів) у групах результатів спостережень. Невизначеності результатів спостережень і вимірювань, так само, як і результати вимірювань фізичних величин, потрібно під час аналізу розглядати як випадкові величини, що підкоряються ймовірнісним законам розподілу. Грубі невизначеності (промахи) – це аномально великі за абсолютним значенням випадкові невизначеності.

Під час інженерних експериментів між результатами, в яких містяться грубі невизначеності, та результатами, що не обтяжені промахами (заслужують довіри), у більшості випадків не можна встановити чітку межу. Отже, завдання визначення того чи містить даний результат грубу похибку, вирішується одним із відомих способів перевірки статистичних імовірнісних гіпотез. Відносно виявлення грубих невизначеностей статистична гіпотеза, яка перевіряється, допускає той факт, що кожен результат із ряду спостережень U_i не містить грубої невизначеності та є одним із можливих значень вимірюваної величини U зі встановленим законом розподілу ймовірностей (у більшості практичних випадків – це нормальний закон розподілу).

Проте, сумнів у першу чергу мають викликати найбільший (U_{max}) і найменший (U_{min}) із аналізованих рядів результатів спостережень. Методика перевірки гіпотези відносно вияву грубих невизначеностей зводиться до виконання наступного алгоритму.

Для кожного аналізованого ряду результатів спостережень фізичних величин, які отримано під час багаторазових вимірювань, розраховують статистичні оцінки параметрів нормального закону розподілу ймовірностей, а саме:

– оцінку математичного очікування ряду:

$$\bar{U} = \frac{1}{N} \sum_{i=1}^N U_i, \quad (5.10)$$

– оцінку дисперсії ряду:

$$D_U = \frac{1}{N-1} \cdot \sum_{i=1}^N (U_i - \bar{U})^2. \quad (5.11)$$

Після цього виконується розрахунок допоміжних відношень за нижченаведеними залежностями:

$$\begin{cases} v_1 = \frac{U_{\max} - \bar{U}}{\sqrt{D_U}}, \\ v_2 = \frac{\bar{U} - U_{\min}}{\sqrt{D_U}}. \end{cases} \quad (5.12)$$

Наступним кроком є знаходження допустимого граничного значення коефіцієнта (v_p), яке випадкова величина (v) може приймати. Це завдання виконується за довідковими даними залежно від кількості результатів спостережень у відповідній групі (N) та прийнятої довірчої ймовірності (P).

У тому випадку, якщо обчислене значення коефіцієнта v_1 (або v_2 , або v_1 та v_2 одночасно) виявляється меншим за гранично допустимий коефіцієнт v_p , то вищезаявлена гіпотеза щодо грубих невизначеностей приймається, і формулюється висновок про те, що результат $U_{\max}$ (або $U_{\min}$, або $U_{\max}$ та $U_{\min}$ одночасно) не містить грубої невизначеності.

У тому випадку, якщо розраховане значення коефіцієнта v_1 (або v_2 , або v_1 та v_2 одночасно) виявиться більшим за гранично допустимий коефіцієнт v_p , то відповідний результат спостереження $U_{\max}$ (або $U_{\min}$, або $U_{\max}$ та $U_{\min}$ одночасно) вважається таким, що обтяжений грубою невизначеністю. У такому разі відповідний результат спостережень $U_{\max}$ (або $U_{\min}$, або $U_{\max}$ та $U_{\min}$ одночасно) має бути виключений із ряду спостережень та отриманий «новий» ряд має бути знову перевірений на наявність промахів за вищенаведеним алгоритмом. Варто зазначити, що під час перевірки нового ряду мають бути прийняті до уваги

оцінки статистичних параметрів після зменшення числа результатів спостережень.

Також необхідно прийняти до уваги, що надійність кожного з можливих суджень визначається прийнятою довірчою ймовірністю (P).

Оцінка невизначеності опосередкованих вимірювань. Опосередковане вимірювання – це вид непрямого вимірювання однієї фізичної величини з подальшим перетворенням її роду та/або розрахунками за результатами прямих вимірювань інших фізичних величин, з якими шукана величина пов'язана відомою залежністю.

На теперішній час найбільшого розповсюдження набули дві методики оцінки невизначеності опосередкованих вимірювань:

- оцінка невизначеності за умови обліку некорельованих вхідних величин;
- оцінка невизначеності за умови обліку корельованих вхідних величин.

У випадку некорельованих вхідних фізичних величин алгоритм оцінки результату та невизначеності опосередкованих вимірювань є наступним.

1. Для прямих вимірювань виконується оцінка результату вимірювання (U_i) та стандартної невизначеності (σ_{U_i}) за методикою, яку вище.

2. Знаходиться оцінка результату опосередкованого вимірювання за відомою функціональною залежністю: $Y=f(U_1, U_2, \dots, U_n)$.

3. Визначаються вагові коефіцієнти кожної частинної невизначеності $\left(\partial Y / \partial U_i\right)$.

4. Знаходиться стандартна невизначеність результату опосередкованого вимірювання за формулою:

$$\sigma_Y = \sqrt{\sum_{i=1}^N \left(\frac{\partial Y}{\partial U_i} \cdot \sigma_{U_i} \right)^2}. \quad (5.13)$$

5. Виконується розрахунок розширеної невизначеності за нижченаведеною формулою:

$$\varepsilon_Y = \pm k_{t_p} \cdot \sigma_Y, \quad (5.14)$$

де k_{t_p} – значення коефіцієнту охоплення, що чисельно дорівнює коефіцієнту Стюдента, який визначається на підставі довірчої ймовірності (P) та кількості багаторазових спостережень при знаходженні оцінок прямих вимірювань (U_i).

У випадку корельованих вхідних фізичних величин алгоритм оцінки результату та невизначеності опосередкованих вимірювань є аналогічним до наведеного вище за виключенням того, що стандартна невизначеність має враховувати коефіцієнти кореляції між результатами прямих вимірювань. Формула для розрахунку стандартної невизначеності в цьому випадку є наступною:

$$\sigma_Y = \sqrt{\sum_{i=1}^N \left(\frac{\partial Y}{\partial U_i} \cdot \sigma_{U_i} \right)^2 + 2 \sum_{j=1}^{N-1} \sum_{i=2}^N \left(\frac{\partial Y}{\partial U_i} \cdot \frac{\partial Y}{\partial U_j} \cdot r(U_i, U_j) \cdot \sigma_{U_i} \cdot \sigma_{U_j} \right)}, \quad (5.15)$$

де $r(U_i, U_j)$ – коефіцієнт кореляції, що розраховується на підставі попарних оцінок кореляційних моментів.

Вищеописана методика у вигляді блок-схеми алгоритму обчислень наведена на рисунку 5.2.

Отже, методика, яку у вигляді відповідної блок-схеми алгоритму наведено на рисунку 5.2, може бути використана в якості базової під час опрацювання результатів опосередкованих вимірювань із корельованими та некорельованими вхідними параметрами.

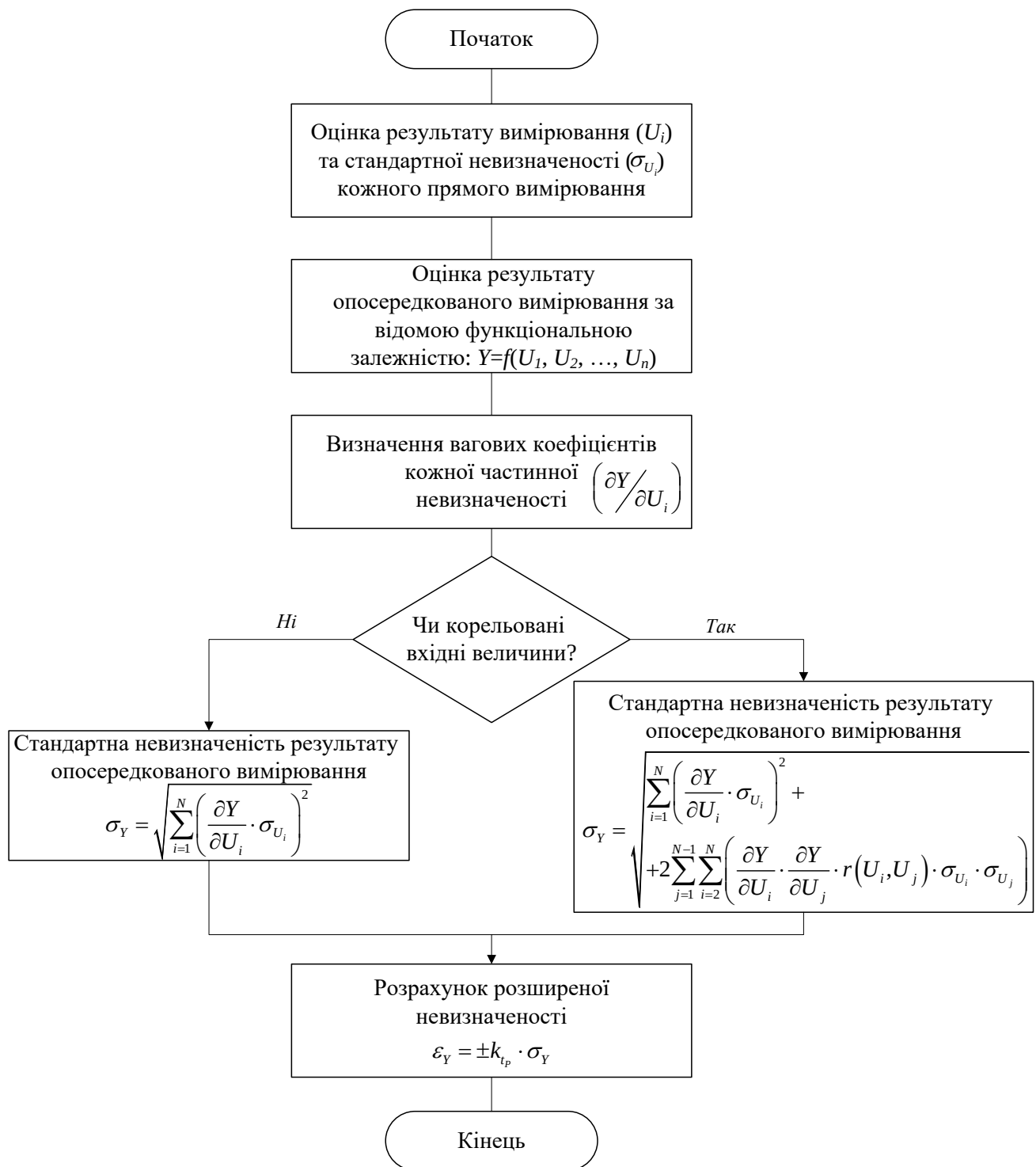


Рисунок 5.2 – Блок-схема алгоритму оцінювання невизначеності опосередкованих вимірювань

Правила складання невизначеностей та запису кінцевого результату вимірювання. На теперішній час у галузі інформаційно-вимірювальних технологій існує значна кількість підходів до методик складання

невизначеностей вимірювань від різних джерел у залежності від класифікації факторів, якими обумовлено їх появу.

Отже, на практиці в більшості випадків на результат вимірювання можуть одночасно впливати декілька джерел невизначеностей. У прикладній теорії інформаційно-вимірювальної техніки існує правило, відповідно до якого обов'язково потрібно оцінювати межі можливих значень невизначеностей, якщо їх облік і подальше виключення в будь-який відомий спосіб є неможливим.

Таким чином, для отримання сумарної граничної невизначеності результату вимірювання необхідно скласти стандартні невизначеності від впливу всіх джерел. При цьому розподіл складових граничної невизначеності всередині відповідних меж рекомендується вважати таким, що підкорюється рівномірному закону розподілу, а отже сумарна невизначеність результату вимірювання має розраховуватись за формулою:

$$\varepsilon_S(P) = k_P \cdot \sqrt{\sum_{i=1}^m \varepsilon_{S_i}^2}, \quad (5.16)$$

де k_P – коефіцієнт, який визначається встановленою довірчою ймовірністю (P); m – кількість складових сумарної невизначеності; ε_{S_i} – межа відповідної складової невизначеності вимірювання.

Правила округлення результатів і невизначеностей вимірювань є наступними.

Невизначеність результату вимірювання, яка представлена в абсолютній формі, вказується двома значущими цифрами (будь-яка цифра, за виключенням нуля, якщо він знаходиться не в середині), якщо перша з них 1 або 2, і однією, якщо перша цифра дорівнює 3 та більше.

У тому випадку, коли цифра старшого з розрядів, що відкидається, менше 5, то цифри, які залишаються не змінюються. Якщо цифра старшого з розрядів, що відкидається, більше або дорівнює 5, проте за нею слідує відмінні від

нуля цифри, то остання цифра, яка зберігається, збільшується на одиницю.

У тому випадку, коли цифра, що відкидається, дорівнює 5, а наступні за нею є невідомими або нуль, то остання цифра числа не змінюється, якщо вона є парною та збільшується на одиницю, якщо вона є непарною.

Результат вимірювання фізичної величини округлюється до того ж десяткового знака, на якому закінчується округлене значення невизначеності результату.

Процедура округлення реалізується лише під час остаточного запису результату та невизначеності вимірювання, а всі проміжні обчислення виконуються з одним або двома надлишковими знаками.

Оцінка невизначеності сумісних вимірювань. Під час реалізації сумісних вимірювань невідомі фізичні величини Y_i , які підлягають безпосереднім вимірюванням, мають визначатись за результатами вимірювань інших величин, що функціонально пов'язані з ними.

Основним завданням обробки результатів сумісних вимірювань є побудова аналітичної залежності за наявними сумісними вимірюваннями двох (або декількох) фізичних величин. У загальному випадку вид залежності $Y=f(T)$ заздалегідь є невідомим та визначається на підставі наявних результатів експериментальних випробувань. У більшості випадків передбачуваний вид функціональної залежності $Y=f(T)$ відомий заздалегідь на підставі будь-яких теоретичних міркувань та фізичних законів, а невідомими залишаються параметри цієї залежності.

Під час визначення частинного виду залежності $Y=f(T)$ в аналітичному вигляді рекомендується дотримуватись наступного алгоритму:

1. Побудувати графічний вигляд залежності $Y=f(T)$ на підставі експериментальних даних.
2. Встановити загальний вигляд залежності апроксимації функції:

$$Y = f(T, A_i), \quad (5.17)$$

де A_i – параметри залежності, які підлягають знаходженню.

3. Визначити метод апроксимації або регресійного аналізу функціональної залежності (5.17).

4. Обчислити чисельні значення параметрів (A_i), що входять до функціональної залежності (5.17).

5. Оцінити невизначеність апроксимації залежності (5.17).

У теперішній час в інженерній практиці знайшли широке застосування багато способів апроксимації та регресійного аналізу функцій, проте найпоширенішим є метод найменших квадратів (МНК).

У практичній діяльності під час сумісних вимірювань найбільш розповсюдженою задачею є встановлення залежності вимірюваної фізичної величини (Y) від дестабілізуючого фактора (T). При цьому в більшості випадків для апроксимації використовується лінійна залежність, яка встановлюється за методом найменших квадратів із використанням пакетів прикладних програм MS Excel або MathCad.

Невизначеність апроксимації результату сумісного вимірювання ($\sigma_{Y_{appr}}$) визначається за формулою:

$$\sigma_{Y_{appr}} = \sqrt{\frac{1}{m} \sum_{i=1}^m (Y_{teor_i} - Y_{exper_i})^2}, \quad (5.18)$$

де Y_{teor_i} – значення вимірюваної величини, що отримане в результаті апроксимації; Y_{exper_i} – експериментальне значення вимірюваної величини; m – кількість експериментальних точок, за якими виконувалась апроксимація.

Типовий зміст звіту:

1. Титульний лист.
2. Мета та задачі практичної роботи.
3. Результати досліджень практичної роботи за наступними пунктами:
 - 3.1. Розробка структурної схеми інформаційно-вимірювальної процедури результату сумісного вимірювання фізичної величини.

3.2. Опрацювання результатів прямих вимірювань із багаторазовими спостереженнями.

3.3. Оцінка результату опосередкованого вимірювання фізичної величини.

3.4. Опрацювання результатів сумісного вимірювання.

3.5. Формулювання рекомендацій зі зменшення сумарної граничної невизначеності результату вимірювання фізичної величини.

4. Висновки.

Контрольні питання:

1. Надайте узагальнену характеристику інформаційно-вимірювальної процедури щодо оцінки функціонально-метрологічних характеристик прямого та опосередкованого вимірювання фізичних параметрів технологічних процесів?

2. Дайте визначення наступним термінам: невизначеність вимірювання, груба невизначеність, опосередковане вимірювання та сумісне вимірювання.

3. Охарактеризуйте основні етапи алгоритму оцінювання невизначеності груп прямих багаторазових вимірювань.

4. Охарактеризуйте основні етапи алгоритму оцінювання невизначеності опосередкованих вимірювань.

5. Охарактеризуйте основні етапи алгоритму оцінювання невизначеності сумісних вимірювань.

6. Наведіть приклади прикладного застосування результатів, отриманих у цій практичній роботі, під час проєктування та застосування систем Industry 4.0 у реальних умовах експлуатації.

Перелік рекомендованих джерел інформації:

1. Поджаренко В.О., Васілевський О.М., Кучерук В.Ю. Опрацювання результатів вимірювань на основі концепції невизначеності: навч. посібник. Вінниця: ВНТУ, 2008. 128 с.

2. Kharchenko V.S. Internet of Things for Industry and Human Application. In Volumes 1–3. Vol. 1. Fundamentals and Technologies / V.S. Kharchenko (ed.). Kyiv: Ministry of Education and Science of Ukraine, NAU ‘KhAI’, 2019. 605 p.

3. Вовна О.В., Лактіонов І.С., Лебедєв В.А. Комп'ютерно-інтегрований моніторинг та керування в промислових теплицях: поточні результати і перспективи досліджень: монографія. Покровськ: ДВНЗ «ДонНТУ», 2020. 255 с.

4. Kharchenko V.S. Internet of Things for Industry and Human Application. In Volumes 1–3. Vol. 3. Assessment and Implementation / V.S. Kharchenko (ed.). Ministry of Education and Science of Ukraine, Nation. Aeros. Un. KhAI, 2019. 918 p.

5. Laktionov I., Vovna O., Getman I., Maryna A., Lebediev V. Results of experimental research on computerized intellectual monitoring means of effective greenhouse illumination. International Journal on Smart Sensing and Intelligent Systems. 2019. Vol. 12 (1). P. 1–19.

6. Vovna O.V., Laktionov I.S., Dobrovolska L.O., Kabanets M.M., Lebediev V.A. Evaluation of metrological characteristics of a computerized conductivity meter of irrigation solution based on the uncertainty theory. Journal Europeen des Systemes Automatises. 2019. Vol. 52 (4). P. 333–340.

7. Vovna O.V., Laktionov I.S., Koyfman O.O., Stashkevych I.I., Lebediev V.A. Study of Metrological Characteristics of Low-Cost Digital Temperature Sensors for Greenhouse Conditions. Serbian Journal of Electrical Engineering. 2020. Vol. 17 (1). P. 1–20.

ПРАКТИЧНА РОБОТА № 6

РОЗРОБКА ТА ДОСЛІДЖЕННЯ МОДЕЛЕЙ АПРОКСИМАЦІЇ ЧАСОВИХ РЯДІВ РЕЗУЛЬТАТІВ КОМП'ЮТЕРИЗОВАНОГО МОНІТОРИНГУ ПАРАМЕТРІВ ТЕХНОЛОГІЧНИХ ПРОЦЕСІВ

Мета роботи: отримати практичні навички розробки та дослідження штучних нейронних мереж для вирішення прикладних завдань щодо алгоритмів апроксимації експериментальних рядів результатів комп'ютеризованого моніторингу параметрів технологічних процесів.

Обладнання та програмне забезпечення: персональний комп'ютер, пакет прикладних програм Matlab & Simulink (студентська безкоштовна версія може бути завантажена за посиланням <https://www.mathworks.com/campaigns/products/trials.html>), MS Office.

Основні теоретичні відомості. Під штучними нейронними мережами маються на увазі обчислювальні структури, які моделюють прості біологічні процеси, які зазвичай асоціюються з процесами, що протікають у людському мозку. Вони являють собою розподілені та паралельні системи, що здатні до адаптивного навчання шляхом аналізу позитивних та негативних впливів. Елементарним структурним вузлом (перетворювачем) у даних мережах є штучний нейрон. На теперішній час науковцями запропоновано та досліджено значну кількість моделей нейроноподібних елементів і нейронних мереж. Нейрон є обов'язковою функціональною складовою нейронної мережі загальну структуру, якої наведено на рисунку 6.1.

Така типова структура штучної нейронної мережі складається з трьох основних елементів: помножувачів (синапсів), суматора та нелінійного перетворювача. Синапси відповідають за зв'язки між нейронами, множать вхідний сигнал на число, що характеризує силу зв'язку (ваговий коефіцієнт синапсу). Суматор виконує додавання сигналів, що надходять за синаптичними

зв'язками від інших нейронів та зовнішніх входних сигналів. Нелінійний перетворювач реалізує нелінійну функцію одного аргументу – виходу суматора. Ця функція називається функцією активації або передавальною функцією нейрона. Нейрон у цілому реалізує скалярну функцію векторного аргументу.

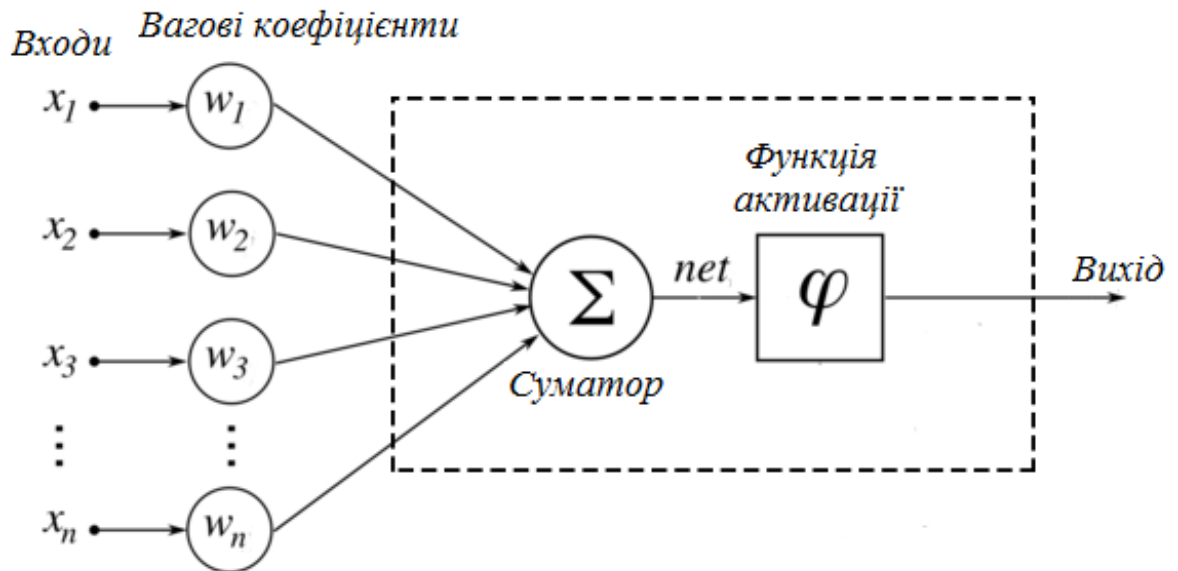


Рисунок 6.1 – Загальна структура штучної нейронної мережі

Математична модель нейрона є наступною:

$$S = \sum_{i=1}^n w_i \cdot x_i + b; \quad (6.1)$$

$$y = f(S),$$

де S – результат складання; w_i – вага відповідного синапса; x_i – відповідний входний сигнал; b – адитивна складова зсуву сигналу; i – номер входу; n – кількість входів; y – вихідний сигнал; f – функція активації (нелінійного перетворення).

У загальному випадку входний сигнал, вагові коефіцієнти та зсув сигналу можуть приймати дійсні значення, проте в багатьох практичних задачах – лише деякі фіксовані значення. Вихідний сигнал (y) визначається конкретним видом функції активації та може бути як дійсним, так і цілим числом.

Синаптичні зв'язки з позитивною вагою є такими, що збуджують мережу, а з негативною – такими, що гальмують. Описаний формулою (6.1) обчислювальний елемент можна вважати спрощеною математичною моделлю біологічних нейронів. На вхідний сигнал (S) за допомогою функції активації нелінійний перетворювач відповідає вихідним сигналом $f(S)$, який представляє собою вихід нейрона (y). Приклади активаційних функцій наведено в табл. 6.1.

Таблиця 6.1 – Приклади функцій активації

№ з/п	Найменування функції	Аналітичний вираз	Область значень
1	Лінійна	$F(\varphi) = k \cdot \varphi$	від $-\infty$ до $+\infty$
2	Гранична	$F(\varphi) = \begin{cases} 0, & \varphi \leq 0 \\ 1, & \varphi > 0 \end{cases}$	0; 1
3	Знакова	$F(\varphi) = \begin{cases} -1, & \varphi \leq 0 \\ 1, & \varphi > 0 \end{cases}$	-1; 1
4	Сигмовидна	$F(\varphi) = \frac{1}{1 + e^{-\varphi}}$	від 0 до 1
5	Напівлінійна	$F(\varphi) = \begin{cases} 0, & \varphi \leq 0 \\ k \cdot \varphi, & \varphi > 0 \end{cases}$	від 0 до $+\infty$
6	Радіальна базисна	$F(\varphi) = e^{-\varphi^2}$	від 0 до 1
7	Напівлінійна з насиченням	$F(\varphi) = \begin{cases} 0, & \varphi \leq 0 \\ k \cdot \varphi, & 0 < \varphi < 1 \\ 1, & \varphi \geq 1 \end{cases}$	від 0 до 1
8	Квадратична	$F(\varphi) = \varphi^2$	від 0 до $+\infty$
9	Гіперболічний тангенс	$F(\varphi) = \frac{e^{\varphi} - e^{-\varphi}}{e^{\varphi} + e^{-\varphi}}$	від -1 до 1
10	Трикутна	$F(\varphi) = \begin{cases} 1 - \varphi , & \varphi \leq 1 \\ 0, & \varphi > 1 \end{cases}$	від 0 до 1

Тип функції активації обирається з урахуванням конкретного завдання, що підлягає розв'язанню зі застосуванням штучних нейронних мереж. Наприклад, у задачах, які пов'язані з апроксимацією та класифікацією перевага віддається нелінійним сигмоїдальним кривим. Для побудови штучних нейронних мереж із чітким технічним завданням потрібно обрати тип з'єднання нейронів, визначити вид передавальних функцій елементів та підібрати вагові

коефіцієнти зв'язків між нейронами мережі.

Перелік функцій для роботи зі штучними нейронними мережами у пакеті прикладних програм Matlab з детальним описом їх синтаксису й призначення є наступним:

**network*: $\text{net}=\text{network}(\text{numInputs}, \text{numLayers}, \text{biasConnect}, \text{inputConnect}, \text{layerConnect}, \text{outputConnect}, \text{targetConnect})$. Функція повертає створену нейронну мережу з ім'ям *net* із наступними характеристиками: *numInputs* – кількість входів, *numLayers* – кількість шарів, *biasConnect* – булевий вектор із числом елементів, що дорівнює кількості шарів, *inputConnect* – булева матриця з числом рядків, що дорівнює кількості шарів, *layerConnect* – булева матриця з числом рядків і стовпців, що дорівнює кількості шарів, *outputConnect* – булевий вектор-рядок з числом елементів, що дорівнює кількості шарів, *targetConnect* – вектор-рядок, з числом елементів, що дорівнює кількості шарів.

**newsc*: $\text{net}=\text{newsc}(\text{PR}, \text{S}, \text{KLR}, \text{CLR})$. Функція створення конкурентного шару Кохонена. Аргументи функції: *PR* – матриця мінімальних і максимальних значень для *N* вхідних елементів, *S* – кількість нейронів, *KLR* – коефіцієнт навчання Кохонена, *CLR* – коефіцієнт розшарування.

**newscf*: $\text{net}=\text{newscf}(\text{PR}, [\text{S}_i], [\text{TF}_i], \text{BTF}, \text{BLF}, \text{PF})$. Функція створення різновиду багатошарової нейронної мережі зі зворотним поширенням помилки. Аргументи функції: *PR* – матриця мінімальних і максимальних значень для *R* вхідних елементів, *S_i* – розмір *i*-го прихованого шару, *TF_i* – функція активації нейронів *i*-го шару, *BTF* – функція навчання мережі, *BLF* – функція налаштування ваг і зсуву, *PF* – функція помилки.

**newclm*: $\text{net}=\text{newclm}(\text{PR}, [\text{S}_i], [\text{TF}_i], \text{BTF}, \text{BLF}, \text{PF})$. Функція створення мережі Елмана. Призначення аргументів співпадає з функцією **newscf*.

**newff*: $\text{net}=\text{newff}(\text{PR}, [\text{S}_i], [\text{TF}_i], \text{BTF}, \text{BLF}, \text{PF})$. Функція створення односпрямованої мережі з навчанням за методом зворотного поширення помилки. Призначення аргументів співпадає з функцією **newscf*.

**newfftd*: $\text{net}=\text{newfftd}(\text{PR}, \text{ID}, [\text{S}_i], [\text{TF}_i], \text{BTF}, \text{BLF}, \text{PF})$. Функція створення

односпрямованої мережі зі вхідними затримками. Призначення аргументів співпадає з функцією **newscf*, за виключенням додаткового аргументу *ID* – вектору вхідних затримок.

Типові приклади.

Приклад № 1. Апроксимація даних поліноміальною залежністю виду $y = x_1^3 + x_2^2 - 2 \cdot x_1 \cdot x_2$ у діапазоні вхідних параметрів (x_1 та x_2) від 0 до 10.

1.1. Для формування вектора навчальних прикладів у вікні команд необхідно набрати код програми, як показано на рисунку 6.2.

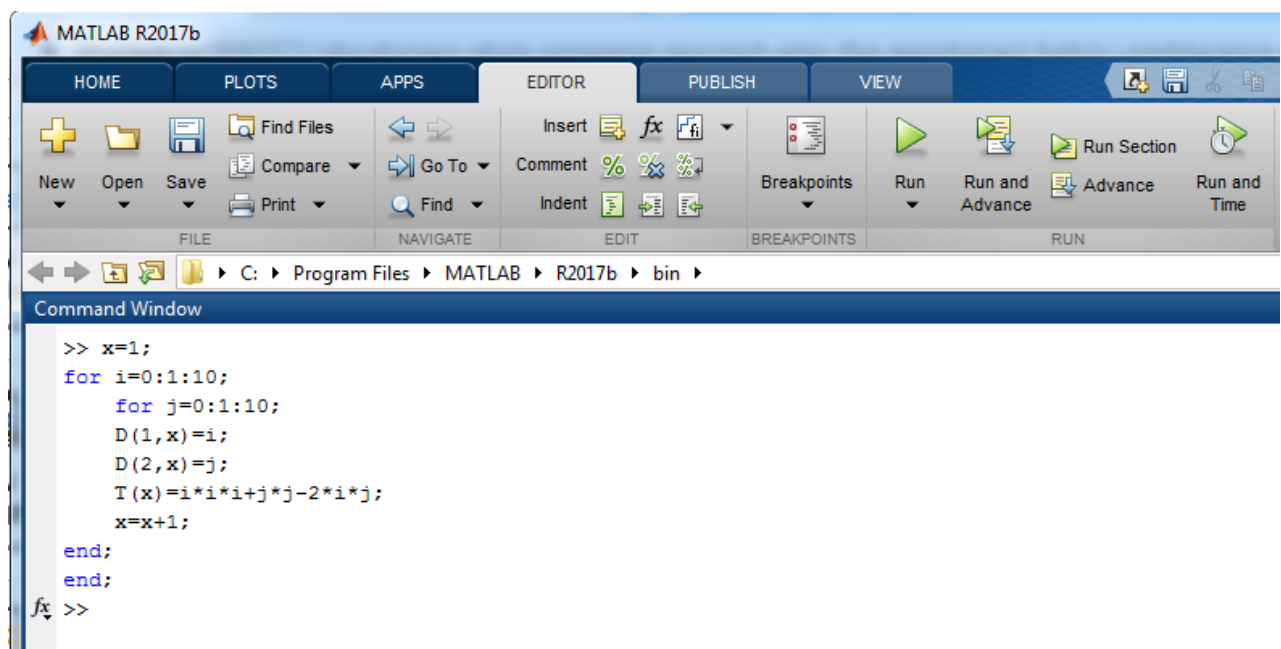


Рисунок 6.2 – Фрагмент програмного коду до прикладу № 1

1.2. Для створення нейронної мережі апроксимації даних необхідно запустити інструментальний засіб *Neural Network Tools Network / Data Manager* шляхом введення команди *nnstart*.

1.3. Виконується імпорт даних (D – вектор вхідних даних, T – вектор цілей), як показано на рисунку 6.3.

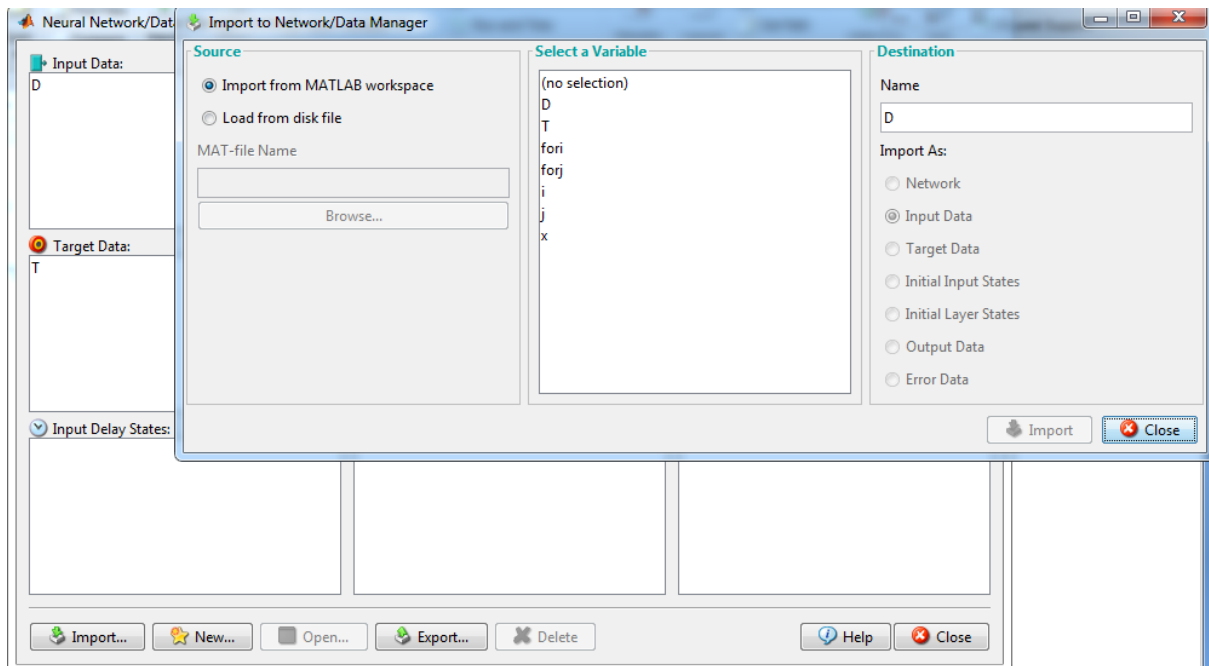


Рисунок 6.3 – Вікно імпорту даних

1.4. Виконується створення нової нейронної мережі та задаються її налаштування (D – вектор вхідних даних, T – вектор цілей, кількість нейронів у першому шарі – 5, у другому – 1, функція активації нейронів першого шару – *TANSIG*, другого – *PURELIN*), як показано на рисунку 6.4.

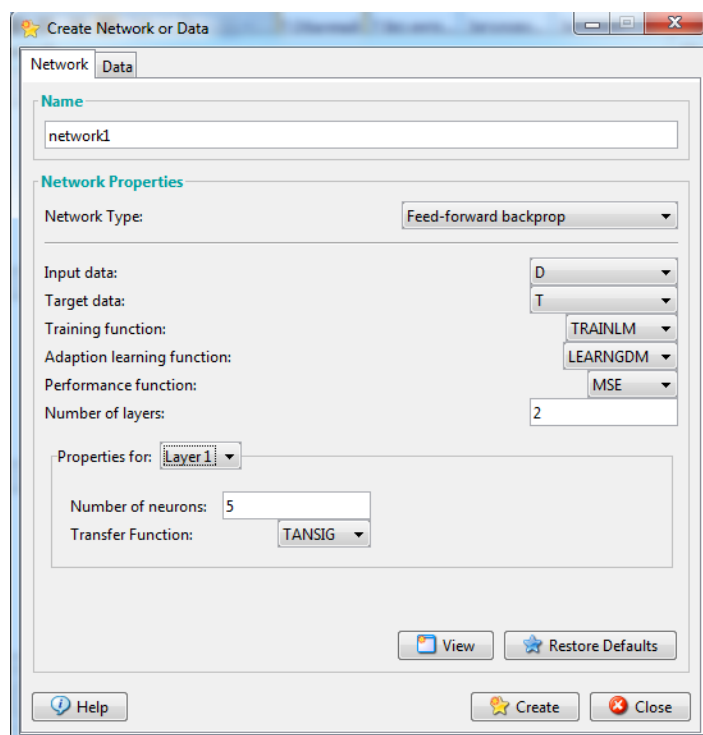


Рисунок 6.4 – Налаштування нейронної мережі

1.5. Створену нейронну мережу потрібно «навчити» на тестовій вибірці, як показано на рисунку 6.5.

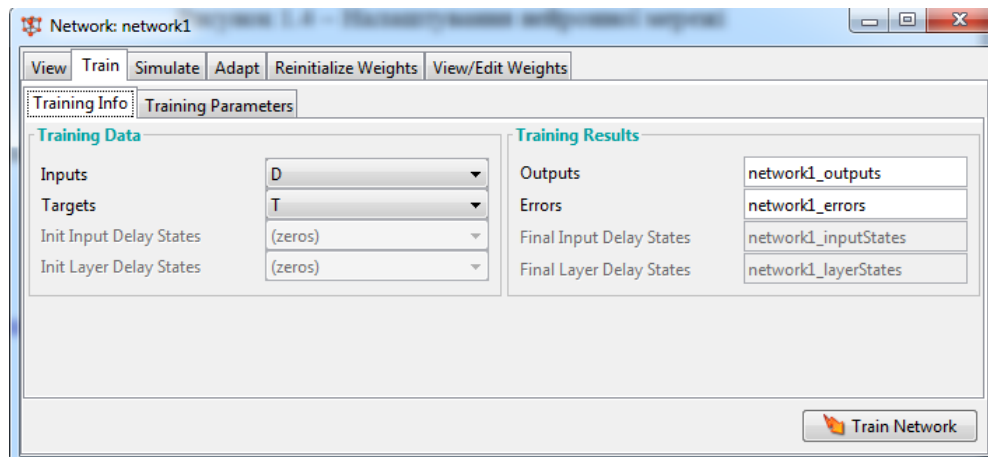


Рисунок 6.5 – Вікно налаштувань процедури «навчання» мережі

1.6. Отримані результати «навчання» створеної нейронної мережі для апроксимації даних наведено на рисунках 6.6 – 6.9.

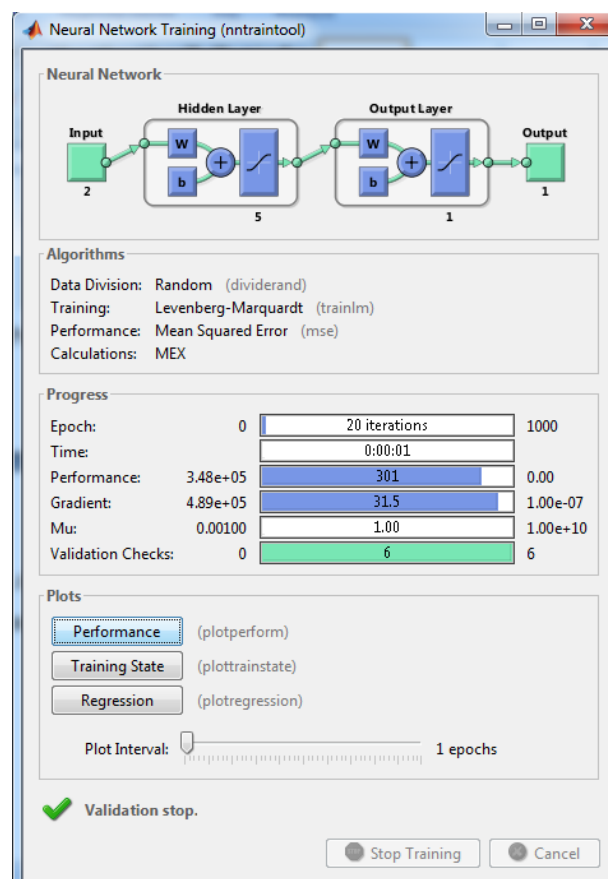


Рисунок 6.6 – Вікно результатів «навчання» мережі

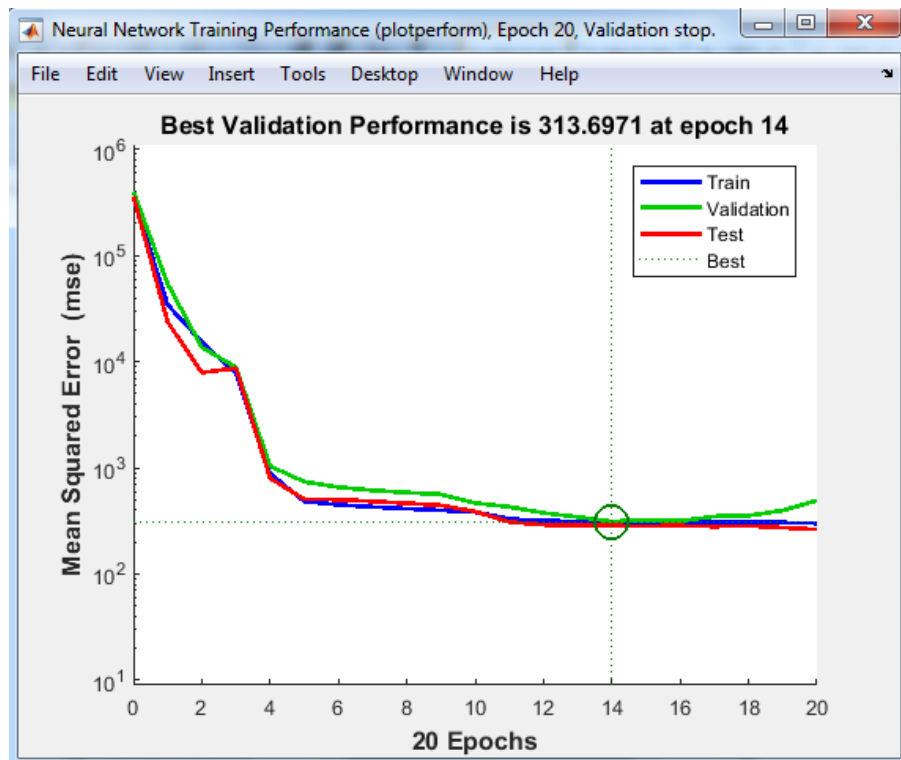


Рисунок 6.7 – Динаміка помилки «навчання» мережі

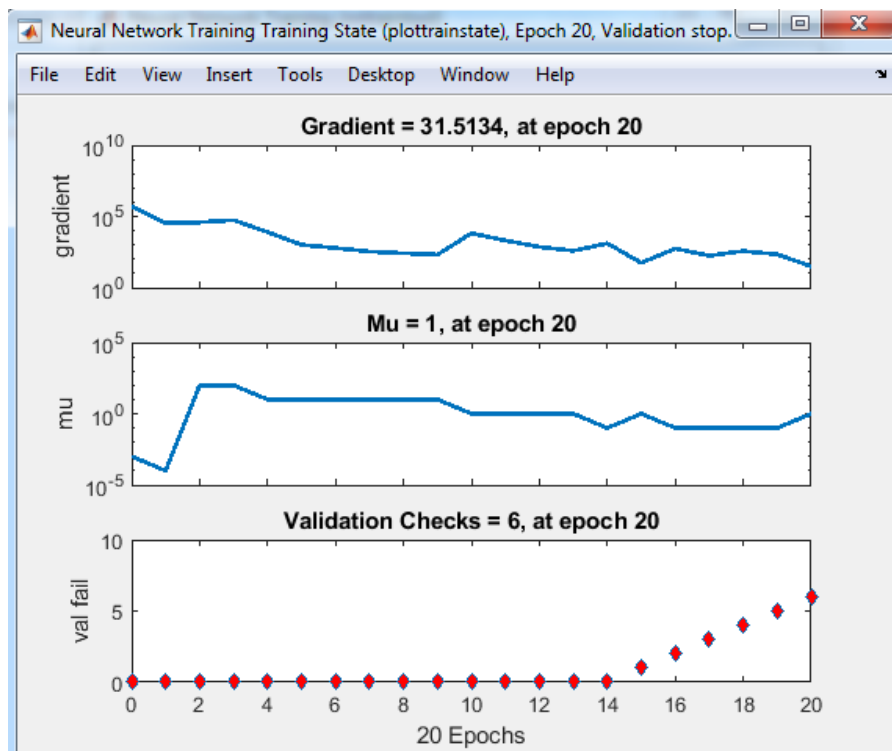


Рисунок 6.8 – Динаміка стану «навчання» мережі

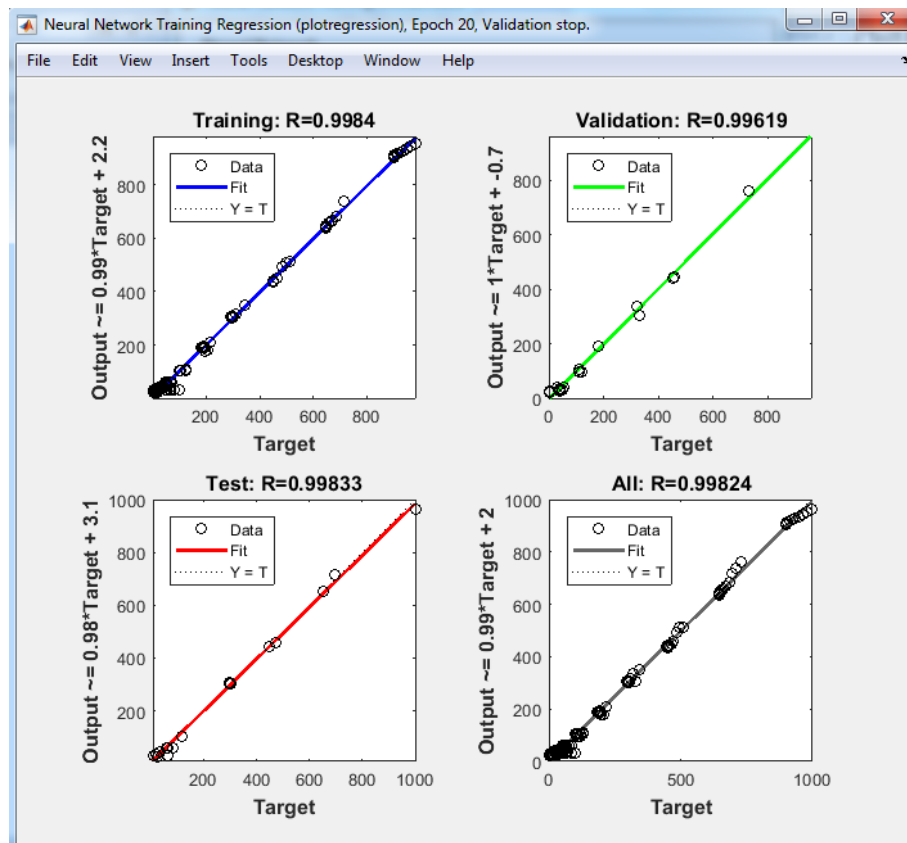


Рисунок 6.9 – Результати регресійного аналізу «навчання» мережі

1.7. Навчена мережа симулюється вектором даних, у результаті чого отримуємо вихід мережі *network1_outputs* на основі вхідних даних матриці *D*, як показано на рисунку 6.10.

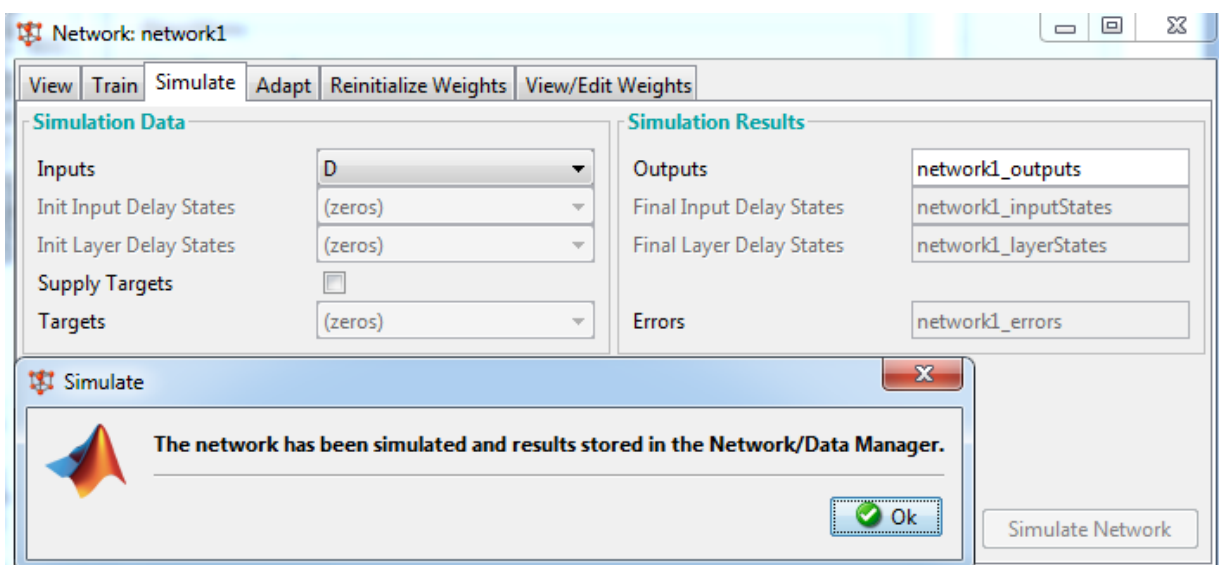


Рисунок 6.10 – Вікно симуляції нейронної мережі

1.8. Виконується експорт отриманих даних у робочу область Matlab, як показано на рисунку 6.11.

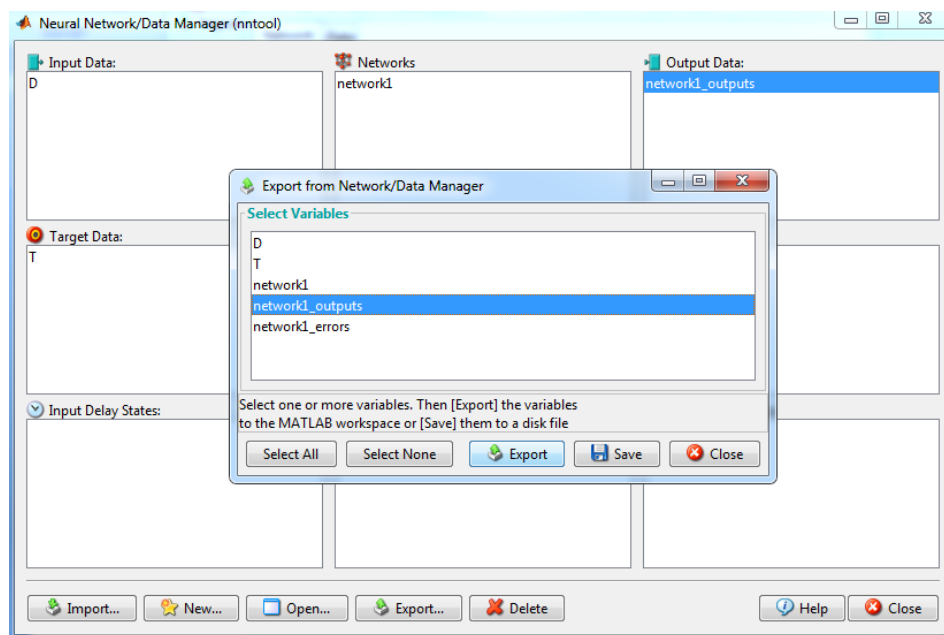


Рисунок 6.11 – Вікно експорту отриманих даних

1.9. Знаходиться абсолютна похибка апроксимації, як різниця виходу навченої мережі при симуляції *network1_outputs* та цілі *T* в робочій області, як показано на рисунку 6.12.

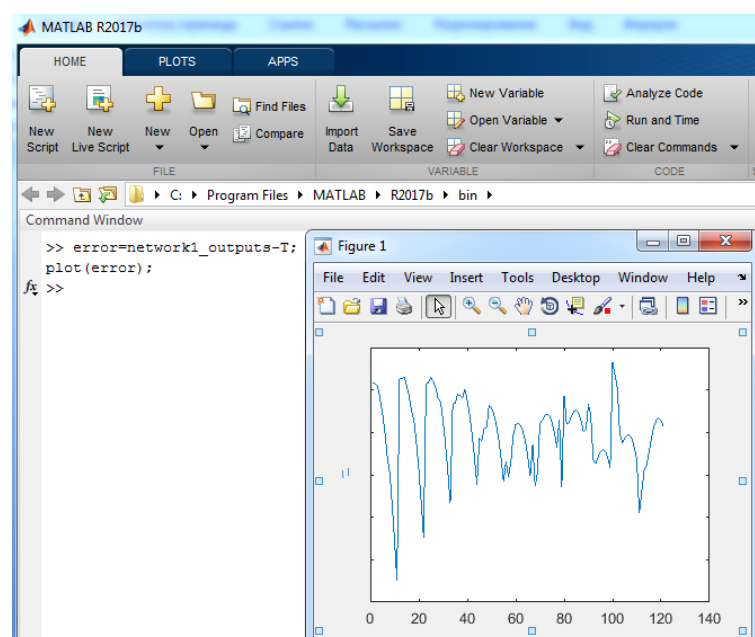


Рисунок 6.12 – Графік зміни похибки апроксимації

Приклад № 2. Апроксимація даних нелінійною залежністю

виду $y = \cos\left(\frac{3 \cdot \pi \cdot x}{N}\right) + \sin\left(\frac{5 \cdot \pi \cdot x}{N}\right)$ у діапазоні зміни вхідного параметру (x) від 0 до 200.

2.1. Порядок розв'язання цієї задачі є аналогічним до того, який наведено в прикладі № 1. Основні етапи розв'язку наведено на рисунках 6.13 – 6.18.

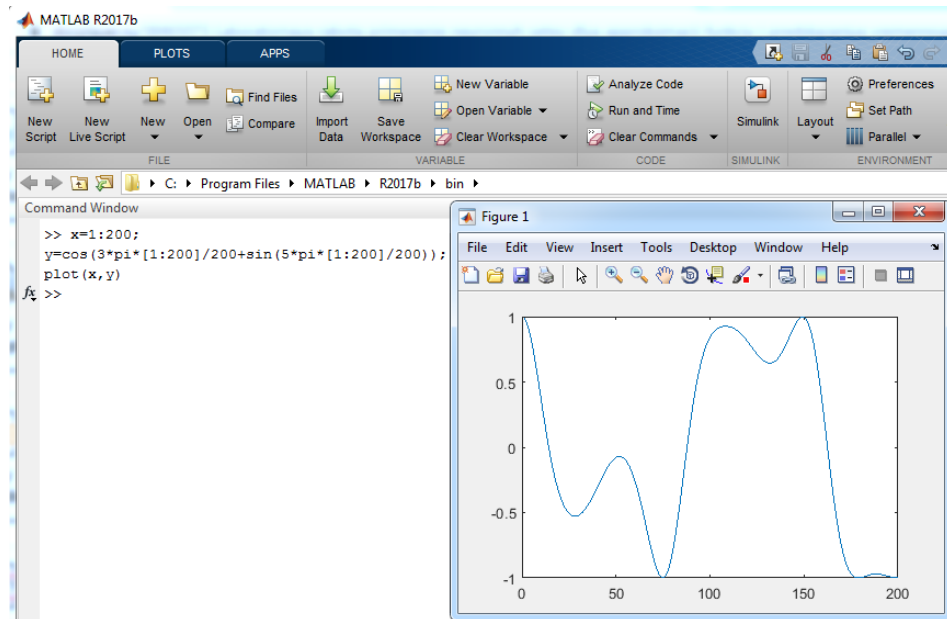


Рисунок 6.13 – Фрагмент програмного коду до прикладу № 2

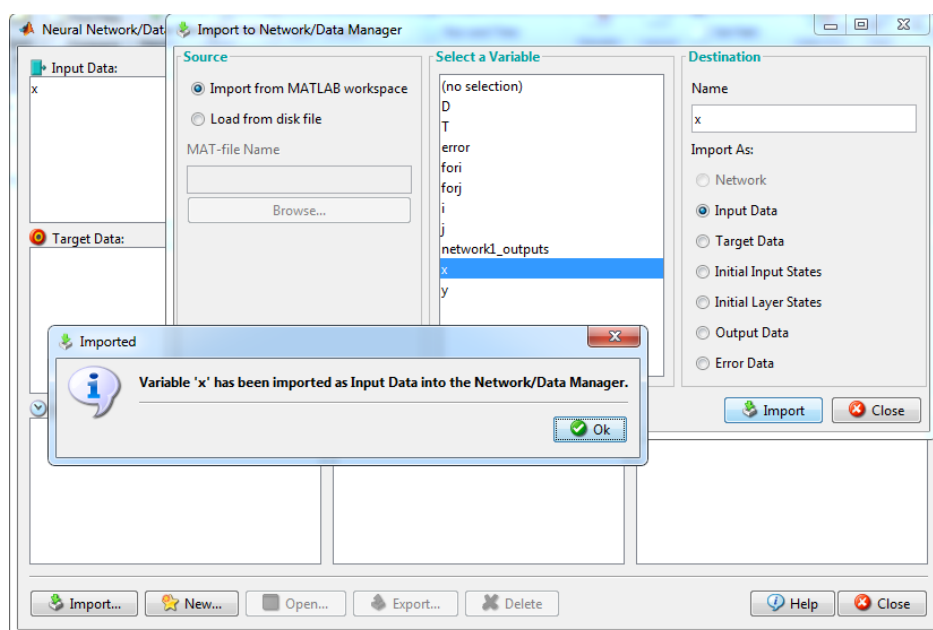


Рисунок 6.14 – Вікно імпорту даних

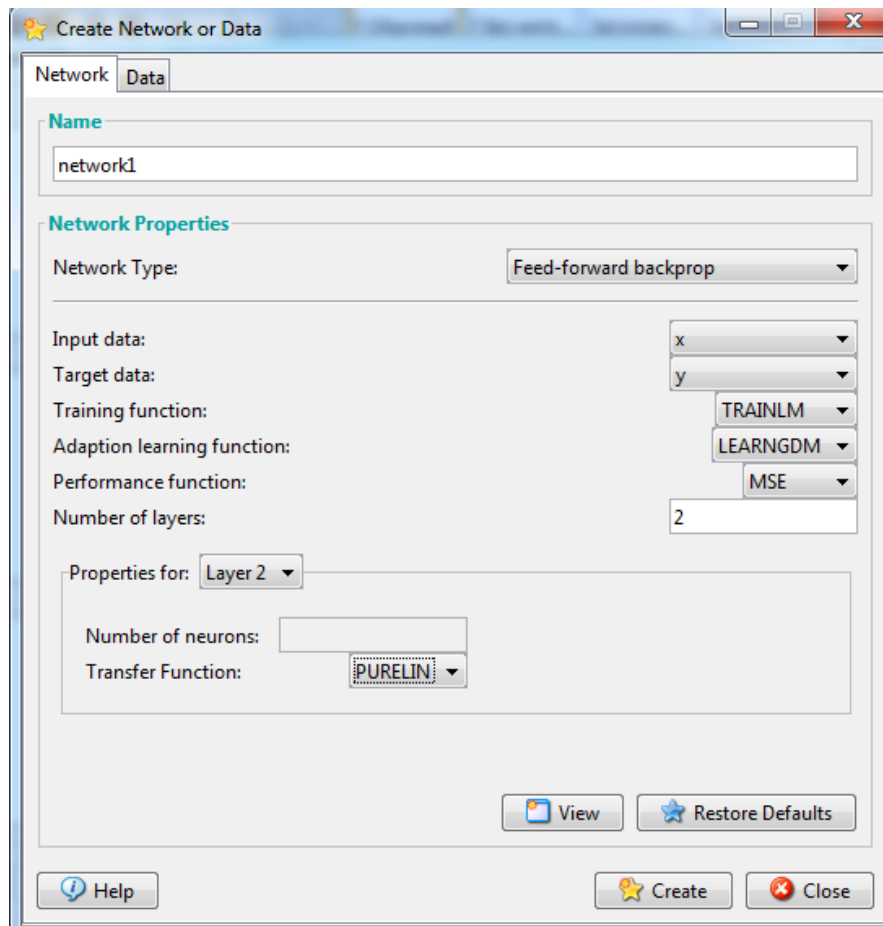


Рисунок 6.15 – Вікно налаштування нейронної мережі

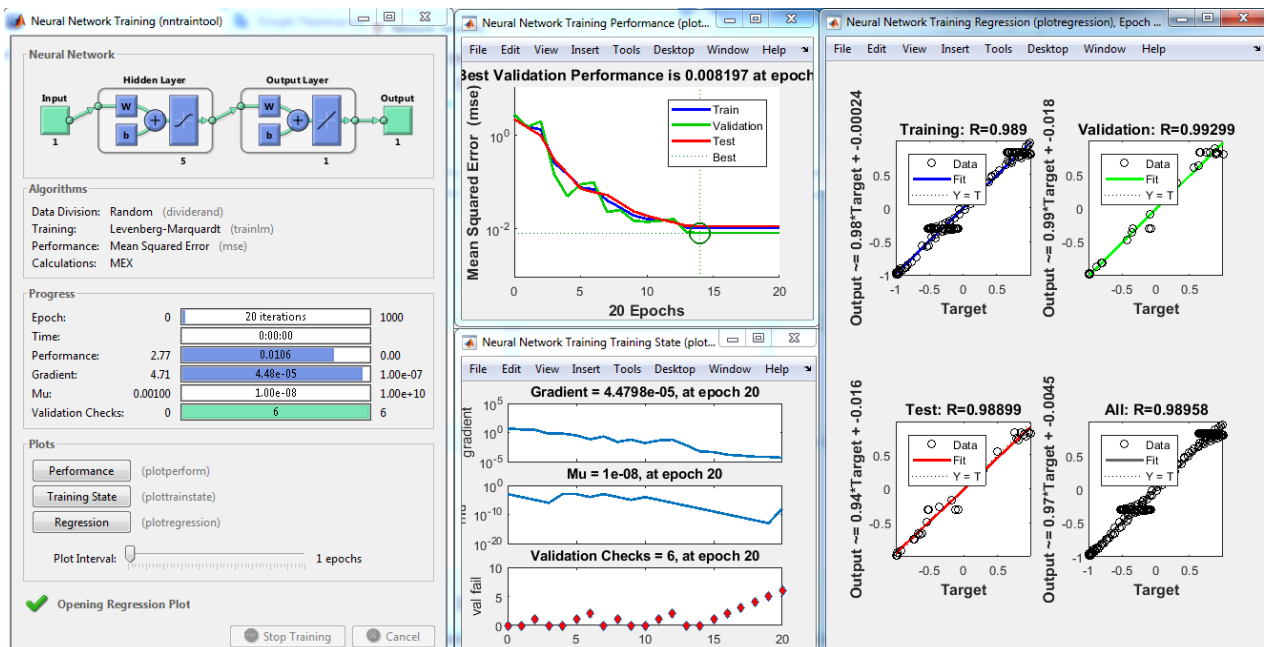


Рисунок 6.16 – Вікно результатів «навчання» мережі

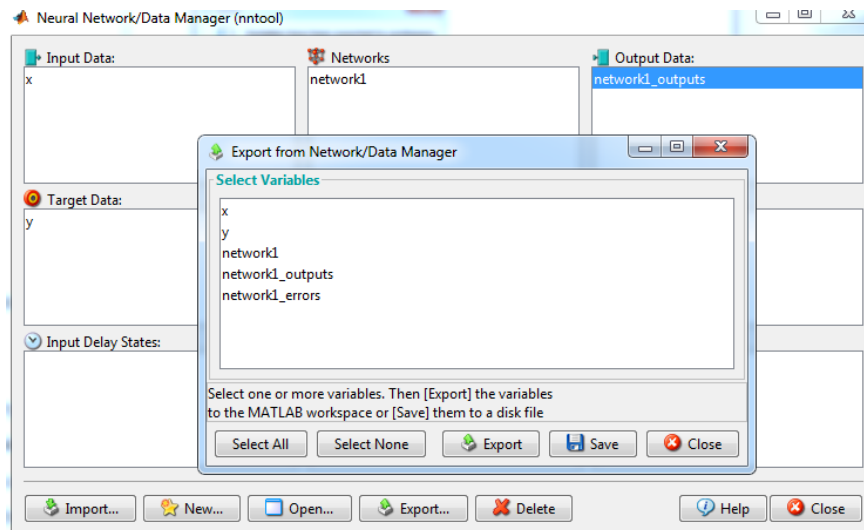


Рисунок 6.17 – Вікно експорту даних

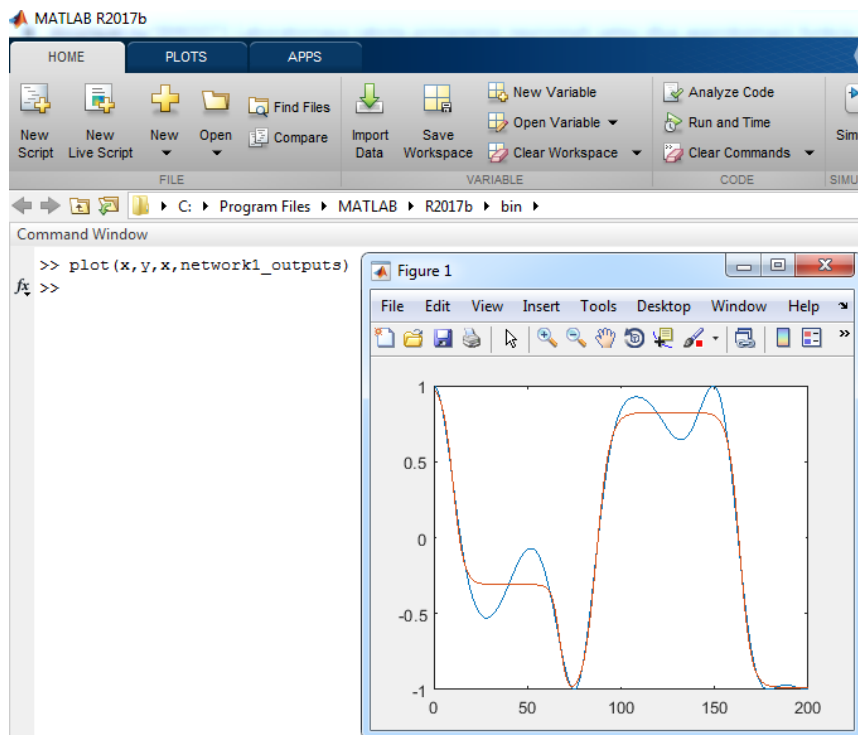


Рисунок 6.18 – Отримані результати апроксимації функції

Хід виконання роботи:

1. Створити та дослідити нейронну мережу, яка виконує функцію апроксимації експериментальних даних поліноміальною залежністю, згідно індивідуальних варіантів, які наведено в таблиці 6.2. Під час створення та дослідження нейронної мережі рекомендується використовувати алгоритм, який наведено в рішенні типового прикладу № 1.

Таблиця 6.2 – Варіанти завдань до п. 1 порядку виконання роботи

№ з/п	Функція	Інтервал значень x_1 та x_2
1	$y = 3 \cdot x_1^2 + 3 \cdot x_2^2 - 2 \cdot x_1 \cdot x_2$	від 2 до 8
2	$y = -2 \cdot x_1^2 - 3 \cdot x_2^2 + 2 \cdot x_1 \cdot x_2$	від 1 до 5
3	$y = 5 \cdot x_1^2 + 2 \cdot x_2^2 - 4 \cdot x_1 \cdot x_2$	від 2 до 6
4	$y = -4 \cdot x_1^2 + 3 \cdot x_2^2 - x_1 \cdot x_2$	від 0 до 7
5	$y = -x_1^2 + x_2^2 - 6 \cdot x_1 \cdot x_2$	від 3 до 9
6	$y = 2 \cdot x_1^2 + 3 \cdot x_2^2 - 7 \cdot x_1 \cdot x_2$	від 1 до 6
7	$y = -5 \cdot x_1^2 - 2 \cdot x_2^2 + 4 \cdot x_1 \cdot x_2$	від 2 до 8
8	$y = 4 \cdot x_1^2 - 3 \cdot x_2^2 - 2 \cdot x_1 \cdot x_2$	від 0 до 5
9	$y = -8 \cdot x_1^2 - 3 \cdot x_2^2 + 2 \cdot x_1 \cdot x_2$	від 1 до 6
10	$y = 5 \cdot x_1^2 + 2 \cdot x_2^2 - 9 \cdot x_1 \cdot x_2$	від 2 до 7

2. Створити та дослідити нейронну мережу, яка виконує функцію апроксимації експериментальних даних нелінійною залежністю, згідно індивідуальних варіантів, які наведено в таблиці 6.3. Під час створення та дослідження нейронної мережі рекомендується використовувати алгоритм, який наведено в рішення типowego прикладу № 2.

Таблиця 6.3 – Варіанти завдань до п. 2 порядку виконання роботи

№ з/п	Функція	Інтервал значень x_1 та x_2
1	$y = \cos\left(2 \cdot \pi \cdot \frac{x}{N}\right) + \sin\left(3 \cdot \pi \cdot \frac{x}{N}\right)$	від 0 до 100
2	$y = \cos\left(4 \cdot \pi \cdot \frac{x}{N}\right) + \sin\left(6 \cdot \pi \cdot \frac{x}{N}\right)$	від 0 до 200
3	$y = \cos\left(3 \cdot \pi \cdot \frac{x}{N}\right) + \sin\left(7 \cdot \pi \cdot \frac{x}{N}\right)$	від 0 до 150
4	$y = \sin\left(2 \cdot \pi \cdot \frac{x}{N}\right) + \sin\left(3 \cdot \pi \cdot \frac{x}{N}\right)$	від 0 до 120
5	$y = \cos\left(2 \cdot \pi \cdot \frac{x}{N}\right) + \cos\left(3 \cdot \pi \cdot \frac{x}{N}\right)$	від 0 до 100
6	$y = \sin\left(2 \cdot \pi \cdot \frac{x}{N}\right) + \cos\left(3 \cdot \pi \cdot \frac{x}{N}\right)$	від 0 до 200
7	$y = \sin\left(4 \cdot \pi \cdot \frac{x}{N}\right) + \cos\left(6 \cdot \pi \cdot \frac{x}{N}\right)$	від 0 до 150
8	$y = \sin\left(3 \cdot \pi \cdot \frac{x}{N}\right) + \cos\left(7 \cdot \pi \cdot \frac{x}{N}\right)$	від 0 до 120
9	$y = \sin\left(5 \cdot \pi \cdot \frac{x}{N}\right) + \sin\left(7 \cdot \pi \cdot \frac{x}{N}\right)$	від 0 до 200
10	$y = \cos\left(\pi \cdot \frac{x}{N}\right) + \cos\left(9 \cdot \pi \cdot \frac{x}{N}\right)$	від 0 до 150

Типовий зміст звіту до практичної роботи:

1. Титульний лист.
2. Мета та задачі практичної роботи.
3. Основні результати, отримані в роботі, з детальним описом відповідних дій за пунктами завдання.
4. Результати якісного й кількісного аналізу отриманих даних.
5. Висновки по роботі.

Контрольні питання:

1. Дайте визначення терміну «апроксимація функції».
2. У чому полягають основні етапи апроксимації функції?
3. Дайте визначення терміну «штучна нейронна мережа».
4. У чому полягає призначення функції `*network` та який її синтаксис?
5. У чому полягає призначення функції `*newscf` та який її синтаксис?
6. Наведіть алгоритм процедури навчання штучної нейронної мережі.
7. Яким чином знаходиться похибка апроксимації експериментальних даних?

Список рекомендованої літератури:

1. Ткаліченко С.В. Штучні нейронні мережі: Навчальний посібник. Кривий Ріг: Державний університет економіки і технологій, 2023. 150 с.
2. MatWorks. URL: <https://www.mathworks.com/products/simulink/getting-started.html> (дата звернення: 14.01.2024)
3. Лактіонов І.С. Обґрунтування моделі багатофакторної оцінки та прогнозування параметрів якості вирощування культур на захищеному ґрунті. Наук. пр. ДонНТУ. Сер.: «Обчислювальна техніка та автоматизація». Вип. 1 (30)' 2017. С. 29–37.
4. Laktionov I., Diachenko G., Rutkowska D., Kisiel-Dorohinicki M. An Explainable AI Approach to Agrotechnical Monitoring and Crop Diseases Prediction in Dnipro Region of Ukraine. JAISCR, 2023, Vol. 13 (4). P. 247–272.

ПРАКТИЧНА РОБОТА № 7

РОЗРОБКА ТА ДОСЛІДЖЕННЯ АЛГОРИТМІВ НЕЧІТКОЇ ТРАНСФОРМАЦІЇ ДАНИХ У КОМП'ЮТЕРИЗОВАНИХ СИСТЕМАХ МОНІТОРИНГУ ТА КЕРУВАННЯ ТЕХНОЛОГІЧНИМИ ПРОЦЕСАМИ

Мета роботи: отримати практичні навички розробки та дослідження моделей нечіткої трансформації даних у комп'ютеризованих системах моніторингу й керування технологічними процесами методами комп'ютерного моделювання.

Обладнання та програмне забезпечення: персональний комп'ютер, пакет прикладних програм Matlab & Simulink (студентська безкоштовна версія може бути завантажена за посиланням <https://www.mathworks.com/campaigns/products/trials.html>), MS Office.

Основні теоретичні відомості. Пакет Fuzzy Logic Toolbox – це сукупність прикладних програм, що дозволяють конструювати та моделювати нечіткі експертні системи та/або системи керування. До складу програмних засобів Fuzzy Logic Toolbox входять наступні підпрограми, що дозволяють працювати в режимі графічного інтерфейсу:

- редактор нечіткої системи виведення Fuzzy Inference System Editor (FIS-редактор);
- редактор функцій належності (Membership Function Editor);
- редактор правил (Rule Editor);
- програма перегляду правил (Rule Viewer);
- програма перегляду 3D-поверхні системи нечіткого виведення (Surface Viewer).

Редактор систем нечіткого виводу (FIS-редактор) є основним засобом, що використовується для створення та налаштування систем нечіткого виводу в графічному режимі. FIS-редактор може бути відкритий за допомогою введення

функції *fuzzy* у вікні команд Matlab.

Головне меню редактора містить наступні позиції:

File – робота з файлами моделей (створення, зберігання, зчитування та ін.), а саме:

- *New FIS...* – дозволяє обрати тип нової системи нечіткого керування;
- *Import* – дозволяє завантажувати в FIS-редактор існуючу систему нечіткого керування;

- *Export* – дозволяє зберегти відредаговану систему нечіткого керування.

Edit – операції редагування (додавання і виключення вхідних і вихідних змінних), а саме:

- *Undo* – скасовує виконання останньої дії;
- *Add Variable...* – дозволяє додати в редаговану систему нечіткого керування вхідну або вихідну змінну;
- *Remove Selected Variable* – видаляє обрану змінну з редагованої системи нечіткого виведення;

- *Membership Functions...* – викликає редактор функцій належності;

- *Rules* – викликає редактор правил нечіткого виводу.

View – перехід до додаткового інструментарію:

- *Rules* – викликає програму перегляду правил нечіткого виводу;

- *Surface* – викликає програму перегляду поверхні нечіткого виводу.

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

And method – дозволяє задати метод для виконання логічної кон'юнкції в умовах нечітких правил;

Or method – дозволяє задати метод для виконання логічної диз'юнкції в умовах нечітких правил;

Implication method – дозволяє задати метод для виконання (активації) логічного виведення в кожному з нечітких правил;

Aggregation method – дозволяє задати метод для агрегації значень функції належності кожної з вихідних змінних у виведеннях нечітких правил;

Defuzzification method – дозволяє задати метод для виконання дефазифікації вихідних змінних в системі нечіткого виведення.

Редактор функцій належності (Membership Function Editor) призначений для задавання та редагування функцій належності окремих термів системи нечіткого керування в графічному режимі. Програму-редактор функцій належності можна відкрити за допомогою пункту меню *View – Edit membership functions*. Для відображення графіків функцій належності слід обрати необхідну змінну в лівій частині графічного інтерфейсу редактора під заголовком FIS Variables.

Головне меню редактора містить наступні позиції:

File – містить такі ж операції, що і відповідний пункт меню редактора FIS.

Edit – містить наступні операції:

Undo – скасовує виконання останньої дії;

Add MF... – дозволяє додати вбудовану функцію належності термів для виділеної змінної;

Add Custom MF... – дозволяє додати функцію належності для окремої змінної;

Remove Current MF – дозволяє видалити окрему функцію належності;

Remove All MFs – дозволяє видалити всі функції належності для окремої змінної;

FIS Properties... – викликає редактор FIS;

Rules... – викликає редактор правил нечіткого виводу.

– *View* – містить наступні операції:

Rules – викликає підпрограму перегляду правил нечіткого виводу;

Surface – викликає підпрограму перегляду поверхні нечіткого виводу.

Редактор правил системи нечіткого виводу призначений для задавання та редагування окремих правил системи нечіткого керування в графічному режимі. Редактор правил може бути відкритий за допомогою головного меню редактора FIS за допомогою пункту меню *Edit – Rules*.

Головне меню редактора містить наступні позиції:

File – містить такі ж операції, що і відповідний пункт меню редактора FIS.

Edit – містить наступні операції:

Undo – скасовує виконання останньої дії;

FIS Properties... – викликає редактор FIS;

Membership Functions... – викликає редактор функцій належності.

View – містить наступні операції:

Rules – викликає програму перегляду правил;

Surface – викликає програму перегляду поверхні виводу.

Options – містить наступні операції:

Language – дозволяє обрати мову для запису правил у формі тексту;

Format – формат запису правил системи нечіткого виводу.

Програма перегляду поверхні системи нечіткого виводу дозволяє переглядати поверхню системи нечіткого керування та візуалізувати графіки залежності вихідних змінних від окремих вхідних змінних. Графічний інтерфейс програми перегляду правил може бути відкритий за допомогою головного меню редактора FIS за допомогою пункту меню *View – Surface*.

Головне меню редактора містить наступні позиції:

File – містить такі ж операції, що і відповідний пункт меню редактора FIS.

Edit – містить наступні операції:

Undo – скасовує виконання останньої дії;

FIS Properties... – викликає редактор FIS;

Membership Functions... – викликає редактор функцій належності;

Rules... – викликає програму редагування правил.

View – містить наступну операцію:

Rules – викликає програму перегляду правил.

Options – містить наступні операції:

Plot – дозволяє обрати один із 8 стилів зображення графіка поверхні

виводу;

Color Map – дозволяє обрати одну з 4 кольірних схем зображення графіка поверхні виводу;

Always evaluate – позначка галочкою цього пункту вкладеного меню призводить до автоматичного формування нової поверхні виведення щоразу, коли вносяться зміни в систему нечіткого виводу, що впливають на форму графіка поверхні виведення.

Типовий приклад: розробити та дослідити систему нечіткого керування крапельним зрошення тепличних культур. В якості входних параметрів прийнята температура ґрунту (T_s) в нормальному діапазоні зміни від 17 до 24 °C та вологість ґрунту в нормальному діапазоні 65 до 80 % (W_s). Функція керування системою крапельного зрошення має бути реалізована, як двохпозиційне керування (ON/OFF).

1. Формалізація задачі нечіткого керування в табличному вигляді, як наведено в таблиці 7.1.

Таблиця 7.1 – Матриця відповідності до функцій належності параметрів підсистеми керування

W_s	T_s		
	Нижче норми	Норма	Вище норми
Нижче норми	ON	ON	ON
Норма	OFF	OFF	ON
Вище норми	OFF	OFF	OFF

2. Створення системи нечіткого керування у FIS-редакторі, як показано на рисунку 7.1. Командою *fuzzy* запускаємо FIS-редактор.

Стандартний вихідний алгоритм виведення – Mamdani (про що говорить напис в центральному блоці). Так як в системі повинно бути два входи, через пункт меню *Edit – Add Variable – Input* додаємо в систему другий вхід. Далі задамо нові імена блокам (T_s та W_s) і створюваній системі. Через пункт меню *File – Export – To Workspace...* присвоюємо ім'я всій системі (*Poliv*).

2. Введення та редагування функцій належності вхідних змінних, як показано на рисунку 7.2. Для цього необхідно відкрити редактор функцій належності. В полях *Range* та *Display Range* встановлюються діапазони зміни і відображення цієї змінної. Через пункт меню *Edit – Remov All MF* видаляються всі функції належності, які задані стандартно, а за допомогою команди *Edit – Add MF* задаються нові функції належності із загальним числом 3 (за кількістю якісних значень змінної, як показано в таблиці 7.1).

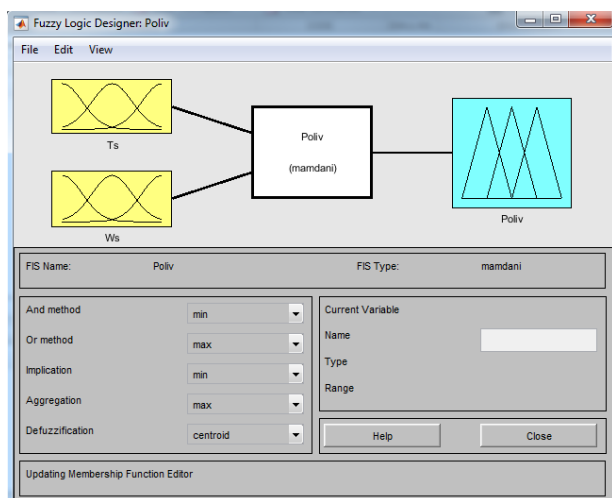


Рисунок 7.1 – Вікно FIS-редактора

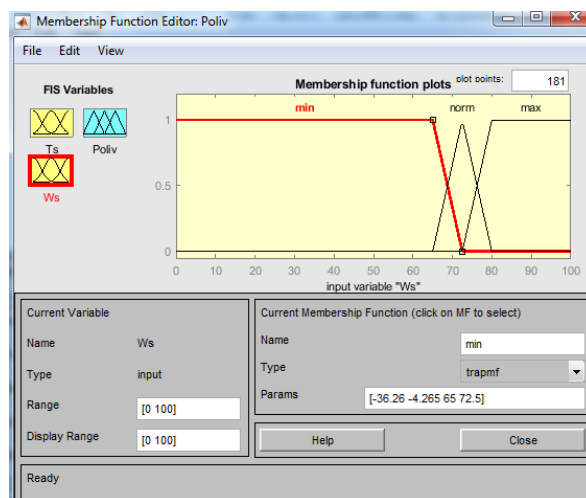


Рисунок 7.2 – Редактор вхідних змінних

3. Аналогічно до п. 2 виконується редагування вихідної змінної, відповідно до даних наведених у таблиці 7.1, як показано на рисунку 7.3.

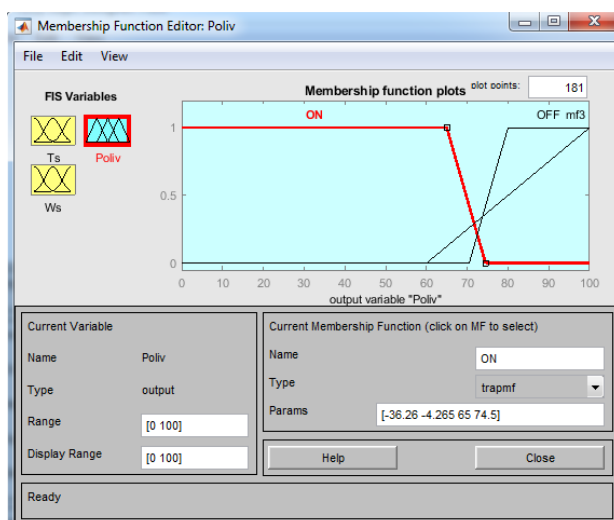


Рисунок 7.3 – Вікно редактора вихідної змінної

3. Конструювання правил за допомогою редактора правил системи нечіткого керування із використанням команди *Edit – Rules...* з вікна FIS-редактора. Підсумковий набір правил для даної системи формується на підставі матриці відповідності до функцій належності параметрів підсистеми керування (див. табл. 7.1) та має вигляд, як показано на рисунку 7.4.

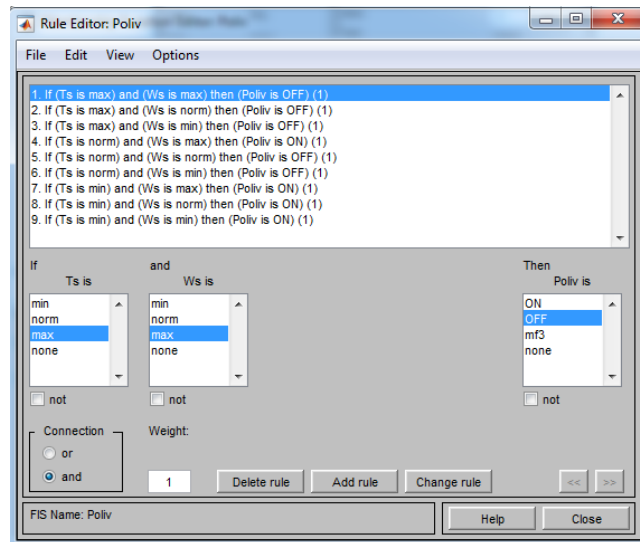


Рисунок 7.4 – Вікно редактора правил системи нечіткого керування

4. Моделювання роботи системи нечіткого керування. Для перевірки роботи створеної системи потрібно відкрити через пункт меню *View – Rules...* Отримані результати моделювання наведено на рисунку 7.5.

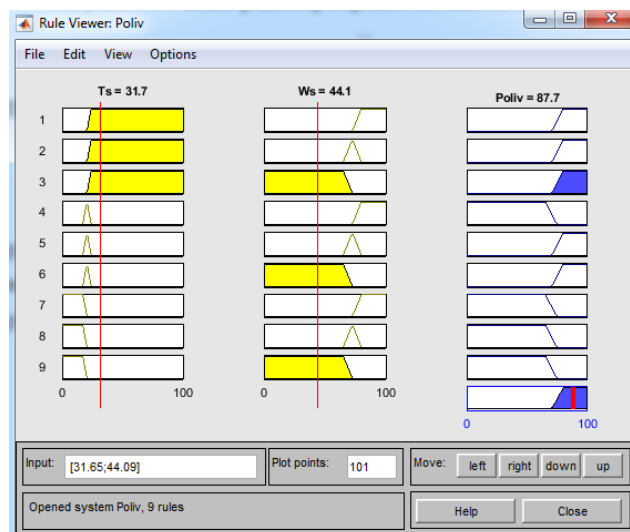


Рисунок 7.5 – Вікно графічного відображення результатів моделювання

5. Підтвердженням зазначеної залежності вихідної змінної від вхідних за допомогою графічного відображення поверхні відгуку, який формується за допомогою пункту меню *View – Surface*, як показано на рисунку 7.6.

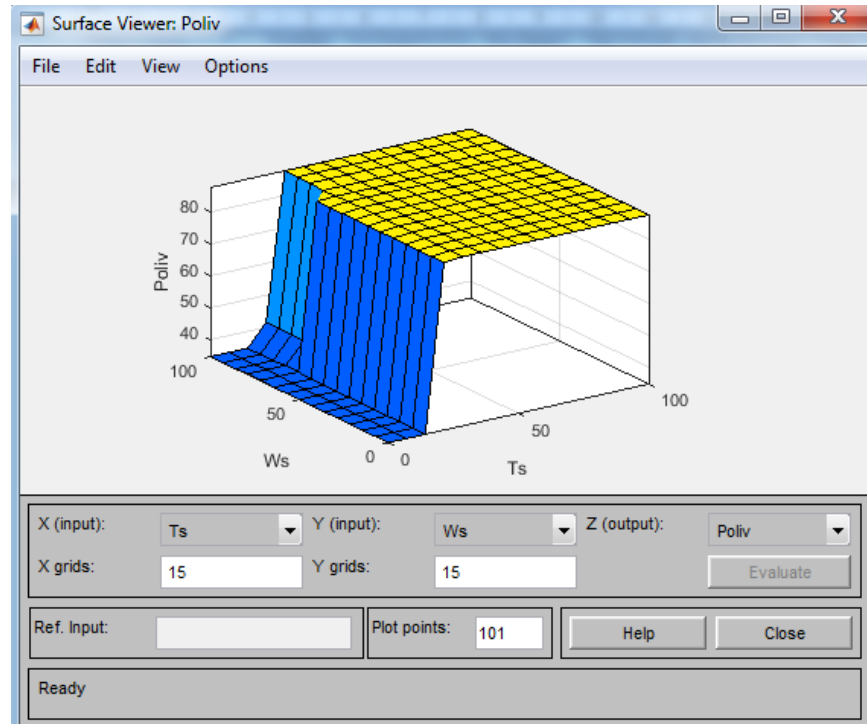


Рисунок 7.6 – Вікно графічного відображення поверхні відгуку системи

Хід виконання роботи:

1. Створити систему нечіткого керування мікрокліматом технологічних приміщень. В якості вхідних параметрів прийняти температура (T) та відносну вологість повітря (W). Робочі діапазони регулювання вхідних змінних прийняти відповідно до таблиці 7.2. Функція керування системою керування мікрокліматом має бути реалізована, як двохпозиційне керування (ON/OFF). Вихідний алгоритм виведення – Mamdani.

2. Виконати редагування створеної системи нечіткого керування у відповідності до індивідуальних даних, які наведено в таблиці 7.2.

3. Синтезувати правила керування за допомогою редактора правил системи нечіткого керування у відповідності до кількості піддіапазонів, які наведено в таблиці 7.2.

Таблиця 7.2 – Варіанти завдань до порядку виконання роботи

№ з/п	T, °C	W, %	Кількість піддіапазонів (T)	Кількість піддіапазонів (W)	Тип вхідних функцій	Тип вихідної функції
1	10 – 20	60 – 80	2	3	trapmf	pimf
2	12 – 18	50 – 70	3	2	pimf	trapmf
3	14 – 22	55 – 75	2	3	trapmf	pimf
4	16 – 24	65 – 85	3	2	pimf	trapmf
5	12 – 20	60 – 70	2	3	trapmf	pimf
6	12 – 20	60 – 80	3	2	pimf	trapmf
7	16 – 24	50 – 70	2	3	trapmf	pimf
8	14 – 22	55 – 75	3	2	pimf	trapmf
9	12 – 18	65 – 85	2	3	trapmf	pimf
10	10 – 20	60 – 70	3	2	pimf	trapmf

4. Провести дослідження синтезованої системи нечіткого керування методами імітаційного моделювання у відповідності до п. 4 та 5 типового прикладу даної роботи.

Типовий зміст звіту до практичної роботи:

1. Титульний лист.
2. Мета та задачі практичної роботи.
3. Основні результати, отримані в роботі, з детальним описом відповідних дій за пунктами завдання.
4. Результати якісного й кількісного аналізу отриманих даних.
5. Висновки по роботі.

Контрольні питання:

1. Дайте визначення терміну «нечітка система».
2. У чому полягає основний функціонал редактора нечіткої системи виведення?
3. У чому полягає основний функціонал редактора функцій належності?

4. У чому полягає основний функціонал редактора правил?
5. У чому полягає основний функціонал програми перегляду правил?
6. У чому полягає основний функціонал програми перегляду 3D-поверхні системи нечіткого виведення?
7. У чому полягають основні етапи створення та налаштування нечітких систем із використанням Fuzzy Logic Toolbox?
8. У чому полягають відмінності в алгоритмах Mamdani та Sugeno?

Список рекомендованої літератури:

1. Зайченко, Ю. П. Нечіткі моделі й методи в інтелектуальних системах / Ю. П. Зайченко. – К.: Видавничий дім «Слово», 2008. 344 с.
2. MathWorks: Design and simulate fuzzy logic systems. URL: www.mathworks.com/products/fuzzy-logic.html (дата звернення 18.01.2024).
3. Laktionov I., Rutkowski L., Vovna O., Byrski A., Kabanets M. A novel approach to intelligent monitoring of gas composition and light mode of greenhouse crop growing zone on the basis of fuzzy modelling and human-in-the-loop techniques. Engineering Applications of Artificial Intelligence, 2023, Vol. 126 (Part B), P. 1–21. <https://doi.org/10.1016/j.engappai.2023.106938>.
4. Laktionov I., Vovna O., Kabanets M. Computer-Oriented Method of Adaptive Monitoring and Control of Temperature and Humidity Mode of Greenhouse Production. Baltic J. Modern Computing, 2023, Vol. 11 (1). P. 202–225.
5. Laktionov I., Vovna O., Kabanets M. Information Technology for Comprehensive Monitoring and Control of the Microclimate in Industrial Greenhouses Based on Fuzzy Logic. Journal of Artificial Intelligence and Soft Computing Research, 2023, Vol. 13 (1). P. 19–35.

ПРАКТИЧНА РОБОТА № 8

РОЗРОБКА ТА ДОСЛІДЖЕННЯ ГІБРИДНИХ АЛГОРИТМІВ НЕЙРОНЕЧІТКОЇ ТРАНСФОРМАЦІЇ ДАНИХ У КОМП'ЮТЕРИЗОВАНИХ СИСТЕМАХ МОНІТОРИНГУ ТА КЕРУВАННЯ ТЕХНОЛОГІЧНИМИ ПРОЦЕСАМИ

Мета роботи: отримати практичні навички розробки та дослідження гібридних алгоритмів нейронечіткої трансформації даних у інтелектуалізованих інформаційних технологіях моніторингу й керування технологічними процесами методами комп'ютерного моделювання.

Обладнання та програмне забезпечення: персональний комп'ютер, пакет прикладних програм Matlab & Simulink (студентська безкоштовна версія може бути завантажена за посиланням <https://www.mathworks.com/campaigns/products/trials.html>), MS Office.

Основні теоретичні відомості. Необхідність у розробці та дослідженні засобів, які забезпечують можливість налаштування й оптимізації нечітких моделей керування, виникає на стадії практичного застосування нечітких систем. Характеристики точності та якості нечіткої моделі мають сильний корелятивний зв'язок із формою та взаємним розташуванням функцій належності. Налаштування параметрів нечіткої моделі в значній мірі автоматизується шляхом використання нейромережових нечітких систем, у яких нейронна (адаптивна) мережа виступає в якості механізму налаштування параметрів нечіткої моделі. Множина параметрів, що налаштовується визначається параметрами функцій належності умовної частини правил та лінійними коефіцієнтами у вихідній частині системи. Наприклад:

Z_1 : Якщо $X_1=B_{11}$ та $X_2=B_{12}$ та ... та $X_n=B_{1n}$, то $Y=D_1$;

Z_2 : Якщо $X_1=B_{21}$ та $X_2=B_{22}$ та ... та $X_n=B_{2n}$, то $Y=D_2$;

...

Z_m : Якщо $X_1=B_{m1}$ та $X_2=B_{m2}$ та ... та $X_n=B_{mn}$, то $Y=D_m$.

Нечіткі множини є параметрами нечіткої моделі, а структура правил та їх кількість m визначають структуру моделі. В процесі нечіткого моделювання можливе використання нейронних мереж для проведення структурної й параметричної ідентифікації нечіткої моделі.

Під час розробки та досліджень у даній практичній роботі розглядається двоетапна процедура побудови нечітких моделей типу Sugeno. На першому етапі синтезується множина нечітких правил з вибірки експериментальних даних із використання субтрактивної кластеризації. На другому етапі виконується налаштування параметрів нечіткої моделі за допомогою ANFIS-алгоритму.

Субтрактивна кластеризація може використовуватися в якості швидкого автономного методу синтезу нечітких правил із вибірки експериментальних даних. Крім того, цей метод може розглядатися як свого роду препроцесінг для ANFIS-алгоритму – синтезована нечітка модель є початковою точкою для процедури навчання. Важливою перевагою застосування кластеризації для синтезу нечіткої моделі є те, що правила бази знань виходять об'єктно-орієнтованими.

Типовий приклад: на базі заданої навчальної вибірки експериментальних даних за допомогою редактора нейро-нечіткого виводу ANFIS системи Matlab розробити систему нечіткого керування технологічним процесом, що визначає залежність вихідного параметра від вхідної вибірки даних.

1. Генерація вибірки результатів спостережень, які відповідають нормальному закону розподілу.

2. Активізація ANFIS-редактора у пакеті Matlab за допомогою команди *anfisedit*.

3. Завантаження даних результатів спостережень до ANFIS-редактора за допомогою команди *Load Data* (формат файлу для зчитування .dat), як показано на рисунку 8.1.

4. Генерація структури системи нечіткого виводу типу Сугено за допомогою опції *Sub.clustering* та *Generate Fis*.

5. Налаштування параметрів системи за допомогою діалогового вікна властивостей, як показано на рисунку 8.2.

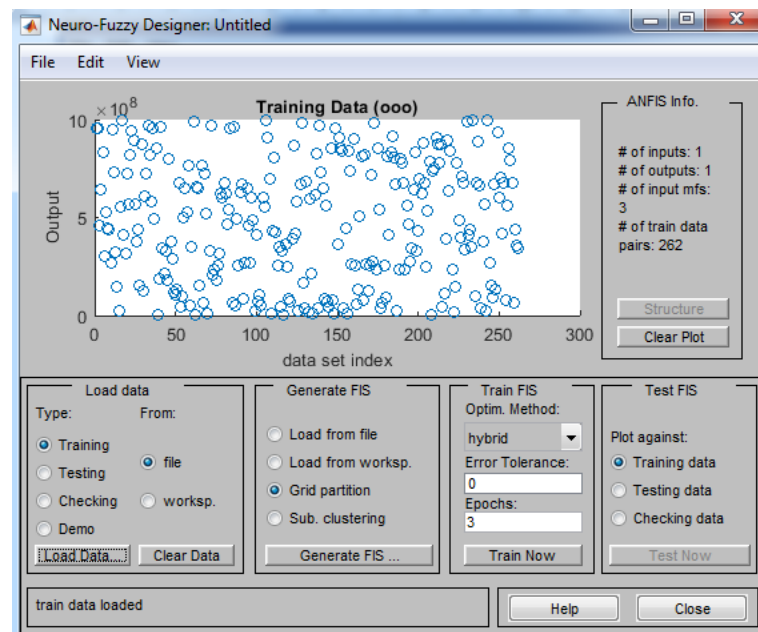


Рисунок 8.1 – Вікно завантаження навчальної вибірки до редактор ANFIS

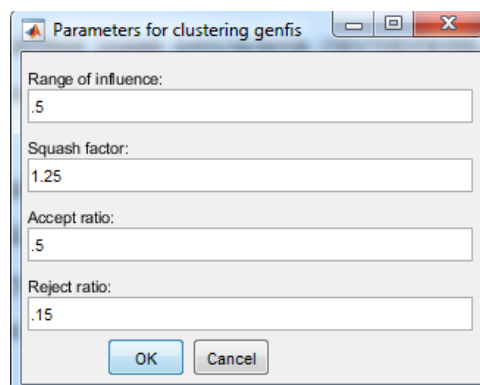


Рисунок 8.2 – Вікно налаштування параметрів моделі

6. Запуск процесу навчання гібридної моделі шляхом використання опції *Train Now*. У результаті на екрані з'явиться графік динаміки помилки навчання,

який показує яким чином проходить процедура навчання, як показано на рисунку 8.3.

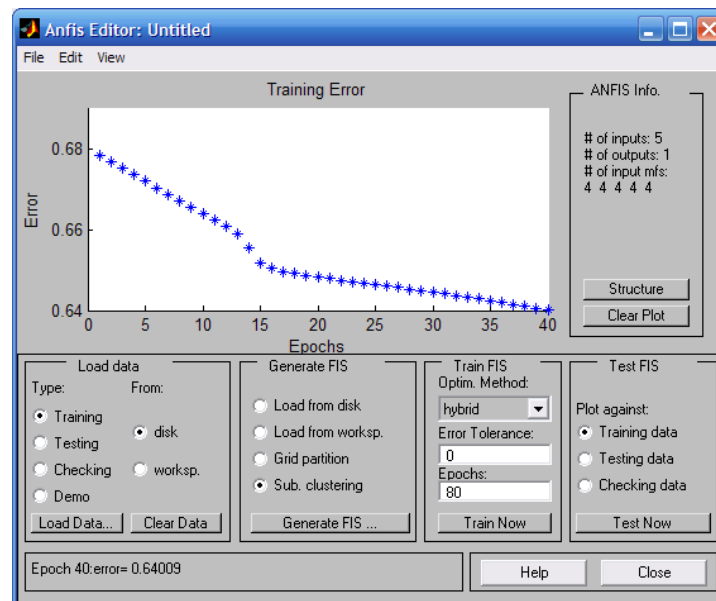


Рисунок 8.3 – Графік зміни помилки навчання гібридної моделі

7. Графічна візуалізація та аналіз отриманих результатів розробки гібридної мережі: структура мережі – опція *Structure* (див. рис. 8.4), база правил – опція *Edit – Rules...* (див. рис. 8.5), графік залежності цільової функції від вхідних параметрів – опція *View – Surface*.

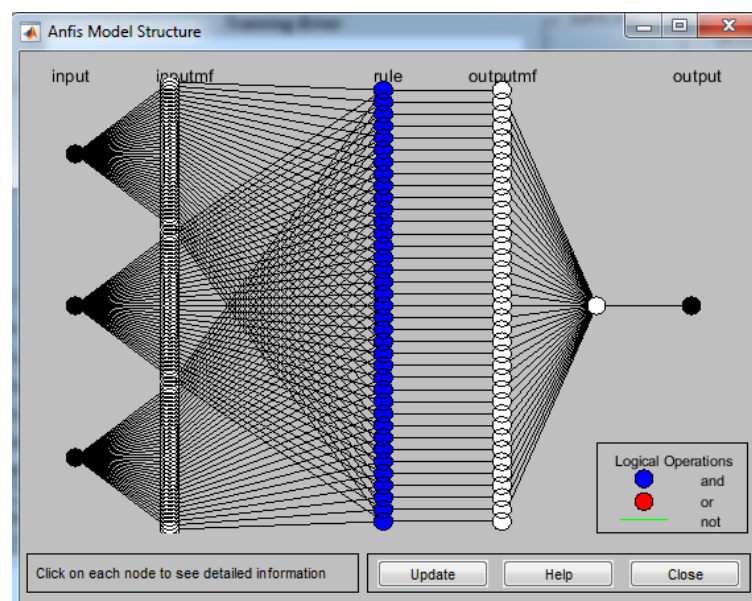


Рисунок 8.4 – Структура моделі гібридної системи

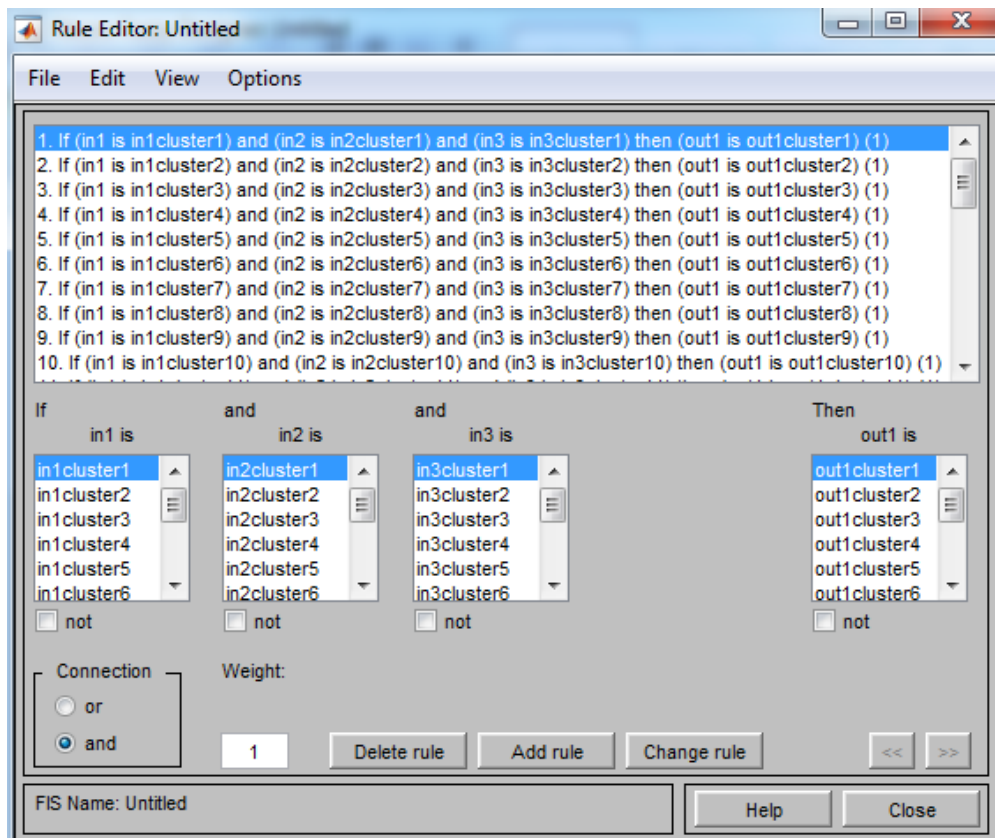


Рисунок 8.5 – Вікно бази правил моделі гібридної системи

Хід виконання роботи:

1. Згенерувати у програмі MS Excel файл вибірки випадкових даних, який містить кількість стовпчиків відповідно до кількості вхідних змінних з таблиці 8.1. Кількість рядків у кожному стовпці прийняти рівною відповідно до таблиці 8.1.
2. Запустити ANFIS-редактор.
3. Завантажити дані до ANFIS-редактора.
4. Згенерувати структуру гібридної системи нечіткого виводу типу Сугено.
5. Провести налаштування системи відповідно до п. 5 та 6 типового прикладу роботи.
6. Провести моделювання створеної гібридної системи відповідно до п. 7 типового прикладу роботи.
7. Провести кількісний та якісний аналіз отриманих результатів моделювання.

Таблиця 8.1 – Варіанти завдань до порядку виконання роботи

№ з/п	Кількість вхідних змінних	Кількість рядків даних
1	3	200
2	4	100
3	3	180
4	4	150
5	3	160
6	3	175
7	4	155
8	3	190
9	4	195
10	3	165

Типовий зміст звіту до практичної роботи:

1. Титульний лист.
2. Мета та задачі практичної роботи.
3. Основні результати, отримані в роботі, з детальним описом відповідних дій за пунктами завдання.
4. Результати якісного й кількісного аналізу отриманих даних.
5. Висновки по роботі.

Контрольні питання:

1. Дайте визначення терміну «гібридна нейронечітка модель».
2. За допомогою якої команди активізується ANFIS-редактор у пакеті Matlab?
3. У чому полягає основний функціонал ANFIS-редактора у середовищі Matlab & Simulink?
4. У чому полягає призначення опції *Structure*? Яким чином ця опція була використана в цій практичній роботі?
5. У чому полягає призначення опції *Edit – Rules...*? Яким чином ця опція була використана в цій практичній роботі?

6. У чому полягає призначення опції *View – Surface...*? Яким чином ця опція була використана в цій практичній роботі?

7. Який тип нечіткого виводу використовується під час створення нейронечітких моделей? У чому полягають його характерні особливості?

8. У чому полягає алгоритм створення нейронечіткої моделі в ANFIS-редакторі?

9. У чому полягає алгоритм налаштування та навчання нейронечіткої моделі в ANFIS-редакторі?

Список рекомендованої літератури:

1. MathWorks: Neuro-Adaptive Learning and ANFIS. URL: <https://www.mathworks.com/help/fuzzy/neuro-adaptive-learning-and-anfis.html> (дата звернення 21.01.2024).

2. Субботін С.О. Подання й обробка знань у системах штучного інтелекту та підтримки прийняття рішень: навчальний посібник. Запоріжжя: ЗНТУ, 2008. 341 с.

4. Laktionov I., Diachenko G., Rutkowska D., Kisiel-Dorohinicki M. An Explainable AI Approach to Agrotechnical Monitoring and Crop Diseases Prediction in Dnipro Region of Ukraine. *Journal of Artificial Intelligence and Soft Computing Research*, 2023, Vol. 13 (4). P. 247–272.

5. Олійник А.О., Субботін С.О., Олійник О.О. Інтелектуальний аналіз даних: навчальний посібник. Запоріжжя: ЗНТУ, 2012. 271 с.

КРИТЕРІЇ ОЦІНЮВАННЯ ПРАКТИЧНИХ РОБІТ

Навчальні досягнення здобувачів вищої освіти за результатами вивчення курсу оцінюватимуться за шкалою, що наведена нижче:

Рейтингова шкала	Інституційна шкала
90–100	відмінно
74–89	добре
60–73	задовільно
0–59	незадовільно

Здобувачі вищої освіти можуть отримати підсумкову оцінку з навчальної дисципліни на підставі поточного оцінювання знань за умови, якщо набрана кількість балів складатиме не менше як 60 балів.

Максимальне оцінювання:

Теоретична частина	Практична частина	Разом
36	64	100

Виконання практичних робіт № 1–4 передбачено в першому семестрі, практичних робіт № 5–8 – у другому семестрі. При своєчасному (протягом двох тижнів від дати проведення заняття) та правильному виконанні завдань відповідної практичної роботи в повному обсязі здобувач отримує оцінку 16 балів, із яких: 10 балів – виконання та оформлення звіту з практичної роботи; 6 балів – захист практичної роботи. За відсутності об'єктивних причин у разі несвоєчасного виконання практичної роботи передбачено -1 бал за кожен прострочений тиждень. У практичних роботах № 1–4 передбачено опціональні завдання (завдання 8 у практичних роботах № 1 і 2, завдання 7 у практичній роботі № 3, завдання 6 у практичній роботі № 4). Здобувач виконує ці опціональні завдання за власним бажанням, і результат їх виконання не впливає на підсумкову оцінку з відповідної практичної роботи.

Критерії оцінювання практичних робіт:

Бали за виконання та оформлення	
10	Роботу виконано в повному обсязі і правильно, результати описано і проаналізовано, звіт оформлено коректно
8–9	Роботу виконано переважно правильно, але наявні окремі неточності, які істотно не впливають на загальний результат, наявний аналіз одержаних результатів
6–7	Роботу виконано переважно правильно, але наявні окремі неточності, які не впливають істотно на загальний результат, відсутній аналіз одержаних результатів
4–5	Роботу виконано частково, наявні помилки під час виконання етапів роботи або недостатньо представлено результати перевірки
1–3	Виконано лише окремі етапи, звіт неповний
0	Роботу не виконано або поданий результат не відповідає завданню
Бали за захист	
6	Здобувач упевнено та повно пояснює виконані етапи роботи, обґрунтовує використані методи та отримані результати, правильно й аргументовано відповідає на запитання
4–5	Здобувач загалом правильно пояснює виконані етапи роботи та отримані результати, розуміє використані методи, але допускає окремі неточності або неповно відповідає на окремі запитання
2–3	Здобувач розуміє основний зміст роботи, але має труднощі з поясненням окремих етапів, методів або отриманих результатів, а також допускає принципові неточності у відповідях на запитання
1	Здобувач має загальне уявлення про виконану роботу, може частково пояснити її зміст або окремі етапи, але не може обґрунтувати отримані результати та відповісти на більшість запитань
0	Здобувач не може пояснити виконання роботи, не розуміє її основного змісту та не відповідає на основні запитання

Навчальне видання

Лактіонов Іван Сергійович

Каштан Віта Юріївна

Гнатушенко Володимир Володимирович

МЕТОДИ ТА ЗАСОБИ ПОБУДОВИ СИСТЕМ INDUSTRY 4.0

Методичні рекомендації до виконання практичних робіт

для здобувачів ступеня доктора філософії

спеціальності 123 Комп'ютерна інженерія

Видано в авторській редакції.

Електронний ресурс.

Підписано до видання 10.04.2024. Авт. арк. 2,2.

Національний технічний університет «Дніпровська політехніка».

49005, м. Дніпро, просп. Дмитра Яворницького, 19.