

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет інформаційно-аналітичних технологій та менеджменту

(повна назва)

Кафедра прикладної математики

(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти другий (магістерський)

Управління мобільним роботом

(тема)

Виконав:

студент 2 курсу, групи ПМм-22-1

Калініченко А.В.

(прізвище, ініціали)

Спеціальність 113 Прикладна математика

(код і повна назва спеціальності)

Тип програми освітньо-професійна

(освітньо-професійна або освітньо-наукова)

Освітня програма Прикладна математика

(повна назва освітньої програми)

Керівник доц. Матвієнко О.І.

(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри ПМ

(підпис)

Сидоров М.В.

(прізвище, ініціали)

2024 р.

Харківський національний університет радіоелектроніки

Факультет інформаційно-аналітичних технологій та менеджменту

Кафедра прикладної математики

Рівень вищої освіти другий (магістерський)

Спеціальність 113 Прикладна математика

(код і повна назва)

Тип програми освітньо-професійна

(освітньо-професійна або освітньо-наукова)

Освітня програма Прикладна математика

(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри ПМ _____

(підпис)

“06” листопада 2023 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові Калініченко Аліні Віталіївні

(прізвище, ім'я, по батькові)

1. Тема роботи Управління мобільним роботом

затверджена наказом по університету від 2 листопада 2023 р. № 1276 Ст

2. Термін подання студентом роботи до екзаменаційної комісії 10 січня 2024 р.

3. Вихідні дані до роботи відстань та напрям до перешкоди

4. Перелік питань, що потрібно опрацювати в роботі _____

1. Аналіз предметної області

2. Вибір і обґрунтування методу розв'язання

3. Програмна реалізація

4. Результати обчислювального експерименту

5. Аналіз можливих застосувань

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій _____

1. Актуальність теми роботи _____

2. Постановка задачі _____

3. Аналіз предметної області _____

4. Метод чисельного аналізу _____

5. Результати обчислювального експерименту _____

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Підбір та вивчення технічної літератури за темою роботи	6 – 12 листопада 2023 р.	виконано
2	Вибір та обґрунтування методу	13 – 26 листопада 2023 р.	виконано
3	Розробка алгоритму і програми	27 листопада – 10 грудня 2023 р.	виконано
4	Проведення аналітичних досліджень та розрахунків	11 грудня – 24 грудня 2023 р.	виконано
5	Робота над текстом пояснювальної записки	25 грудня 2023 р. – 9 січня 2024 р.	виконано
6	Представлення роботи на рецензію в ЕК	10 січня 2024 р.	виконано

Дата видачі завдання 6 листопада 2023 р.

Студент _____
(підпис)

Керівник роботи _____ доц. Матвієнко О.І.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка: 65 с., 9 табл., 29 рис., 1 дод., 11 джерел.

ЛІНГВІСТИЧНА ЗМІННА, МЕТОД МАМДАНІ, МОБІЛЬНИЙ РОБОТ,
НЕЧІТКА ЛОГІКА, ТЕРМ, ФАЗИФІКАЦІЯ, PYTHON.

Об'єкт дослідження – нечітка математична модель управління рухом мобільного роботу.

Мета роботи – розробка алгоритму для керування рухом мобільного роботу за допомогою теорії нечітких множин.

Методи дослідження – методи теорії нечітких множин.

Наразі актуальне застосування роботизованих і роботехнічних систем в різних галузях. Система управління таких систем повинна чітко відображати у собі поставлену перед роботом задачу і дозволяти приймати адекватні рішення в різних нестандартних ситуаціях.

Теорія нечітких множин широко застосовується для розв'язання задач управління складними технічними системами. На її основі отримано рішення великої кількості завдань аналізу та управління. Практичний досвід розробки систем нечіткого логічного висновку свідчить, що терміни та вартість їх проектування значно менші, ніж при використанні традиційного математичного апарату. Ця теорія дозволяє вирішувати завдання управління в ситуаціях, коли традиційні методи неефективні чи взагалі неприйнятні через брак точної інформації про об'єкт дослідження.

В кваліфікаційній роботі досліджено застосування методів нечіткої логіки для керування рухом мобільного роботу. Розроблено алгоритм, який дозволяє роботу, що рухається з постійною швидкістю, обходити перешкоди, які зустрічаються на його шляху. Було створено програмний продукт мовою програмування Python, який реалізує роботу алгоритму та візуалізує результати.

ABSTRACT

Introductory note: 65 pages, 9 tables, 29 figures, 1 appendix, 11 sources.

FUZZY LOGIC, FUZZIFICATION, LINGUISTIC VARIABLE, MAMDANI METHOD, MOBILE ROBOT, PYTHON, THERM.

Object of research – fuzzy mathematical model of mobile robot motion control.

Purpose of work – development of an algorithm for controlling the motion of a mobile robot using fuzzy set theory.

Methods of research – methods of the theory of fuzzy sets.

Currently, the application of robotic and robotic systems in various industries is relevant. The management system of such systems should clearly reflect the task set before the robot and allow adequate decisions to be made in various non-standard situations.

The theory of fuzzy sets is widely used to solve problems of managing complex technical systems. On its basis, solutions to a large number of analysis and management tasks were obtained. The practical experience of developing systems of fuzzy logical inference shows that the terms and cost of their design are significantly less than when using a traditional mathematical apparatus. This theory allows solving management tasks in situations where traditional methods are ineffective or generally unacceptable due to the lack of accurate information about the research object.

The application of fuzzy logic methods for controlling the movement of a mobile robot is investigated in the qualification work. An algorithm has been developed that allows a robot moving at a constant speed to bypass obstacles in its path. A software product was created in the Python programming language, which implements the operation of the algorithm and visualizes the results.

ЗМІСТ

	С.
Вступ	9
1 Аналіз предметної області та постановка задач дослідження	11
1.1 Fuzzy-управління.....	11
1.2 Структура регуляторів, побудованих з урахуванням нечіткої логіки	13
1.3 Методи фазифікації	17
1.3.1 Лінгвістичні терми та функції належності	17
1.3.2 Алгоритми фазифікації	20
1.4 Правила нечіткого логічного виведення	22
1.4.1 Fuzzy-логічні правила та оператори	22
1.4.2 Агрегування	24
1.5 Метод проєктування Fuzzy-регулятора	25
1.5.1 Проєктування Fuzzy-регулятора на основі досвіду та знань експерта	25
1.5.2 Вибір оптимальних параметрів Fuzzy-контролера методами імітаційного моделювання	26
1.5.3 Програмні системи проєктування Fuzzy-регулятора	28
1.6 Змістовна та формальна постановка задачі	29
1.7 Постановка задач дослідження	31
2 Вибір та обґрунтування методу розв’язання	33
2.1 Різні підходи до побудови Fuzzy-регуляторів	33
2.1.1 Концепт побудови Fuzzy-регулятора, запропонований проф. Мамдані (E.Mamdani)	33
2.1.2 Метод Sungeno	35
Висновки за розділом 2	36
3 Програмна реалізація	37
3.1 Мова програмування Python	37
3.2 Вхідні та вихідні дані	38

	7
3.3 Структура програми	39
3.3.1 Бібліотеки Python	39
3.3.2 Опис синтаксису програми	41
3.4 Результати роботи програми	44
Висновки за розділом 3	47
4 Результати обчислювального експерименту та їх аналіз	48
4.1 Основні позначення розділу	48
4.2 Обчислювальний експеримент для першого випадку	48
4.3 Обчислювальний експеримент для другого випадку	54
Висновки за розділом 4	59
Висновки	60
Перелік джерел посилання	61
Додаток А Лістинг програми	63

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАК, ОДИНИЦЬ І ТЕРМІНІВ

ВВ чи VN – велике від’ємне;

ВЗ або NN – від’ємне (негативне) значення;

ВП або VP – велике позитивне;

ЛО –літаючий об’єкт;

МВ або NN – мале від’ємне (негативне);

МП або NP – мале позитивне;

MP – мобільний робот;

ПЗ або PP – позитивне значення;

ПІ – пропорційно-інтегральний;

ПІД – пропорційно-інтегрально-диференційний;

ПН або ZE – приблизно нуль;

ПО або ZE – приблизно нуль;

СВ чи MN – середнє від’ємне;

СП або MP – середнє позитивне.

ВСТУП

Актуальність теми. У сучасному світі в різних галузях все частіше застосовуються роботизовані і роботехнічні системи. Але для продуктивної роботи таких систем потрібен точний та ємний опис системи управління. Система управління повинна чітко відображати поставлену перед роботом задачу і дозволяти приймати адекватні рішення в різних нестандартних ситуаціях. В роботі пропонується розглянути ідею побудови системи управління рухом мобільного робота на основі нечіткої логіки, яка має дозволити роботу, що рухається з постійною швидкістю, обійти перешкоду, яка зустрічається йому на шляху.

Наразі дуже часто встає питання в оптимізації виробництва. Мобільні роботи активно входять в наше повсякденне життя, облегшуючи і модернізуючи звичні нам справи. На сьогодні ця тема є актуальною і знаходиться в активній стадії дослідження. Можна виділити декілька авторів, що зробили суттєвий вклад у цю тему: В. Г. Фокін та С. В. Шанигін, М. А. Подалов та М. М. Поліщук. Наразі запропоновано кілька алгоритмів нечіткого виведення. Деякі із них застосовуються найчастіше, зокрема алгоритм Мамдані, алгоритм Цукамото, алгоритм Ларсена, алгоритм Сугено та інші.

Мета і завдання кваліфікаційної роботи. Метою кваліфікаційної роботи є розробка алгоритму для керування мобільним роботом. Для досягнення поставленої мети необхідно виконати наступні завдання:

- провести огляд і аналіз сучасного стану задачі «Управління мобільним роботом»;
- пошук інформаційних джерел та їх критичний аналіз;
- аналіз предметної області та постановка задач дослідження;
- вибір та обґрунтування методу дослідження поставленої задачі;
- розробка алгоритму розв’язання поставленої задачі.

Об’єктом дослідження є мобільний робот.

Предметом дослідження є система управління рухом мобільного робота.

Методи дослідження. У кваліфікаційній роботі використовуються методи нечіткої логіки.

Публікації. Результати, отримані у кваліфікаційній роботі, було представлено на 27-й Міжнародний молодіжний форум «РАДІОЕЛЕКТРОНІКА І МОЛОДЬ У ХХІ СТОЛІТТІ» (м. Харків 10 – 12 травня 2023р.) [5] та I Міжнародній молодіжній науково-практичній конференції англійською мовою «НАВЧАННЯ І ВИКЛАДАННЯ: після війни та під час миру» (м. Харків, 10 листопада 2023 року) [1].

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧ ДОСЛІДЖЕННЯ

1.1 Fuzzy-управління

Останнім часом найчастіше використання методу Fuzzy-логіки пов'язано з системами автоматичного управління та регулювання. Нині добре розвинений математичний апарат автоматичного регулювання. Відома велика кількість методів розрахунку та налаштування регуляторів, синтезу замкнутої оптимальної динамічної системи автоматичного управління, яка задовольняє всім наперед заданим обмеженням, на якість процесу регулювання. Однак, для складного динамічного об'єкта побудова ефективної системи управління на сьогоднішній день є мистецтвом і багато в чому визначається досвідом та інтуїцією конструктора чи проєктувальника. Особливо складним це завдання виявляється для динамічних об'єктів зі змінними параметрами та неконтрольованими збуреннями, для яких дуже складно отримати адекватний математичний опис динаміки процесу. У ряді випадків дуже важко вирішити завдання синтезу системи управління з такими об'єктами, тоді як оператор процесу, не використовуючи ніякого математичного опису системи, на основі своїх знань, досвіду та інтуїції успішно керує такими об'єктами. Можливість використання в таких ситуаціях різних евристик та експертних методів, досвіду та знань операторів, а також алгоритмів, що навчаються, і є основною передумовою широкого впровадження Fuzzy-технології.

Зрозуміло, що Fuzzy-логіка не є «панацеєю від усіх бід», і там, де успішно можуть бути застосовані традиційні методи безперервного чи цифрового регулювання, системи на базі Fuzzy-технологій не можуть з ними успішно конкурувати. Однак у тих випадках, коли закон регулювання дуже простий і для реалізації його не потребує побудови динамічної моделі об'єкта у вигляді системи диференціальних рівнянь, або для складних нелінійних нестационарних систем з параметрами, що змінюються, для яких математична модель є дуже

складною і згодом стає неадекватною до процесу, застосування систем управління та регулювання, заснованих на Fuzzy-логіці, може призвести до істотно менших витрат на створення, технічну реалізацію та проектування. Така система забезпечує якість процесу управління та регулювання не гірше, ніж у системах, синтезованих класичними традиційними методами. Крім того, в даний час широкого поширення набувають комплексні комбіновані системи управління технологічними процесами, що включають як безперервний (або цифровий) динамічний пропорційно-інтегральний (ПІ) або пропорційно-інтегрально-диференціальний (ПІД) регулятор, що функціонують на традиційній основі, та адаптивний блок, побудований на базі Fuzzy-логіки.

На думку фахівців, область застосування регуляторів на базі Fuzzy-логіки доцільна там, де сформульована природною людською мовою стратегія управління об'єктом у вигляді логічних правил, що пов'язують різні лінгвістичні терми вхідних, вихідних змінних і керуючих впливів, є повною, що реалізується на практиці і дає позитивні результати. Ця стратегія повинна бути перетворена на деякий алгоритм, що забезпечує на основі вимірних значень вхідних та вихідних змінних обчислення значень керуючих впливів. Перевага методів Fuzzy-технологій в даному випадку в тому, що можливості обробки нечіткої інформації, отриманої від оператора та вираженої на якісному чи логічному рівні, та моделюванні його процесів прийняття рішень, наприклад у вигляді: якщо значення 1-ї вхідної змінної дуже низьке і воно падає, і значення 2-ї змінної середнє і воно зростає, тоді необхідно збільшити значення 1-го керуючого впливу.

Методи Fuzzy-логіки дозволяють формалізувати даного виду стратегію, використовуючи функції приналежності кожного з логічних термів і систематизуючи інформацію експерта (оператора) як бази даних. Таким чином Fuzzy-регулятор, використовуючи представлену у вигляді бази даних стратегію управління, перетворює на основі алгоритмів фазифікації, нечіткого логічного висновку та методу дефазифікації вимірянні та обчислені значення відхилень величин фактичних регульованих параметрів від їх необхідних в даний момент

часу значень управляючого впливу. При цьому для кожної можливої комбінації лінгвістичних термів вхідних (для регулятора) параметрів у базі даних має міститися відповідне правило прийняття рішень.

1.2 Структура регуляторів, побудованих з урахуванням нечіткої логіки

Під регулюванням розуміється замкнутий цикл, у якому здійснюється:

- безперервний вимір контрольованої величини (вхідної змінної);
- порівняння виміряного значення з необхідним в даний момент часу значенням даного параметра та обчислення величини відхилення;
- обчислення відповідно до величини та знаку відхилення значення керуючого (коригувального) впливу та реалізація його на об'єкті.

Метою регулювання є підтримання постійного значення регульованої величини в умовах дії на об'єкті контрольованих та неконтрольованих впливів. Замкнений контур регулювання можна схематично подати на рисунку 1.1.

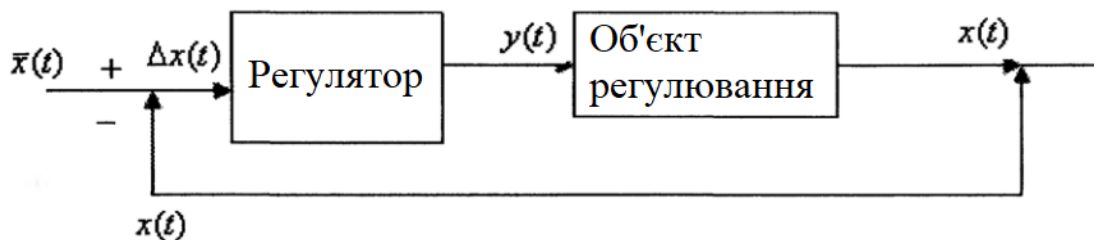


Рисунок 1.1 – Схема замкнутого контуру регулювання

На рисунку 1.1 використані такі позначення: $\bar{x}(t)$ – необхідне значення регульованої величини на момент часу t ; $x(t)$ – фактичне значення регульованої величини на час t ; $\Delta x(t) = \bar{x}(t) - x(t)$ – величина відхилення значення регульованої величини в момент часу t ; $y(t)$ – обчислене значення керуючого впливу.

У класичній теорії автоматичного регулювання запропоновано методи

обчислення значень керуючих впливів $y(t)$ на основі адекватного математичного опису динаміки та статички об'єкта регулювання, регулятора та замкнутого контуру регулювання. При цьому для вибору параметрів регуляторів використовують методи ідентифікації об'єктів регулювання на основі аналітичного математичного опису, що враховує фізичні, кінематичні, хімічні, механічні та інші закономірності процесів, що протікають в установках, конструктивні параметри, а також експериментальні методи ідентифікації, засновані на обробці даних активних або (і) пасивних експериментів. Для складних об'єктів найефективнішими є комбіновані методи, які використовують як теоретичні, і експериментальні дослідження. При цьому розрізняють методи математичного опису динаміки об'єкта у часовій та частотній області.

У багатьох випадках однією з найважливіших передумов при отриманні математичного опису об'єктів є припущення про лінійний характер залежності регульованої величини від збурень і впливів, що управляються в аналізованій області. Таке припущення дозволяє використовувати принцип суперпозиції щодо впливів, що збуджують і управляють, що діють на об'єкт регулювання. Однак при більш точному розгляді реальних технічних об'єктів управління доводиться, на жаль, констатувати, що більшість таких об'єктів є нелінійними. З іншого боку, параметри математичних моделей багатьох об'єктів є функціями від часу. Тобто багато реальних об'єктів є нестационарними. У ряді випадків впливи, що збуджують, що діють на об'єкт, є не детермінованими, а стохастичними функціями в часі.

Все вищевикладене дуже ускладнює застосування методів ідентифікації для отримання адекватного математичного опису реальних технічних та технологічних об'єктів. Особливо ці проблеми виявляються в описах багатозв'язкових (кілька входів і виходів) об'єктів і з розподіленими параметрами.

На рисунку 1.2 і рисунку 1.3 представлені структура, основні компоненти та змінні Fuzzy-контролера.

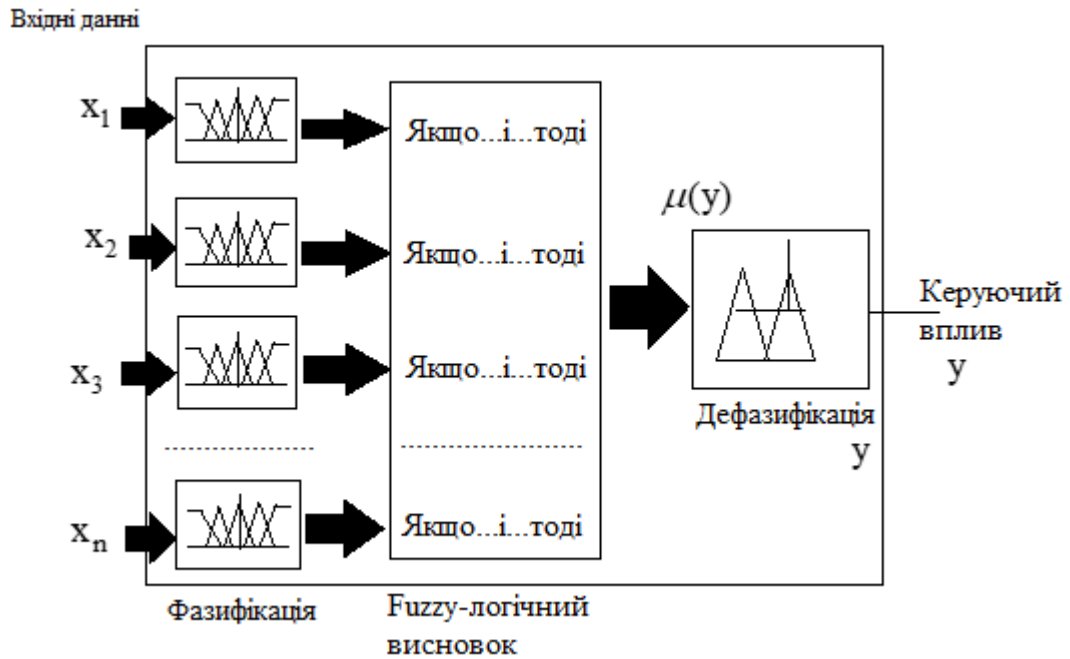


Рисунок 1.2 – Структура Fuzzy-контролера



Рисунок 1.3 – Структура основні елементи і змінні Fuzzy-контролера

Позначення на рис. 1.2, 1.3:

– $Y = y(t)$ – виміряне значення контрольованого параметра в момент часу t ;

– $\bar{Y} = \bar{y}(t)$ – задане значення контрольованого параметра в момент часу t ;

- $X = x(t)$ – обчислене значення керуючого впливу в момент часу t ;
- $\Delta Y = \Delta y(t) = \bar{y}(t) - y(t)$ – відхилення контрольованого параметра в момент часу t ;
- $Z = z(t) = \Delta y(t) - \Delta y(t-1)$ – похідна від відхилення контрольованого параметра на момент часу t ;
- $y(t), \bar{y}(t), z(t), \Delta y(t)$ – дійсні числа;
- $A_i, i = 1, \dots, n$, – різні терми (нечіткі множини) лінгвістичної змінної $\Delta y(t)$ (наприклад, велике позитивне відхилення, мале позитивне відхилення, відсутність відхилення, мале негативне і велике негативне відхилення);
- $\mu_A(y) = \mu_A(\Delta y(t))$ – значення функції приналежності кожного з лінгвістичних термів;
- $B_j, j = 1, \dots, r$, – різні терми (Fuzzy-множини) лінгвістичної змінної;
- $z(t)$ – похідною від відхилення у час t ;
- $\mu_B(z) = \mu_{B_j}(z(t))$ – значення функції приналежності кожного з лінгвістичних термів;
- $G_k, k = 1, \dots, m$, різні терми (нечіткі множини) лінгвістичної змінної $x(t)$ (значення керуючого впливу в момент часу t);
- $\mu_{G_k}(x(t)), k = 1, \dots, m$ – значення функції приналежності кожної з нечітких множин лінгвістичних термів.

Окремі блоки Fuzzy-регулятора, представленого рисунку 1.3 виконують такі функції.

Блок фазифікації. Блок фазифікації здійснює перетворення фактичних значень відхилення $\Delta y(t)$ і похідної від відхилення контрольованої величини $z(t)$, обчислених на основі вимірювань і порівняння із заданими величинами і виражених у вигляді дійсних чисел, в терми відповідних лінгвістичних змінних, які представлені нечіткими множинами (відповідно $A_i, i = 1, \dots, n$, і

$B_j, j=1, \dots, r$). Обчислюються значення функцій приналежності для кожного з цих термів – $\mu_{A_i}(\Delta y(t)), i=1, \dots, n$ і $\mu_{B_j}(z(t)) j=1, \dots, r$.

База знань. База знань містить діапазони зміни кожної з лінгвістичних змінних (відхилення контрольованої величини $\Delta y(t)$ та її похідної $z(t)$), а також керуючого впливу $x(t)$, кількість, найменування, діапазони зміни, математичні вирази та алгоритми розрахунку функцій приналежності для всіх термів кожної лінгвістичної змінної. Крім того, там зберігаються всі правила, оператори та математичні вирази Fuzzy-логічного висновку.

Блок дефазифікації. У блоці дефазифікації здійснюється перетворення значень функції приналежності для всіх термів лінгвістичних змінних керуючих впливів на значення керуючих впливів, виражених дійсним числом, що лежить в допустимому для них діапазоні. В алгоритмі дефазифікації використовується інформація з бази даних про діапазони зміни та формальні вирази функцій приналежності для кожного з термів лінгвістичних змінних керуючих впливів, а також формули вирази обраного алгоритму дефазифікації.

1.3 Методи фазифікації

1.3.1 Лінгвістичні терми та функції приналежності

Лінгвістичні терми та функції приналежності. Усі можливі діапазони зміни відхилення контрольованої величини від заданого в момент 1 значення $\Delta y(t) - [\Delta y_{\min}(t), \Delta y_{\max}(t)]$, а також похідною від цього відхилення $\Delta z(t) - [\Delta z_{\min}(t), \Delta z_{\max}(t)]$, та керуючого впливу $\Delta x(t) - [\Delta x_{\min}(t), \Delta x_{\max}(t)]$, розбивається на низку інтервалів, що відповідають різним лінгвістичним термам. Кожному з цих інтервалів ставиться у відповідність певний лінгвістичний терм. Так, наприклад, при розбиття діапазону зміни змінної на

три інтервали цими термами є: позитивне значення (ПЗ або PP), приблизно нуль (ПО або ZE), від'ємне (негативне) значення (ВЗ або NN).

При розбитті діапазону зміни змінної на п'ять інтервалів вводяться наступні 5 термів: велике позитивне (ВП або VP), мале позитивне значення (МП або NP), приблизно нуль (ПН або ZE), мале від'ємне (негативне) (МВ або NN), велике від'ємне значення (ВВ чи VN). При розбитті на сім інтервалів лінгвістичні терми часто мають такі назви: велика позитивна (БП або BP), середня позитивна (СП або MP), мала позитивна (МП або NP), приблизно нуль (ПО або ZE), мала від'ємне (негативна) (МВ чи NN), середнє від'ємне (СВ чи MN), велике від'ємне значення (БН чи BN). Відповідним чином можуть бути розширені назви термів (дуже велике позитивне або від'ємне значення та ін.) при розбиття всього діапазону зміни змінної на більшу (9 або 11) кількість інтервалів (термів).

На підставі знань та досвіду експертів для нечіткої множини кожного з термів лінгвістичної змінної підбираються вид та параметри функції приналежності.

Найбільшого поширення набули такі види функцій приналежності: для першого і другого термів контрольованих і регульованих параметрів – відповідно у вигляді правої і лівої трапеції (рідше – правого і лівого трикутника або право- і лівосторонніх експоненційних і дробоподібних функцій). Для інших термів цих змінних найчастіше – у вигляді центральних трикутників, центральних трапецій, і навіть центральних експоненціальних чи дробоподібних функцій різного виду. Для термів керуючого впливу також часто використовуються функції приналежності цього ж виду, проте в ряді випадків широко застосовуються і функції приналежності у вигляді одиничного імпульсу.

Трикутні та трапецієподібні функції знаходять широке застосування в системах Fuzzy-регулювання, тому що вони є кусочно-лінійними. При цьому в процесі розрахунків суттєво скорочуються обсяги обчислень, що дозволяє з необхідною точністю проводити всі обчислення. На рисунку 1.4 та рисунку 1.5 представлені відповідно приклади лінгвістичних змінних, що

використовуються в системах Fuzzy-регулювання і містять три терми (грубе розбиття) та сім термів (точно розбиття).

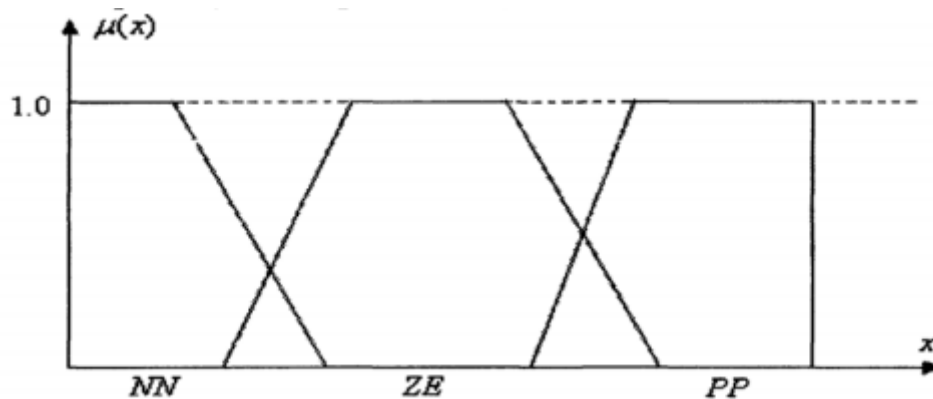


Рисунок 1.4 – Приклад розбиття лінгвістичної змінної на 3 лінгвістичні терми

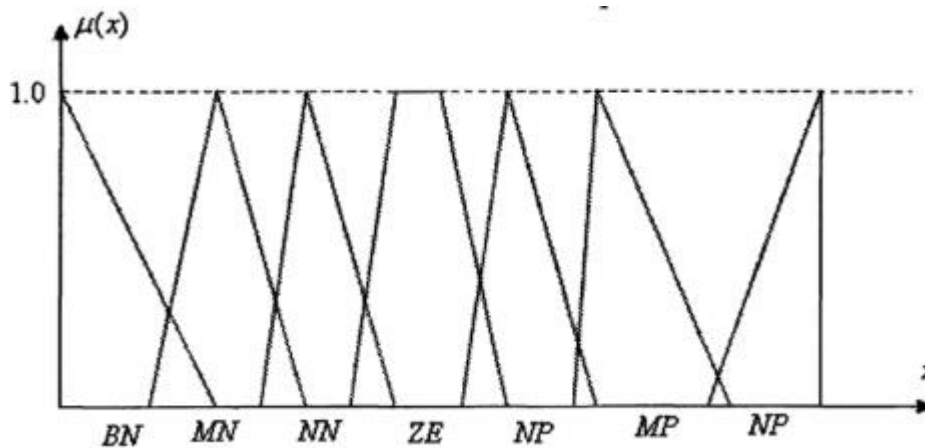


Рисунок 1.5 – Приклад розбиття лінгвістичної змінної на 7 лінгвістичних термів

Отже, для ефективної роботи блоку фазифікації мають бути визначені та занесені до бази знань системи такі дані:

- діапазони зміни лінгвістичних змінних для кожної вхідної змінної та керуючих впливів;
- позначення та градуювання всіх лінгвістичних змінних;
- кількість, найменування та інтервали для всіх термів кожної лінгвістичної змінної;
- різновиди та параметри функцій приналежності для кожного з термів усіх лінгвістичних змінних;

- опорні множини для кожного з термів;
- області перекриття лінгвістичних термів.

1.3.2 Алгоритми фазифікації

У самому загальному випадку обчислене або виміряне значення вхідної змінної Δy або z може також визначатися деяким нечіткою множиною $C(y)$ з функцією приналежності $\mu_C(y)$, яка може, наприклад, мати вигляд експоненти або рівномірного трикутника. Ступінь узгодженості передумови, що обчислене значення деякої вхідної змінної y відноситься до певного лінгвістичного терму, $A_i(y)$ визначається коефіцієнтом довіри, що обчислюється за формулою

$$\bar{G}_P(l) = \max_{y \in Y} \{ \min(\mu_C(y), A_i(y)) \}, l = 1, \dots, L, \quad (1.1)$$

де L – кількість лінгвістичних термів змінної y ,

$\bar{G}_P(l)$ – міра збігу (ваговий коефіцієнт) ідеальної передумови $A_i(y)$ та актуальної передумови $\bar{A}_i(l)$, що обчислене значення вхідної змінної y належить l -му лінгвістичному терму.

Відповідно до формули (1.1) на початку для кожного із значень знаходиться мінімум зі значень відповідних функцій приналежності, а потім серед цих мінімумів визначається максимальне значення. В окремому випадку, коли обчислення або вимірювання значення y виконуються точно, функція приналежності $\mu_C(y)$ є імпульсною, вираз (1.1) спрощується та має вигляд

$$\bar{G}_P(l) = \max_{y \in Y} \{ \min(\mu_{singl}(y), \mu_{A_i}(y)) \} = \min[1; \mu_{A_i}(y^1)] = \mu_{A_i}(y^1), l = 1, \dots, L. \quad (1.2).$$

Графічне уявлення формули (1.2) наведено на рисунку 1.6.

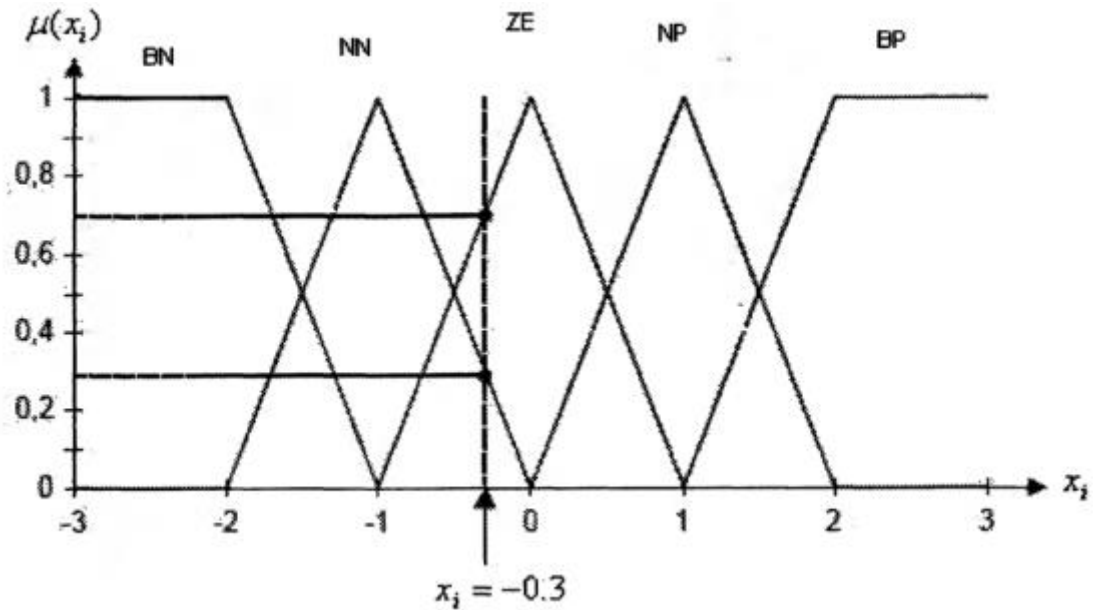


Рисунок 1.6 – Ілюстрація процесу фазифікації вхідної змінної

Надзвичайно важливим є розгляд випадків наявності перекриттів (оверлеїв) окремих нечітких множин термів лінгвістичних змінних контрольованих параметрів. Ця обставина дозволяє вимірюване або обчислене значення контрольованого параметра відносити з різним значенням функції приналежності (коефіцієнта довіри) одночасно до різних терм і на основі цього паралельно обробляти кілька альтернатив.

Отже, результатом роботи блоку фазифікації є n векторів $\bar{G}_p(i, l)$, $i=1, \dots, n$, $l=1, \dots, L$ (де n – кількість вхідних змінних), елементи яких – значення функцій приналежності віднесення обчисленого чи виміряного значення вхідних змінних до різних термів відповідної лінгвістичної змінної.

Рисунок 1.6 ілюструє роботу алгоритму фазифікації вхідних змінних. Значення вхідної змінної $x_i = -0.3$ належить одночасно двом термам: терму (мале негативне значення) зі значенням $\mu_{NN}(x_i) = 0.3$, і терму (приблизно нуль) зі значенням $\mu_{ZE}(x_i) = 0.7$.

навчання. До нього пред'являються вимоги повноти і однозначності, тобто всі можливі ситуації поєднання різних термів лінгвістичних змінних контрольованих параметрів мають бути передбачені основою знань. Для кожної з цих ситуацій має бути знайдена однозначна відповідь щодо вибору вектору термів лінгвістичних змінних керуючих впливів.

В імплікації виду

$$\text{якщо}(y_i \in A_i), \text{тоді}(x_j \in B_j)$$

терми $(y_i \in A_i)$ називаються антецедентом (лівою частиною імплікації), а $(x_j \in B_j)$ – заключенням.

База даних системи (1.3) може бути представлена у вигляді деякої багатовимірної матриці або багатовимірного масиву, розмірність яких визначається кількістю змінних вхідних. Розмірність кожної координати визначається при цьому кількістю термів лінгвістичної змінної відповідного вхідного параметра. Кожен елемент цього багатовимірного масиву – вектор, розмірність якого дорівнює кількості керуючих впливів, що містить номери (або найменування) термів лінгвістичних змінних всіх керуючих впливів, розташованих у порядку їх індексації.

Для прикладу наведемо матрицю бази знань системи Fuzzy-регулювання, що містить дві вхідних змінних і два керуючих впливу. Як вхідні змінні вибрано величину відхилення контрольованого параметра та похідну від цього відхилення. Перша вхідна лінгвістична змінна має п'ять термів: BN, NN, ZE, NP, BP; друга – три терми: NN, ZE, PP. Перший керуючий вплив містить три терми: NN, ZE, PP, а другий керуючий вплив - п'ять термів: BN, NN, ZE, NP, BP. Матриця, що містить базу знань системи, представлена у таблиці 1.1.

Таблиця 1.1 – Вид багатовимірної матриці, що відображає базу знань системи (правила логічного виводу)

	BN	NN	ZE	NP	BP
PP	NP; ZE	ZE; ZE	NN; NN	BN; NN	BN; NN
ZE	BP; PP	NP; PP	ZE; ZE	NN; PP	BN; NN
NN	BP; PP	BP; PP	NP; PP	ZE; ZE	NN; ZE

Fuzzy-логічний вивід перетворює якісну інформацію, що міститься в базі даних і відображає думки експертів, на технічно реалізований висновок. Цей висновок формується на основі системи логічних правил та ступеня достовірності різних формул щодо належності вектора вхідних змінних системи до відповідних термів лінгвістичної змінної.

Процес Fuzzy-логічного виводу умовно можна поділити на три етапи:

- агрегування (зведення у блок) елементарних висловлювань;
- імплікація;
- об'єднання правил та формування загального висновку.

1.4.2 Агрегування

Агрегування об'єднує різні елементарні висловлювання в одне правило і обчислює ступінь достовірності складної (об'єднаної) формули. Кожна складна формула логічного правила складається з n змінних (де n – кількість вхідних змінних) елементарних формул про належність вхідної змінної у певному терму лінгвістичної змінної $A_{il}(y_i \in A_{il})$ з коефіцієнтом довіри, що дорівнює $\bar{G}_p(i,l) = \mu_{A_{il}}(y_i)$.

У системах Fuzzy-регулювання та управління як оператори агрегування найбільше поширення, завдяки їх простій формі реалізації та ясності одержуваного результату, отримали оператори T_{\min} , S_{\max} або T_{algebr} , S_{algebr} .

Оператор мінімум-агрегація має вигляд:

$$\bar{G}_s(y) = \min[\bar{G}_{1,s}(y_1), \bar{G}_{2,s}(y_2), \dots, \bar{G}_{n,s}(y_n)] = \min_{1 \leq i \leq n} \bar{G}_{i,s}(y_i) \quad \forall s = 1, \dots, S.$$

Оператор Product-агрегація має такий вигляд:

$$\bar{G}_s(y) = \prod_{i=1}^n \bar{G}_{i,s}(y_i) \quad \forall s = 1, \dots, S.$$

Якщо всі поодинокі формули, що входять до Fuzzy-логічне правила, мають ступінь достовірності, що дорівнює 1, то $\bar{G}_s(y) = 1$. Якщо хоча б одна з формул має $\bar{G}_{i,s}(y_i) = 0$, то $\bar{G}_s(y) = 0$, і S логічне правило є неактивним.

1.5 Метод проектування Fuzzy-регулятора

1.5.1 Проектування Fuzzy-регулятора на основі досвіду та знань експерта

На основі інформації експерта може бути сформульована база даних, в якій містяться прості та логічно зрозумілі вимоги для проектування Fuzzy-регулятора. Інший шлях побудови системи полягає у копіюванні правил вже функціонуючого аналогового чи цифрового регулятора. Виміряні значення вхідних, вихідних та керуючих параметрів можуть бути основою для побудови та моделювання регулятора, побудованого на основі Fuzzy-технологій. Однак звернення до експертних методів, як відомо, є виправданим тоді, коли відсутня можливість строго формалізованих методів. Якщо ж формальне рішення завдання існує, то використання замість нього експертного рішення неминуче призведе до гірших результатів

1.5.2 Вибір оптимальних параметрів Fuzzy-контролера методами імітаційного моделювання

Проектування систем управління на основі методів Fuzzy-логіки пов'язане з вирішенням наступних проблем:

- вибір кількості лінгвістичних термів для кожної вхідної, вихідної змінної та для кожного керуючого впливу;
- вибір виду нечіткої множини та параметрів функцій приналежності для кожного з термів всіх лінгвістичних змінних вхідних параметрів та керуючих впливів;
- вибір правил логічного висновку та параметрів Fuzzy-логічних операторів у блоці прийняття рішень;
- вибір найефективнішого алгоритму дефазифікації.

Схема проведення експериментальних робіт вибору параметрів Fuzzy-контролера наведена на рис. 1.7.

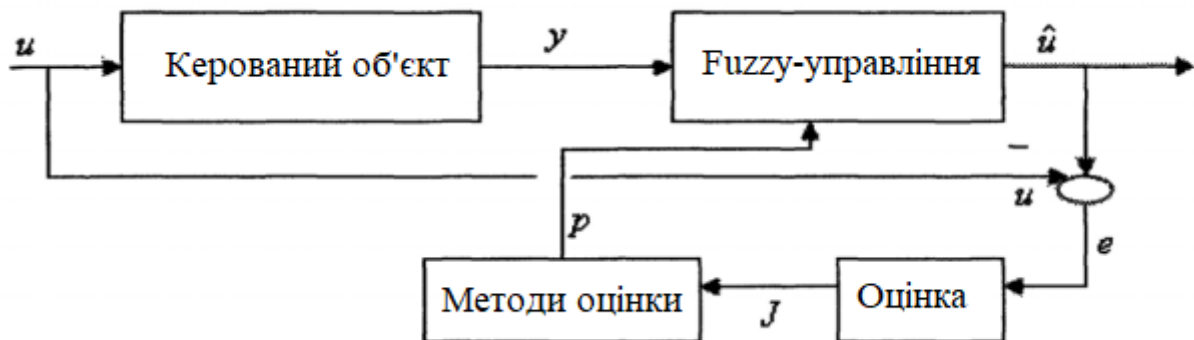


Рисунок 1.7 – Схема проведення експериментальних робіт вибору параметрів Fuzzy-контролера

Вирішення сформульованих проблем під час проектування реальних систем управління пов'язані з вибором значень великої кількості параметрів. Це завдання може бути сформульоване у вигляді багатоекстремальної задачі математичного програмування великого розміру в умовах заданої системи обмежень. Складності вирішення цієї задачі полягають також і в тому, що

функціонал якості системи та ліві частини функцій обмежень задачі не є явними функціями від параметрів, що оптимізуються. В даний час не існує однозначних рекомендацій та аналітичних методів ефективного вирішення цієї проблеми. Рекомендації експертів можуть допомогти лише в рідкісних та дуже простих ситуаціях. Найбільш ефективним методом вибору оптимальних параметрів таких систем є метод імітаційного моделювання. Схема проведення імітаційних експериментів представлена рисунку 1.8.

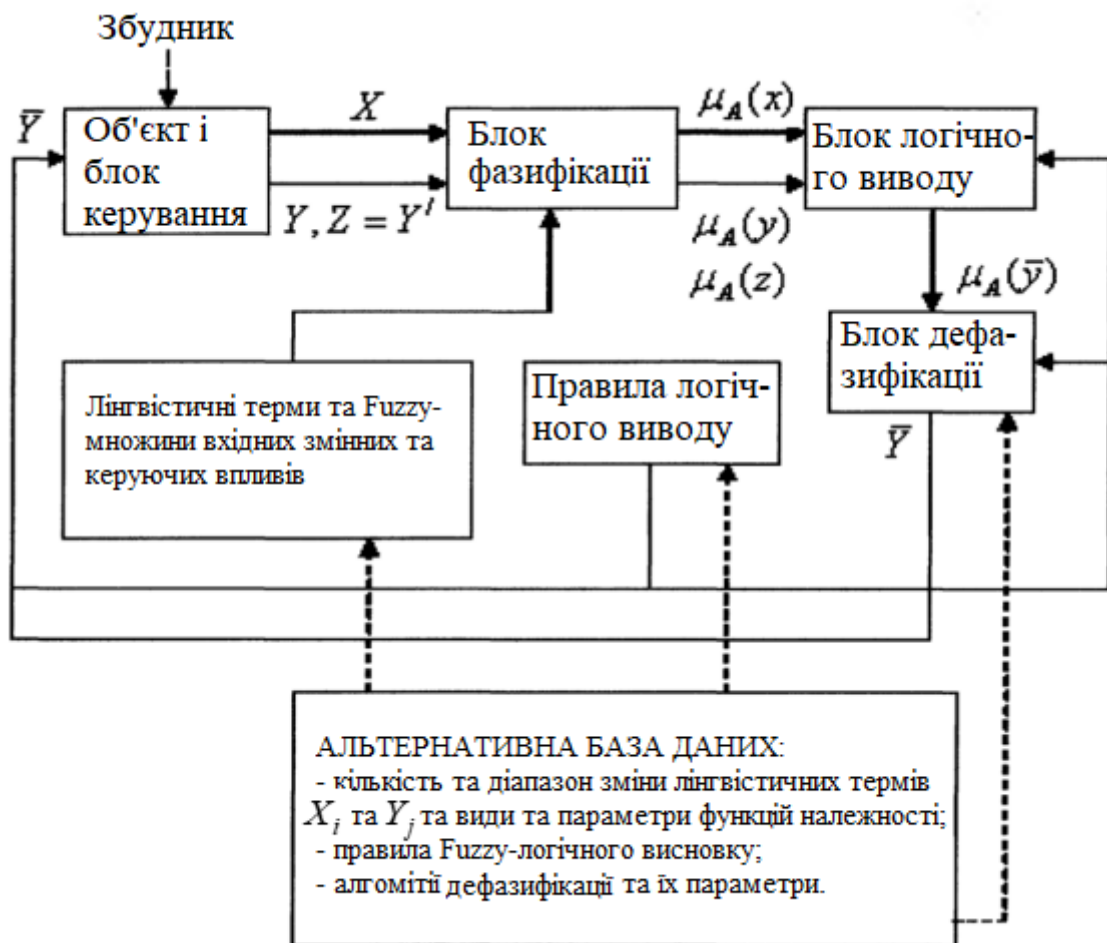


Рисунок 1.8 – Схема вибору параметрів Fuzzy-регулятора методами імітаційного моделювання

На кожному кроці ітеративного процесу випадковим чином або на основі сформульованих евристик, або на основі аналізу результатів раніше проведених експериментів з бази даних можливих альтернатив вибираються:

– кількість лінгвістичних інтервалів кожної з вхідних, вихідних змінних

та керуючих впливів; діапазони їх зміни та види функцій приналежності та параметри цих функцій;

– види та параметри правил Fuzzy-логічного висновку та обчислення значень функцій приналежності обчислюваних значень управляючих впливів;

– види та параметри алгоритмів дефазифікації.

Для альтернативних параметрів аналізованої системи Fuzzy-логічного управління об'єктом обчислюємо значення управляючих впливів і виконуємо оцінки якості управління для різних видів з можливого діапазону впливів, що збуджують. Параметри альтернативної системи, яка задовольняє системі обмежень, та комплексний критерій оцінки якості управління зберігаємо пам'яті системи.

Вибираємо з усіх альтернатив, що зберігаються в пам'яті, ту, що забезпечує виконання всіх обмежень і найкраще значення комплексного показника ефективності.

1.5.3 Програмні системи проектування Fuzzy-регулятора

Вибір ефективних параметрів Fuzzy-регулятора може бути здійснений на основі широкого поширеної в різних галузях математичного моделювання програмної системи MATLAB. Система MATLAB надає користувачеві, поряд з широким спектром можливостей аналізу та візуалізації динамічних характеристик замкнутих систем автоматичного регулювання, спеціальне програмне забезпечення конструювання та аналізу різних систем діагностики та управління, що базуються на Fuzzy-логіці. Вирішення цих проблем може бути здійснено за допомогою версії пакета 3.5.1, додатково включивши програмні системи Fuzzy-Logic-Tolbox (версія 2.0.1) та Simulink-Tolbox (версія 3.0.1).

Fuzzy-Logic-Tolbox дозволяє виконати всі етапи проектування системи (FIS – Fuzzy-Inferenz-Systeme): фазифікація, Fuzzy-логічний вивід та дефазифікація.

Одна з модифікацій пакета (GUI) орієнтована на менш підготовлених користувачів, інша (MATLAB Command Line) – більш досвідчених користувачів.

Якщо вибрано всі параметри Fuzzy-регулятора, програмними пакетами MATLAB та Simulink-Tolbox можуть бути інтегровані всі складові в загальну структуру управління, що дозволить виконати детальний аналіз та оцінку якості керування спроектованої системи.

1.6 Змістовна та формальна постановка задачі

Розглянемо процес керування мобільним роботом і одну із задач, що при цьому зустрічаються: роботу, що рухається з постійною швидкістю, необхідно обійти перешкоду, яка зустрічається на шляху.

Введемо дві вхідні лінгвістичні змінні:

- «дистанція» (відстань d до перешкоди) з термами «далеко», «середньо», «близько», «дуже близько» (рисунок 1.9);
- «напрямок» (кут між поздовжньою віссю робота і направленням на перешкоду) з термами «ліворуч», «прямо», «праворуч» (рисунок 1.10).

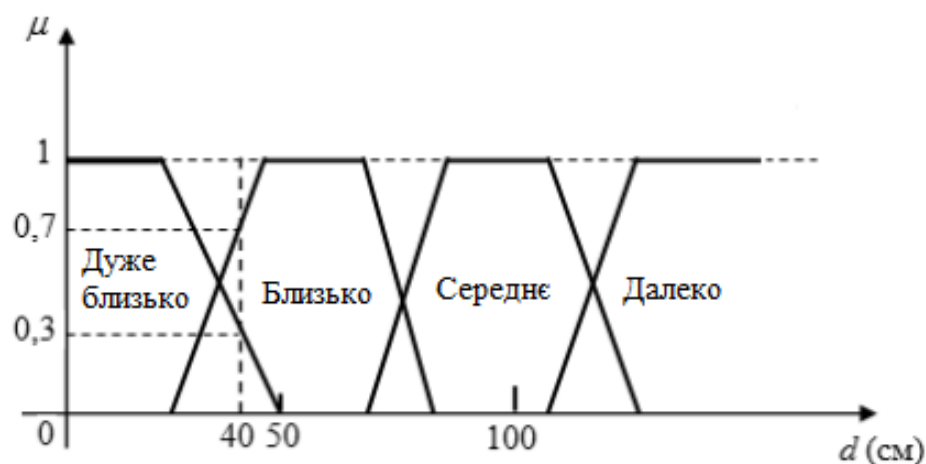


Рисунок 1.9 – Функції приналежності термів змінної «дистанція»

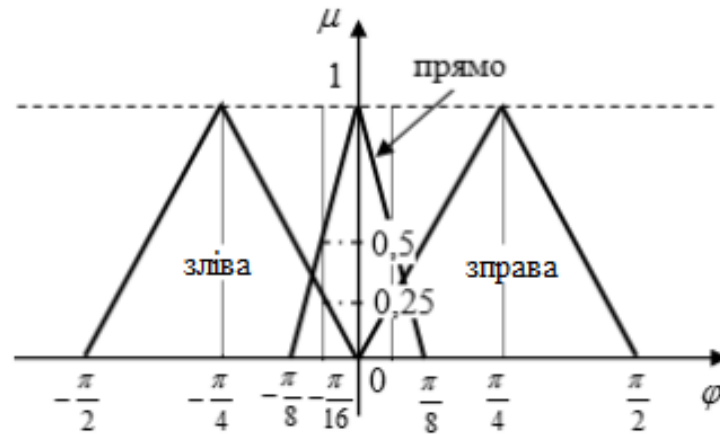


Рисунок 1.10 – Функції приналежності термів змінної «напрямок»

Вихідна змінна – «кермовий кут» (кут повороту керма) ψ с термами «різко вліво», «вліво», «прямо», «вправо», «різко вправо» (рисунок 1.11).

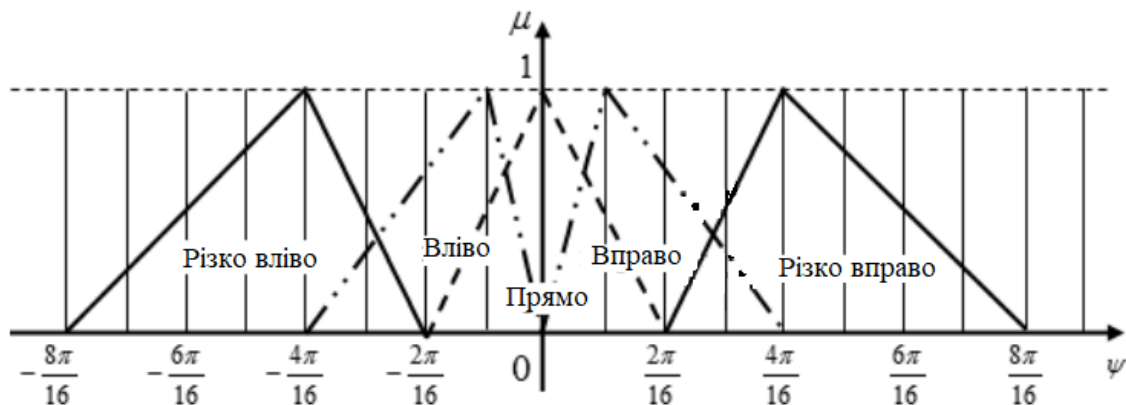


Рисунок 1.11 – Функції приналежності термів вихідної змінної

Задаємо нечітку базу даних для прикладу, що розглядається, за допомогою таблиці 1.2. Кожен запис у цій таблиці відповідає своєму нечіткому правилу, наприклад: ЯКЩО «дистанція» = «близько» І «напрямок» = «праворуч», ТО «кермовий кут» = «ліворуч».

Таким чином, мобільний робот з нечіткою логікою буде працювати за таким принципом: дані з сенсорів про відстань до перешкоди та направлення на неї будуть фазифіковані, оброблені згідно з табличними правилами, дефазифіковані та отримані дані у вигляді керуючих сигналів надійдуть на привід робота.

Таблиця 1.2 – Таблиця нечітких правил (база даних)

Напря́м	Дистан́ція			
	Дуже близько	Бли́зько	Середнє	Далеко
Справа	Вліво	Вліво	Прямо	Прямо
Прямо	Рі́зко влі́во (рі́зко пра́во)	Влі́во (Впра́во)	Влі́во (Впра́во)	Прямо
Злі́ва	Впра́во	Впра́во	Прямо	Прямо

1.7 Постановка задач дослідження

В даний час практично у всіх промислово розвинених країнах інтенсивно ведуться роботи зі створення і дослідження мобільних роботів. Побудова ефективних засобів в області мехатроніки і робототехніки нерозривно пов'язана зі створенням відповідних інструментальних середовищ для випробування навігаційних систем, мобільних роботів (МР). Створення випробувальних стендів різного призначення, розробка методик оцінки надійності, контролю та діагностики МР в цілому і підсистем, що входять до його складу є актуальною науково-технічною задачею. Ефективне виконання функцій МР можливо на основі інтегрованого застосування технологій управління його експлуатацією та обслуговуванням. Такі країни як США, Голландія, Швеція та інші починають повністю переходити до автоматизованого сільськогосподарського виробництва. Впровадження роботів в цю індустрію дозволить істотно підвищити продуктивність і рентабельність сільського господарства, тому, на сьогодні, актуальна необхідність дослідження в цій області, розробка методик побудови універсальних робототехнічних систем.

Система управління мобільним роботом повинна вирішувати такі завдання:

– обробка сенсорних даних (в т.ч. даних від інтерфейсу з оператором) з метою збору інформації про робота і зовнішнє середовище навколо нього;

- планування заходів по з'ясуванню цільового завдання і планування послідовності підзадач, необхідних для виконання цього завдання;

- формування таких програмних траєкторій руху ІМР, які б приводили до виконання роботом локальної підзадачі (наприклад, прибуття до цільової точки в середовищі з перешкодами);

- формування таких впливів, що задаються на виконавчі механізми робота, які б призводили до максимально точного і швидкого виконання ними програмної траєкторії руху.

Однією з особливостей побудови системи управління мобільного робота є те, що вона будується за ієрархічним багаторівневим принципом, згідно з яким з підвищенням ієрархічного рангу підсистеми підвищується її ступінь інтелектуальності. Самою верхньою ланкою цієї ієрархії є система управління поведінкою, далі йде система управління рухом, а система управління виконавчими механізмами є найнижчою ланкою цієї ієрархії. Крім перерахованих підсистем в структурі є інформаційно-вимірювальна система, яка також повинна володіти деякими інтелектуальними можливостями, і інтерфейс з оператором.

2 ВИБІР ТА ОБҐРУНТУВАННЯ МЕТОДУ РОЗВ'ЯЗАННЯ

2.1 Різні підходи до побудови Fuzzy-регуляторів

2.1.1 Концепт побудови Fuzzy-регулятора, запропонований проф. Мамдані (E. Mamdani)

У функції Fuzzy-регулятора входить встановити значення керуючого впливу, що забезпечує стабільність роботи системи, мінімізувати вплив зовнішніх збурень на регульовану величину.

Вперше концепт побудови Fuzzy-регулятора був запропонований у 1972 році L. Zadeh. Перша система управління паровим котлом з використанням Fuzzy-логіки була спроектована E. Mamdani і S. Assilian. Створення проф. E. Mamdani першого Fuzzy-регулятора сприяло широкому застосуванню Fuzzy-логіки в багатьох промислових застосуваннях і в побутовій техніці.

Fuzzy-контролер у філософії проф. E. Mamdani складається з наступних компонентів:

- блок фазифікації, який здійснює перетворення вимірних сигналів і значень вхідних і вихідних змінних, представлених дійсними числами, до Fuzzy-множини відповідних логічних термів;

- база даних, тобто блок Fuzzy-логічного виведення з урахуванням правил виду (1.3) або таблиці 1.1;

- блок прийняття рішень, що перетворює результати Fuzzy-логічного висновку на значення Fuzzy-множин керуючих змінних системи;

- блок дефазифікації, який перетворює ці Fuzzy-множини на значення керуючих змінних, представлених дійсними числами та перетворення цих обчислених значень на керуючі сигнали.

Експерти формулюють алгоритм прийняття рішень у формі деяких Fuzzy-логічних правил. Визначаються та ідентифікуються лінгвістичні терми всіх вхідних X_1, X_2, \dots, X_n і вихідних Y_1, Y_2, \dots, Y_n змінних (наприклад, «дуже низька»,

«низька», «середня», «позитивна» », «дуже високе» або «негативне», «майже нуль», «позитивне» значення). Позначимо ці логічні терми $A_i^1, A_i^2, \dots, A_i^p, i = 1, \dots, n$ та $B_j^1, B_j^2, \dots, B_j^g, j = 1, \dots, m$.

Будуються функції приналежності кожного з термів всіх цих лінгвістичних змінних вхідних параметрів системи. Кожна побудована нечітка множина асоціюється з відповідним термом лінгвістичної змінної конкретного вхідного або вихідного параметра. Отже, множина можливих значень кожної вхідної X_i і вихідний змінної Y_j представлена відповідно p_i та g_k нечіткими множинами

$$\begin{aligned} \{\mu_1(x_i), \dots, \mu_p(x_i)\} &\in F(X_i), i = 1, \dots, n; \\ \{\mu_1(y_j), \dots, \mu_g(y_j)\} &\in F(Y_j), j = 1, \dots, m. \end{aligned}$$

Найбільшого поширення подання цих нечітких множин отримали трикутні і трапецевидні функції приналежності.

Правила прийняття рішень формулюються у вигляді логічних операторів та правил, аргументами яких є ці логічні терми.

Наприклад,

$$\begin{aligned} Pr_k : & \left(\text{якщо } x_1 \in A_1^1, x_2 \in A_2^1, \dots, x_n \in A_n^{p-1} \right) \\ & \text{тоді } (y_1 \in B_1^2, y_2 \in B_2^3, \dots, y_m \in B_m^g); \\ Pr_{(k+1)} : & \left(\text{якщо } x_1 \notin A_1^p, x_2 \in A_2^1 \cup A_2^2, \dots, x_n \notin A_n^{p-1} \cup A_n^{p-1} \right) \\ & \text{тоді } (y_1 \in B_1^1, y_2 \in B_2^1, \dots, y_m \in B_m^g). \end{aligned}$$

Ці Fuzzy-логічні правила згідно з концепцією Мамдані повинні розглядатися не як правила імплікації, а деякі Fuzzy-логічні функції або оператори виду $y_j = \mathcal{G}(x_1, \dots, x_n), j = 1, \dots, m$.

При прийнятті рішень розглядається кожне вирішальне правило, включене до бази знань системи. Для кожного вирішального правила обчислюється значення функції приналежності до відповідного терму цієї вихідної змінної $\mu(B_j^\gamma | Pr_k)$, $\gamma = 1, \dots, g$, $j = 1, \dots, m$, $k = 1, \dots, K$, яке в концепції Мамдані обчислюється за формулою $\mu(B_j^\gamma | Pr_k) = \min \{ \mu(A_1 | Pr_k), \mu(A_2 | Pr_k), \dots, \mu(A_n | Pr_k) \}$. Обчислене значення надалі береться до уваги, якщо воно не нижче встановленого експертами порога довіри ε , тобто $\mu(B_j^\gamma | Pr_k) \geq \varepsilon$.

Якщо кілька Fuzzy-логічних правил визначають той самий терм вихідної змінної $k \in \bar{K}_1(B_j^\gamma)$, то значення функції приналежності $\mu(B_j^\gamma)$ обчислюється за формулою

$$\mu(B_j^\gamma) = \max_{k \in \bar{K}_1(B_j^\gamma)} \mu(B_j^\gamma | Pr_k), \quad \gamma = 1, \dots, g, \quad j = 1, \dots, m.$$

2.1.2 Метод Sugeno

Метод Sugeno є деякою модифікацією методу Мамдані. У цьому методі розбиваються на терми лише множини значень лінгвістичних змінних вхідних параметрів X_1, X_2, \dots, X_n . Для кожної змінної множини керуючих впливів Y_1, Y_2, \dots, Y_n таке розбиття не проводиться, а будується деяка нечітка множина з відповідною функцією приналежності $\mu(y_i)$ для всього діапазону можливих значень цієї змінної. Кожне з вирішальних правил бази знань системи формулюється так:

$$\text{Pr}_k : \left. \begin{array}{l} \text{якщо } x_1 \in A_1^1, x_2 \in A_2^1, \dots, x_n \in A_n^{p-1} \\ \text{тоді } \left\{ \begin{array}{l} y_1 = f_{1k}(x_1, x_2, \dots, x_n); y_2 = f_{2k}(x_1, x_2, \dots, x_n); \dots; \\ y_n = f_{nk}(x_1, x_2, \dots, x_n) \end{array} \right\}, \end{array} \right\}$$

$$k = 1, \dots, K.$$

Завдання фазифікації та Fuzzy-логічного виводу при цьому полягає у перетворенні вектора значень вхідних змінних $(x_1, x_2, \dots, x_n) \in (X_1 \times X_2 \times \dots \times X_n)$ на значення функції приналежності $\mu(y_i)$ кожного з керуючих впливів Y_j , $j = 1, \dots, m$. Методами дефазифікації за формулами

$$y_j = \frac{\sum_{k=1}^K \alpha_{jk} \cdot f_{jk}(x_1, x_2, \dots, x_n)}{\sum_{k=1}^K \alpha_{jk}}, \quad j = 1, \dots, m$$

розраховуються значення управляючих впливів.

Висновки за розділом 2

У другому розділі було розглянуто підходи до побудови Fuzzy-регуляторів: методи Мамдані та Сугено.

Для вирішення задачі в кваліфікаційній роботі обрано метод Мамдані. Цей метод інтуїтивно зрозумілий в вирішення та добре підходить для ручного вводу даних, має більш інтерпретовану базу правил та широкі визначення, що допоможе в вирішенні поставленої задачі.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ

3.1 Мова програмування Python

Python – це інтерпретована мова програмування високого рівня, розроблена в 1991 році Гвідо ван Россумом. Мова Python відрізняється простотою та зрозумілістю синтаксису, що дозволяє розробникам швидше створювати код та використовувати типові бібліотеки для вирішення завдань. Python підтримує різні парадигми програмування, включаючи процедурне, об'єктно-орієнтоване та функціональне програмування. Він також має багато можливостей, таких як динамічна типізація, автоматичне управління пам'яттю і підтримка багатопоточності. Python широко використовується в багатьох областях, таких як веб-розробка, наукові дослідження, аналіз даних та штучний інтелект. Наявність великої спільноти розробників створює можливість швидкого розвитку мови та створення різноманітних інструментів та бібліотек для спрощення роботи з Python. Таким чином, Python є потужною та універсальною мовою програмування, яка може використовуватися для вирішення різних завдань у різних галузях.

Мінімальні вимоги для Python 3.x:

- операційна система: Windows 7/8/10, MacOS X, Linux/UNIX;
- процесор: 1 ГГц чи більше;
- оперативна пам'ять: 1 Гб або більше;
- вільне місце на жорсткому диску: 100 Мб або більше.

Якщо ви використовуєте Windows, необхідно також переконатися, що у вас встановлена версія .NET Framework 4.5 та вище.

Переконавшись, що комп'ютер відповідає всім вище прописаним вимогам, було прийнято рішення використовувати мову програмування Python для реалізації задач кваліфікаційної роботи, так як вона має всі необхідні бібліотеки.

3.2 Вхідні та вихідні дані

Метою роботи є створення програми, що дозволить керувати мобільним роботом на відстані. Для прикладу розглянемо невеликий літаючий об'єкт (ЛО) який рухається з постійною швидкістю. ЛО обладнаний камерою, зображення з якої поступає на екран комп'ютера. Людина, що відповідає за керування роботом, оцінює і аналізує отриману картинку, що дозволяє їй робити висновки для подальшого керування роботом. Аналізує дистанцію до перешкоди та напрямок, кут до перешкоди. Ці дві змінні беремо за вхідні дані.

Вхідна лінгвістична змінна має наступну терм-множину: далеко зверху (NB), близько зверху (N), прямо (Z), близько нижче (P), далеко нижче (PB).

Вихідна лінгвістична змінна має наступну терм-множину: сильно вгору (NB), трохи вгору (N), вперед (Z), трохи нижче (P), сильно нижче (PB) (рис. 3.1).

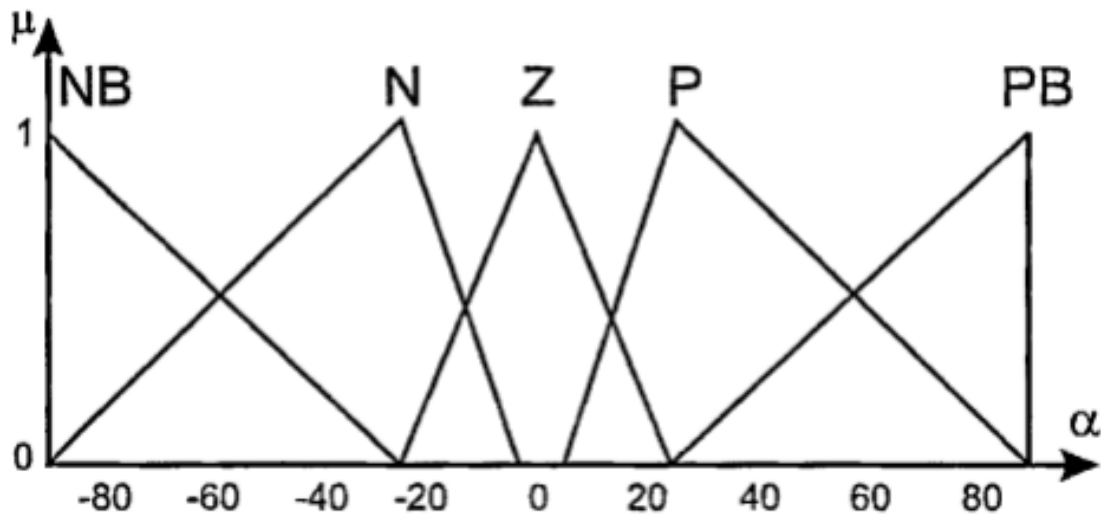


Рисунок 3.1 – Функції приналежності термів змінної.

Звідси випливають 5 основних правил:

- якщо об'єкт "далеко нижче", то "вперед";
- якщо об'єкт "близько нижче", то "трохи вгору";
- якщо об'єкт "прямо", то "сильно вгору" або "сильно вгору";

- якщо об'єкт "близько ліворуч", то " трохи нижче ";
- якщо об'єкт "далеко вгору ", то "вперед".

Кожній вихідній команді відповідає одна клавіша: сильно вгору (NB) – клавіша «I», трохи вгору (N) – клавіша «,», вперед (Z) – клавіша «J», трохи нижче(P) – клавіша «O», сильно нижче (PB) – клавіша «M».

Також для виключних ситуацій задамо клавіши ручного управління: вгору – клавіша «W», вниз – клавіша «S», вперед – клавіша «D», назад – клавіша «A».

3.3 Структура програми

3.3.1 Бібліотеки Python

Для написання програми, що буде моделювати управління мобільним роботом, використовуємо бібліотеку «Pygame».

Pygame – це набір модулів Python, призначених для написання відеоігор. Pygame додає функціональність на додаток до бібліотеки SDL. Це дозволяє створювати повнофункціональні ігри та мультимедійні програми мовою Python. Зокрема в кодї використовуються такі модулі:

- `pygame.display` – модуль для керування вікном дисплея. Використовується для встановлення заголовка вікна, ініціалізації дисплея та створення вікна гри;
- `pygame.time` – модуль для обробки функцій, пов'язаних із часом. Використовується для створення події таймера (`CREATE_ENEMY`), яка періодично створює перешкоди;
- `pygame.event` – модуль для обробки різних подій у ігровому циклі. Використовується для перевірки подій виходу (закриття вікна) і для запуску створення перешкод на основі події таймера;

– `pygame.draw` – модуль для малювання фігур на екрані. Використовується для малювання кіл, що представляють мобільного робота та перешкоди;

– `pygame.key` – модуль для отримання стану клавіш на клавіатурі. Використовується, щоб перевірити, які клавіші натиснуті, і на основі цього оновлюється позиція гравця;

– `pygame.image` – модуль для завантаження та обробки зображень. Використовується для завантаження фонового зображення (рис. 3.2) і масштабування його відповідно до вікна гри.

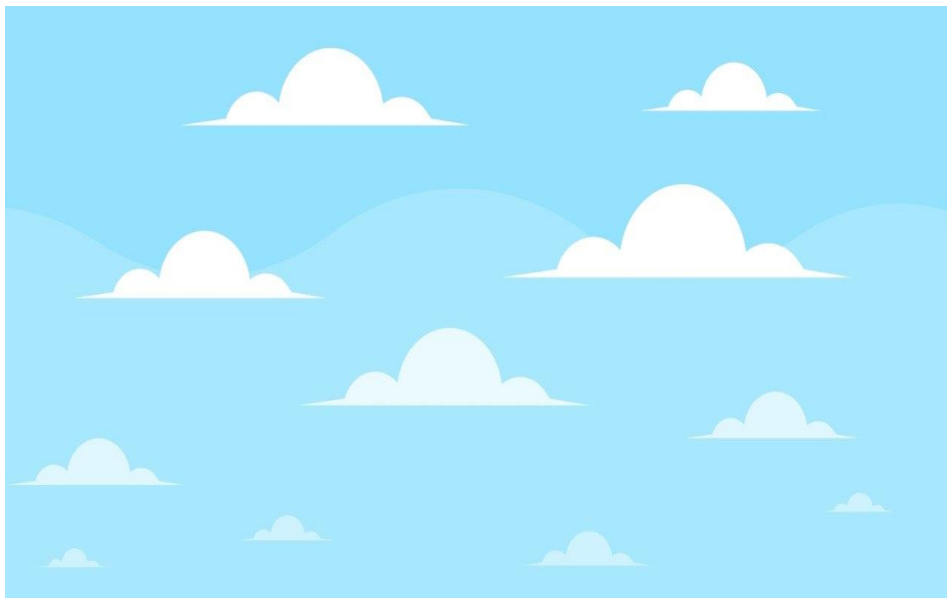


Рисунок 3.2 – Фонове зображення

Також в кодї використовується модуль `random`, який реалізує генератори псевдовипадкових чисел для різних розподілів. Для цілих чисел існує рівномірний вибір із діапазону. Для послідовностей існує рівномірний вибір випадкового елемента, функція для створення випадкової перестановки списку на місці та функція для випадкової вибірки без заміни. На реальній лінії існують функції для обчислення рівномірного, нормального (гауссівського), логарифмічного нормального, від'ємного експоненційного, гамма- та бета-розподілу. Для генерації розподілу кутів доступний розподіл фон Мізеса. В даному випадку він використовується для випадкового вибору позицій для перешкод, які будуть з'являтися на шляху мобільного робота.

3.3.2 Опис синтаксису програми

Для початку ми за допомогою команди «Import» підключаємо необхідні бібліотеки (рис. 3.3).

```
import pygame
import random
```

Рисунок 3.3 – Підключення бібліотек

Далі за допомогою модулів бібліотеки Pygame встановлюємо параметри розміру екрану, часу з'явлення перешкод та їх розміри. Також створюється вікно, де буде виведена гра та налаштовується годинник для контролю чистоти кадрів (рис. 3.4).

```
pygame.init()
pygame.display.set_caption('Visualization')
SCREEN_HEIGHT = 700
SCREEN_WIDTH = 1200
ENEMY_TIMER = 12000
ENEMY_SIZE = 60
PLAYER_SIZE = 60
screen = pygame.display.set_mode((SCREEN_WIDTH, SCREEN_HEIGHT))
clock = pygame.time.Clock()
dt = 0
```

Рисунок 3.4 – Налаштування програми

Завантажуємо зображення, що буде програватись на фоні, та ініціалізуємо дані для руху зображення (рис. 3.5).

```
bg = pygame.transform.scale(pygame.image.load('bg_04.jpg').convert(),
bgX = 0
bgX2 = bg.get_width()
bg_speed = 2
```

Рисунок 3.5 – Задання фону

Задаємо значення для вхідних та вихідних даних, використовуючи на термінологію алгоритму Мамдані (рис. 3.6). Також як вхідне значення задаємо яким чином може пересуватись мобільним робот. На яку відстань рахуючи пікселі буде переміщатись наше жовте коло.

```
lst = [100, 250, 350, 450, 600]
lst_pos = ["NB", "N", "Z", "P", "PB"]
lst_ans = ["Z", "P", "NB or PB", "N", "Z"]
```

Рисунок 3.6 – Вхідні та вихідні дані

Створюємо перешкоди, які будуть з'являтися випадковим чином у вигляді червоних кіл (рис. 3.7).

```
def create_enemy():
    position = random.randint(0, 4)
    enemy_center = pygame.Vector2(screen.get_width() + ENEMY_SIZE, lst[position])
    return enemy_center
```

Рисунок 3.7 – Створення перешкод

На початку ігрового циклу створюємо перешкоди (CREATE_ENEMY) таким чином, щоб ініціювати їх створення через регулярні проміжки часу (рис. 3.8).

```
CREATE_ENEMY = pygame.USEREVENT + 1
pygame.time.set_timer(CREATE_ENEMY, ENEMY_TIMER)
enemies = []
```

Рисунок 3.8 – Створення перешкод

Далі описуємо основний цикл (рис. 3.9).

```

while running:
    # Event handling
    for event in pygame.event.get():
        # Quit event
        if event.type == pygame.QUIT:
            running = False
        # Enemy creation event
        if event.type == CREATE_ENEMY:
            enemies.append(create_enemy())
    ...

- The game loop processes events, such as quitting the game or creating enemies.

```

Рисунок 3.9 – Основний цикл

Прописуємо створення фону та моделей що юдуть відігравати роль мобільного робота та перешкод (рис. 3.10).

```

# Drawing the background
screen.blit(bg, (bgX, 0))
screen.blit(bg, (bgX2, 0))

# Drawing enemies and player
for enemy_center in enemies:
    pygame.draw.circle(screen, "red", enemy_center, ENEMY_SIZE)
pygame.draw.circle(screen, "yellow", player_pos, PLAYER_SIZE)

```

Рисунок 3.10 – Задання візуальної складової програми

Далі пишемо основну складову програми, де прописуємо для кожної клавіши свою команду на основі вхідних і вихідних даних. В даному випадку задаємо для кожної команди, в яку частину екрану переміщається керований об'єкт (рис. 3.11).

```

keys = pygame.key.get_pressed()
if keys[pygame.K_w]:
    player_pos.y -= 300 * dt
# ... (similar blocks for other keys)

```

Рисунок 3.11 – Задання команд

На основі даних, введених в програму користувачем, змінюється положення об'єкту на екрані відповідно вище заданим параметрам (рис. 3.12).

```
if (tmp != tmp1):  
    # Print statements based on player input  
    # Update tmp1 for comparison in the next iteration  
pygame.display.flip()  
dt = clock.tick(60) / 1000
```

Рисунок 3.12 – Керування об'єктом

Коли ігровий цикл буде завершений, програма вимикається автоматично.

Таким чином була створена програма на основі якої ми можемо показати принцип управління мобільним роботом за допомогою алгоритму Мамдані.

3.4 Результати роботи програми

Запускаємо нашу програму з папки де знаходиться програмне забезпечення, що допомагає відкривати файли написані мовою Python. В данному випадку я користувалась програмою «Python 3.12». Початковий екран (рис. 3.13) візуалізує як жовте коло, наш мобільний робот, рухається.

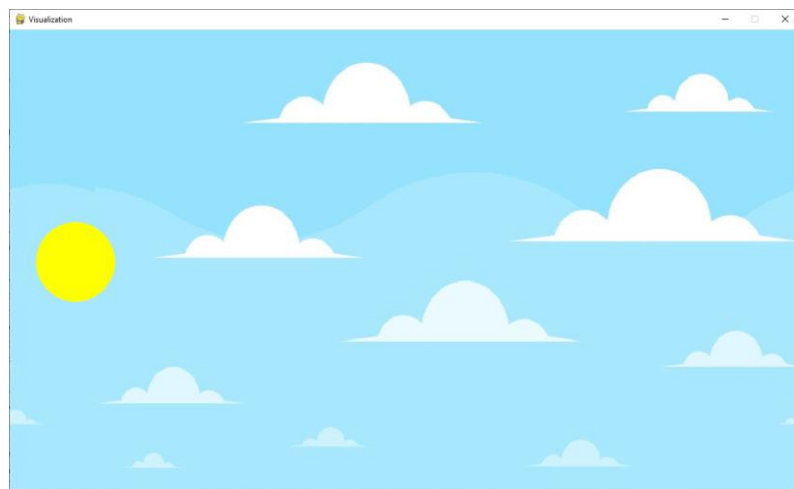


Рисунок 3.13 – Початковий екран

Далі, в випадковому порядку починають з'являтися перешкоди (рис. 3.14), червоні кола.

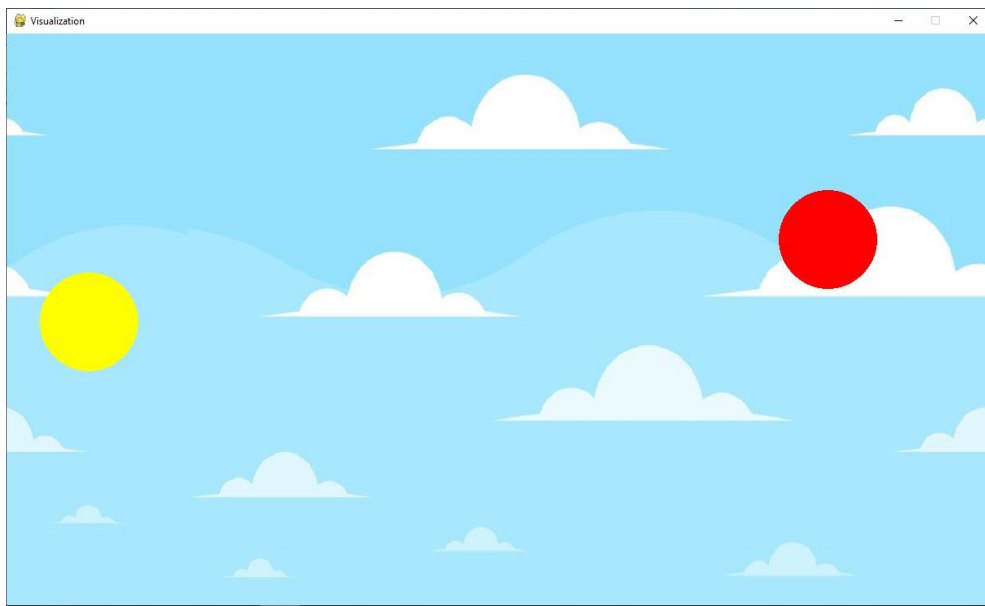


Рисунок 3.14 – Перша перешкода

Ми аналізуємо отриману картинку і приходимо до висновку, що перешкода знаходиться «трохи зверху», тому необхідно відати команду мобільному роботу, щоб він рухався «трохи вниз» або «сильно вниз». Вибираємо команду «трохи вниз» (рис. 3.15).

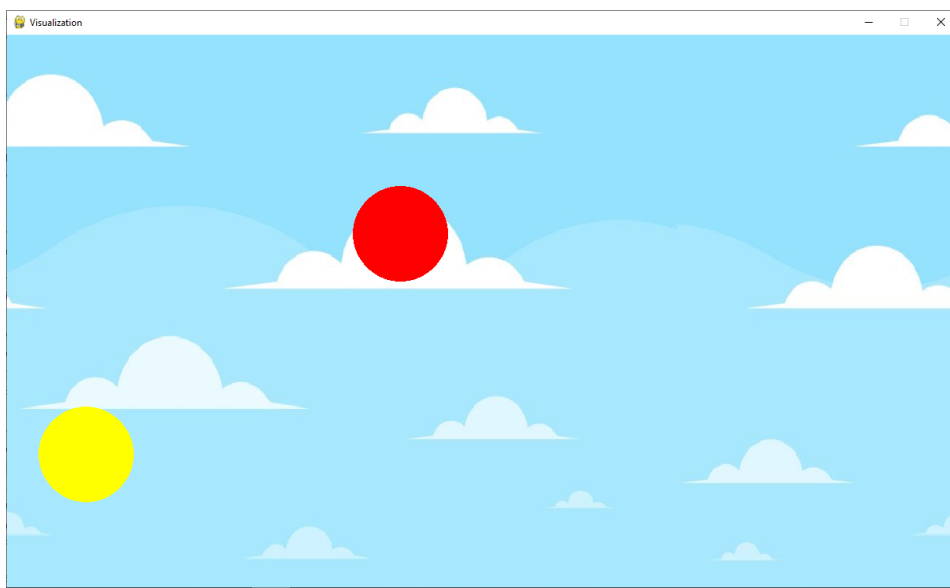


Рисунок 3.15 – Мобільний робот рухається трохи вниз.

Таким чином аналізуючи отриману картинку ми вибираємо необхідну команду для нашого робота. Якщо перешкода «сильно зверху» або «сильно знизу» – вибираємо рухатись прямо (рис. 3.16).

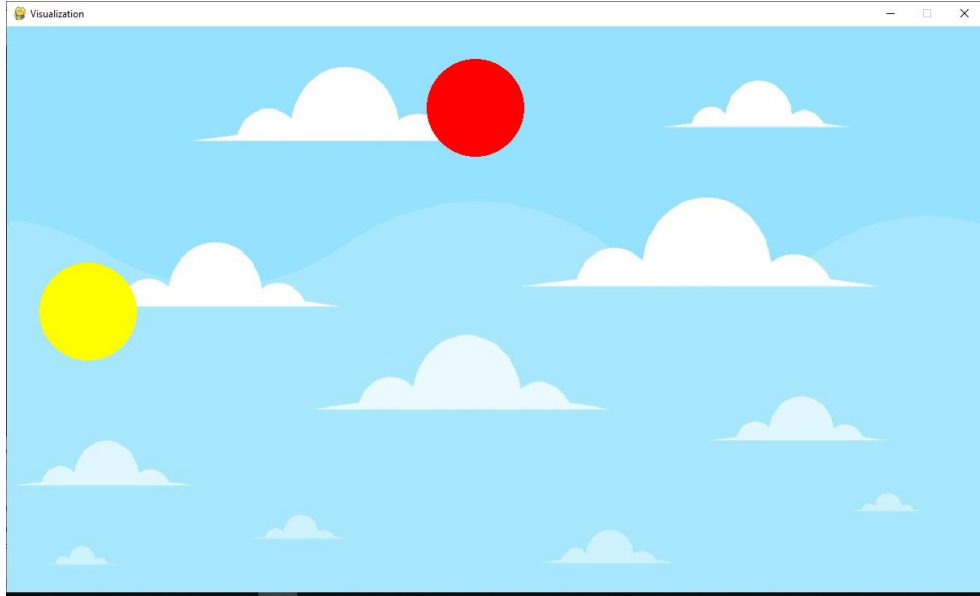


Рисунок 3.16 – Перешкода «сильно зверху»

Якщо перешкода «трохи знизу» – вибираємо рухатись «трохи ввєрх» або «сильно ввєрх» (рис. 3.17).

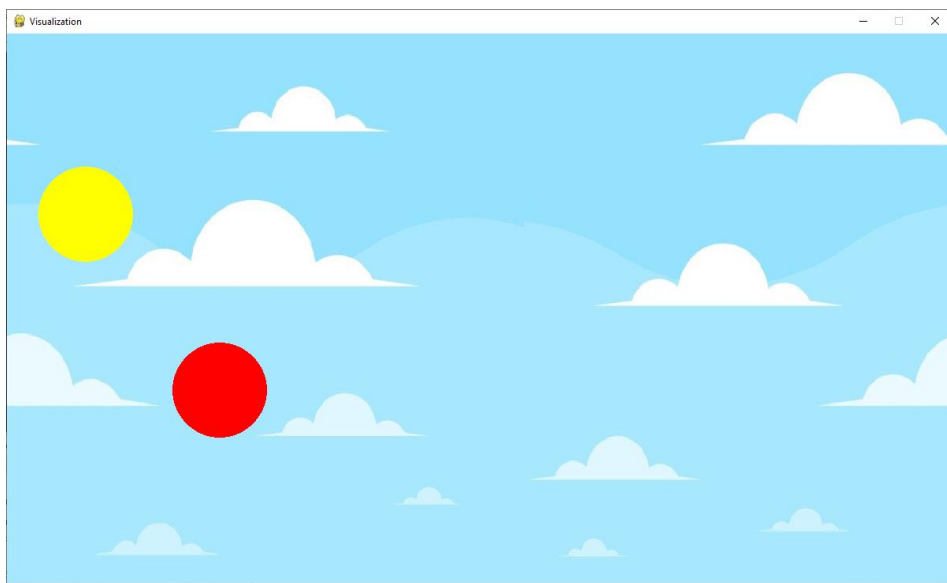


Рисунок 3.17 – Перешкода «трохи знизу»

Якщо перешкода «прямо» – вибираємо рухатись «сильно ввверх» або «сильно вниз». Я вибрала рухатись «сильно вниз» (рис. 3.18).

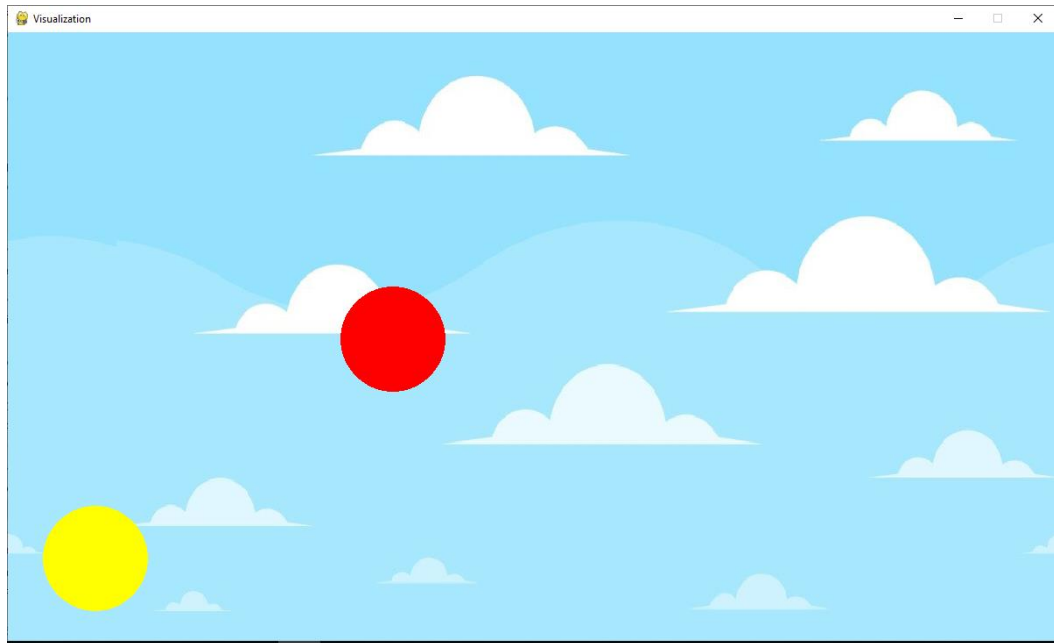


Рисунок 3.18 – Перешкода «прямо»

Висновки за розділом 3

За допомогою мови Python була написана програма, яка моделює роботу мобільного роботу. Робот повинен мати штучний інтелект, що допоможе йому самостійно приймати рішення спираючись на нечітку логіку і базу правил. На разі в розділі представлено моделювання такого роботу через гру, де замість штучного інтелекту рішення приймає людина, яка керує роботом.

4 РЕЗУЛЬТАТИ ОБЧИСЛЮВАЛЬНОГО ЕКСПЕРИМЕНТУ ТА ЇХ АНАЛІЗ

4.1 Основні позначення розділу

A – "напрямок", B – "дистанція", C – "кермовий кут".

A_1 – "праворуч", A_2 – "прямо", A_3 – "ліворуч".

B_1 – "дуже близько", B_2 – "близько", B_3 – "середньо", B_4 – "далеко".

D_1 – "різко вліво", D_2 – "вліво", D_3 – "прямо", D_4 – "вправо", D_5 – "різко вправо".

E_{ij} , $i = 1, 2, 3$, $j = 1, 2, 3, 4$ – нечітке висловлювання (ситуація). «Напрямок» дорівнює A_i ; «дистанція» дорівнює B_j (табл. 4.1).

Таблиця 4.1 – Таблиця нечітких правил у прийнятих позначеннях

Напрямок	Дистанція			
	B_1	B_2	B_3	B_4
A_1	E_{11}	E_{12}	E_{13}	E_{14}
A_2	E_{21}	E_{22}	E_{23}	E_{24}
A_3	E_{31}	E_{32}	E_{33}	E_{34}

4.2 Обчислювальний експеримент для першого випадку

Нехай із сенсорів отримані такі дані: «дистанція» = 80 см, "напрямок" = $-\frac{\pi}{16}$.

Рішення. Функцію приналежності цього нечіткого висловлювання при заданих значеннях φ та d вхідних змінних позначимо $\mu_{E_{ij}}(\varphi, d)$.

Позначимо $\mu_{A_i}(\varphi)$ – значення функції приналежності нечіткої множини, що формалізує терм A_i лінгвістичної змінної «напрямок» при її значенні, що дорівнює φ . Аналогічно $\mu_{B_j}(d)$.

Маємо (рис. 1.9):

$$\mu_{B_1}(80) = 0,30, \mu_{B_2}(80) = 0,52, \mu_{B_3}(80) = 0,00, \mu_{B_4}(80) = 0,00.$$

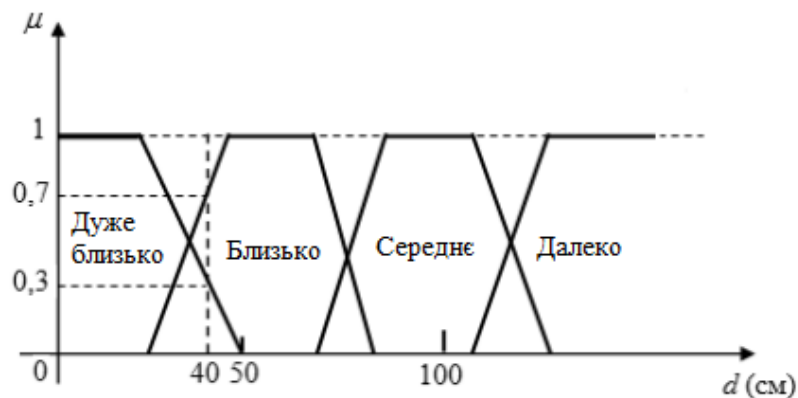


Рисунок 1.9 – Функції приналежності термів змінної «дистанція»

Аналізуючи рис. 1.10 бачимо, що

$$\mu_{A_1}\left(-\frac{\pi}{16}\right) = 0,00, \mu_{A_2}\left(-\frac{\pi}{16}\right) = 0,50, \mu_{A_3}\left(-\frac{\pi}{16}\right) = 0,25.$$

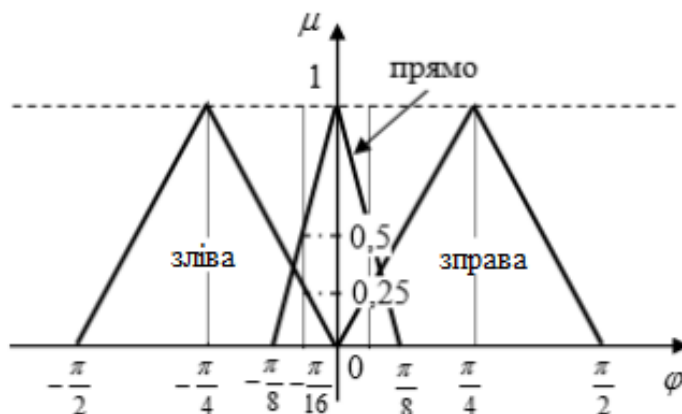


Рисунок 1.10 – Функції приналежності термів змінної «напрямок»

З визначення нечіткої логічної операції I (\wedge) випливає, що

$$\mu_{E_{ij}}(\varphi, d) = \min\{\mu_{A_i}(\varphi), \mu_{B_j}(d)\}.$$

У таблиці 4.2 вказано значення функцій приналежності $\mu_{E_{ij}}(\varphi, d)$ при $\varphi = -\frac{\pi}{16}$ і $d = 80\text{см}$.

Таблиця 4.2 – Значення функції приналежності $\mu_{E_{ij}}\left(-\frac{\pi}{16}, 80\right)$

	$\mu_{B_j}\left(-\frac{\pi}{16}\right)$	B_1	B_2	B_3	B_4
$\mu_{A_i}(80)$		0,3	0,52	0	0
A_1	0	0	0	0	0
A_2	0,5	0,3	0,5	0	0
A_3	0,25	0,25	0,25	0	0

З таблиці 4.1 випливає, що можна розглядати лише чотири ситуації: $A_2 - B_1$, $A_2 - B_2$, $A_3 - B_1$, $A_3 - B_2$. У прийнятих позначеннях це відповідно E_{21} , E_{22} , E_{31} , E_{32} . Вони означають: різко вліво (різко вправо), вліво (вправо), праворуч, праворуч. Так як перешкода знаходиться зліва, то з двох можливостей у ситуаціях E_{21} , E_{22} використовуватимемо повороти вправо. Ситуації E_{21} відповідає терм D_5 лінгвістичної змінної «кермовий кут», ситуації E_{22} відповідає терм D_4 , ситуації E_{31} – терм D_4 , ситуації E_{32} – D_4 .

У ситуаціях E_{22} , E_{31} , E_{32} вихідний терм той самий – D_4 . Відповідно до формули нечіткої логічної операції АБО (\vee), ступінь приналежності $\mu_{D_4}\left(-\frac{\pi}{16}, 80\right)$ вхідного вектору $\left(-\frac{\pi}{16}, 80\right)$ вихідному терму D_4 дорівнює

максимальному із значень істинності нечітких висловлювань E_{22} , E_{31} , E_{32} , тобто (табл. 4.1)

$$\mu_{D_4}\left(-\frac{\pi}{16}, 80\right) = \max\{0,5, 0,25, 0,25\} = 0,5.$$

Цей максимум досягається для ситуації E_{22} , тому надалі з трьох ситуацій E_{22} , E_{31} , E_{32} достатньо розглядати лише ситуацію E_{22} . Отже, нас цікавлять лише дві ситуації E_{21} та E_{22} .

Залишилося визначити кут повороту керма. Знаходити цей кут будемо приблизно. Для цього змінюватимемо кут повороту керма від 0 до $\frac{\pi}{2}$ з кроком $-\frac{\pi}{16}$. Нагадаємо, що кут повороту керма ми позначили літерою ψ . Позначимо функцію приналежності терму D_k , $k = 1, 2, 3, 4, 5$, лінгвістичної змінної «кермовий кут» через $\mu_{D_k}(\psi)$.

Результати обчислень зведені у таблиці 4.3 та 4.4.

Таблиця 4.3 заповнена на підставі графіків функцій приналежності термів вихідної змінної (рис. 1.11).

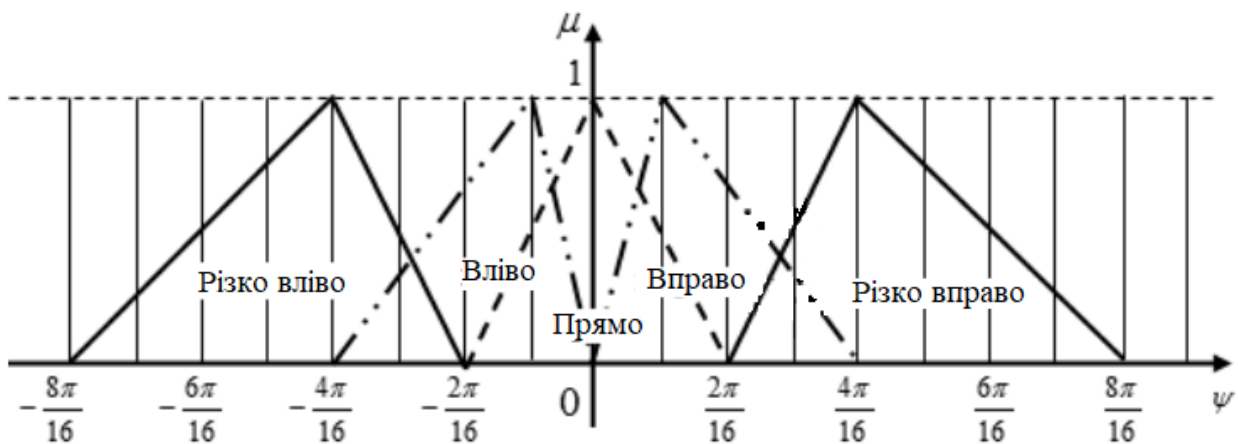


Рисунок 1.11 – Функції приналежності термів вихідної змінної

Таблиця 4.3 – Значення функцій приналежності $\mu_{D_k}(\psi)$ (ступінь впевненості в тому, що, ухваливши рішення D_k , слід повернути на кут ψ)

ψ	0	$\frac{\pi}{16}$	$\frac{2\pi}{16}$	$\frac{3\pi}{16}$	$\frac{4\pi}{16}$	$\frac{5\pi}{16}$	$\frac{6\pi}{16}$	$\frac{7\pi}{16}$	$8\frac{\pi}{16}$
$\mu_{D_4}(\psi)$	0	0,67	0,33	0	0	0	0	0	0
$\mu_{D_5}(\psi)$	0	0	0	0,5	1	0,75	0,5	0,25	0

Введемо ще два позначення. Через $\mu_{ijk}(\varphi, d, \psi)$ позначимо функцію належності нечіткого висловлювання [(значення терму A_i дорівнює φ) I (значення терму B_j дорівнює d) I (значення терму D_k дорівнює ψ)], тоді

$$\mu_{ijk}(\varphi, d, \psi) = \min\{\mu_{A_i}(\varphi), \mu_{B_j}(d), \mu_{D_k}(\psi)\}.$$

Значення функції $\mu_{ijk}(\varphi, d, \psi)$ для двох ситуацій, що розглядаються, вказані в другому і третьому рядках таблиці 4.4.

Таблиця 4.4 – Значення функції приналежності

ψ	0	$\frac{\pi}{16}$	$\frac{2\pi}{16}$	$\frac{3\pi}{16}$	$\frac{4\pi}{16}$	$\frac{5\pi}{16}$	$\frac{6\pi}{16}$	$\frac{7\pi}{16}$	$8\frac{\pi}{16}$
$\mu_{224}\left(-\frac{\pi}{16}, 80, \psi\right)$	0,00	0,50	0,33	0,00	0,00	0,00	0,00	0,00	0,00
$\mu_{215}\left(-\frac{\pi}{16}, 80, \psi\right)$	0,00	0,00	0,00	0,30	0,30	0,30	0,30	0,25	0,00
$\mu_D\left(-\frac{\pi}{16}, 80, \psi\right)$	0,00	0,5	0,33	0,33	0,30	0,30	0,30	0,25	0,00

Через $\mu_D(\varphi, d, \psi)$ позначимо функцію приналежності лінгвістичної змінної «кермовий кут» за умови, що значенням лінгвістичної змінної «напрямок» є кут, рівний φ , а значенням лінгвістичної змінної «дистанція» є d :

$$\mu_D(\varphi, d, \psi) = \max_{(i,j,k)} \mu_{ijk}(\varphi, d, \psi).$$

Значення цієї функції вказані у четвертому рядку таблиці 4.4, вони дорівнюють максимальному числу з чисел відповідної графи.

Перший та четвертий рядки таблиці 4.4 задають функцію приналежності нечіткої множини:

$$\text{«кермовий кут за умови, що } \varphi = -\frac{\pi}{16} \text{ і } d = 80\text{»}.$$

Залишилося здійснити дефазифікацію цієї нечіткої множини, тобто перетворити її у чітке число (визначити, який же кут треба повернути). Дефазифікація методом «центр ваги» дає:

$$\begin{aligned} & \frac{0 + \frac{\pi}{16} \cdot 0,5 + \frac{2\pi}{16} \cdot 0,33 + \frac{3\pi}{16} \cdot 0,33 + \frac{4\pi}{16} \cdot 0,3 + \frac{5\pi}{16} \cdot 0,3 + \frac{6\pi}{16} \cdot 0,3 + \frac{7\pi}{16} \cdot 0,25 + \frac{8\pi}{16} \cdot 0}{0 + 0,50 + 0,33 + 0,33 + 0,30 + 0,30 + 0,30 + 0,25 + 0} = \\ & = \frac{8,4}{2,31} \cdot \frac{\pi}{16} \cong 3,64 \cdot \frac{\pi}{16}. \end{aligned}$$

Отже, повернути треба на кут $3,64 \cdot \frac{\pi}{16}$.

4.3 Обчислювальний експеримент для другого випадку

Нехай із сенсорів отримані такі дані: «дистанція» = 40 см, "напрямок" = $\frac{\pi}{8}$.

Рішення. Функцію належності цього нечіткого висловлювання при заданих значеннях φ та d вхідних змінних позначимо $\mu_{E_{ij}}(\varphi, d)$.

Маємо (рис. 1.9):

$$\mu_{B_1}(40) = 0,30, \mu_{B_2}(40) = 0,70, \mu_{B_3}(40) = 0,00, \mu_{B_4}(40) = 0,00.$$

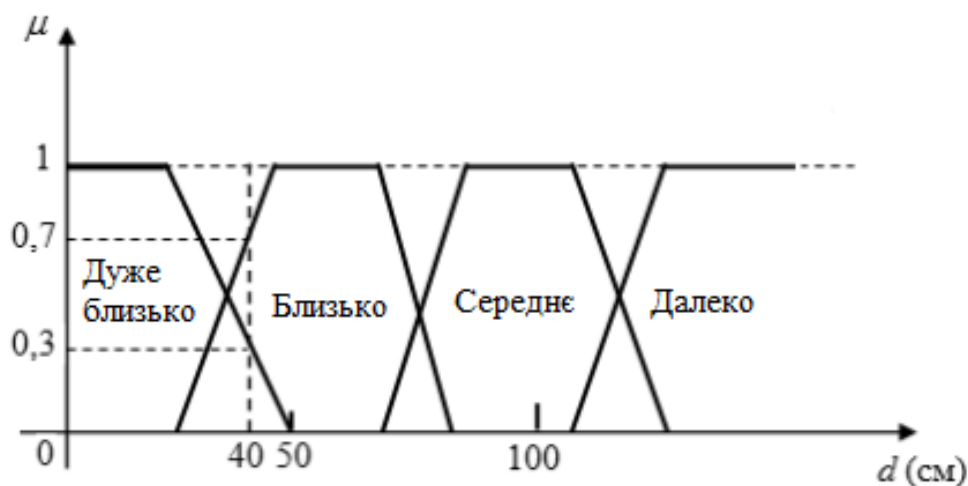


Рисунок 1.9 – Функції приналежності термів змінної «дистанція»

Аналізуючи рисунок 1.10 бачимо, що

$$\mu_{A_1}\left(\frac{\pi}{8}\right) = 0,00, \mu_{A_2}\left(\frac{\pi}{8}\right) = 0,50, \mu_{A_3}\left(\frac{\pi}{8}\right) = 0,00.$$

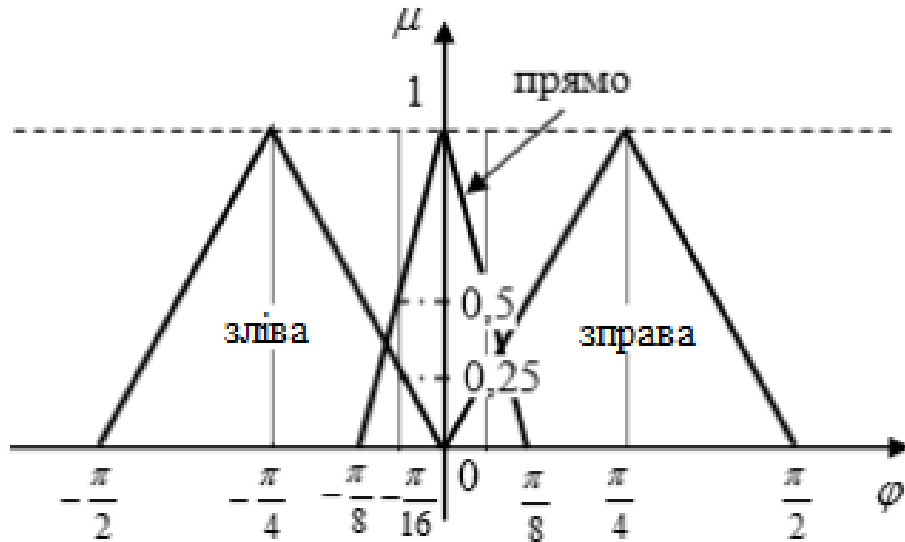


Рисунок 1.10 – Функції приналежності термів змінної «напрямок»

З визначення нечіткої логічної операції I (\wedge) випливає, що

$$\mu_{E_{ij}}(\varphi, d) = \min\{\mu_{A_i}(\varphi), \mu_{B_j}(d)\}.$$

У таблиці 4.5 вказано значення функцій приналежності $\mu_{E_{ij}}(\varphi, d)$ при

$$\varphi = \frac{\pi}{8} \text{ і } d = 40 \text{ см.}$$

Таблиця 4.5 – Значення функції приналежності $\mu_{E_{ij}}\left(\frac{\pi}{8}, 40\right)$

	$\mu_{B_j}\left(\frac{\pi}{8}\right)$	B_1	B_2	B_3	B_4
$\mu_{A_i}(40)$		0,3	0,70	0	0
A_1	0	0	0	0	0
A_2	0,5	0,3	0,5	0	0
A_3	0	0	0	0	0

З таблиці 4.1 випливає, що можна розглядати лише 2 ситуації: $A_2 - B_1$, $A_2 - B_2$. У прийнятих позначеннях це відповідно E_{21} , E_{22} . Вони означають: різко вліво (різко вправо), вліво (вправо). Так як перешкода знаходиться справа, то з двох можливостей у ситуаціях E_{21} , E_{22} використовуватимемо повороти вправо. Ситуації E_{21} відповідає терм D_5 лінгвістичної змінної «кермовий кут», ситуації E_{22} відповідає терм D_4 .

У ситуації E_{22} вихідний терм той самий – D_4 . Відповідно до формули нечіткої логічної операції АБО (\vee), ступінь приналежності $\mu_{D_4}\left(\frac{\pi}{8}, 40\right)$ вхідного вектора $\left(\frac{\pi}{8}, 40\right)$ вихідному терму D_4 дорівнює максимальному із значень істинності нечітких висловлювань, тобто (табл. 4.1)

$$\mu_{D_4}\left(\frac{\pi}{8}, 40\right) = 0,5.$$

Отже, нас цікавлять лише дві ситуації E_{21} та E_{22} .

Залишилося визначити кут повороту керма. Знаходити цей кут будемо приблизно. Для цього змінюватимемо кут повороту керма від 0 до $\frac{\pi}{2}$ з кроком $\frac{\pi}{8}$. Нагадаємо, що кут повороту керма ми позначили літерою ψ . Позначимо функцію приналежності терму D_k , $k = 1, 2, 3, 4, 5$, лінгвістичної змінної «кермовий кут» через $\mu_{D_k}(\psi)$.

Результати обчислень зведені у таблиці 4.6 та 4.7.

Таблиця 4.6 заповнена на підставі графіків функцій приналежності термів вихідної змінної (рис. 1.11).

Таблиця 4.6 – Значення функцій приналежності $\mu_{D_k}(\psi)$ (ступінь впевненості в тому, що, ухваливши рішення D_k , слід повернути на кут ψ)

ψ	0	$\frac{\pi}{8}$	$\frac{2\pi}{8}$	$\frac{3\pi}{8}$	$\frac{4\pi}{8}$	$\frac{5\pi}{8}$	$\frac{6\pi}{8}$	$\frac{7\pi}{8}$	$8\frac{\pi}{8}$
$\mu_{D_4}(\psi)$	0	0,70	0	0	0	0	0	0	0
$\mu_{D_5}(\psi)$	0	0	1	0,5	0	0	0	0	0

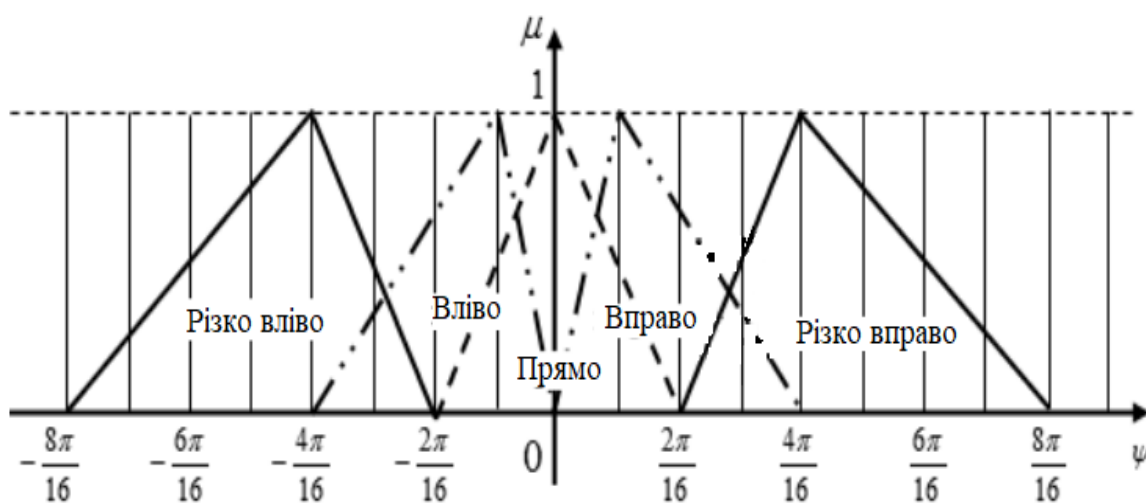


Рисунок 1.11 – Функції приналежності термів вихідної змінної

Введемо ще два позначення. Через $\mu_{ijk}(\varphi, d, \psi)$ позначимо функцію належності нечіткого висловлювання [(значення терму A_i дорівнює φ) І (значення терму B_j дорівнює d) І (значення терму D_k дорівнює ψ)], тоді

$$\mu_{ijk}(\varphi, d, \psi) = \min \{ \mu_{A_i}(\varphi), \mu_{B_j}(d), \mu_{D_k}(\psi) \}.$$

Значення функції $\mu_{ijk}(\varphi, d, \psi)$ для двох ситуацій, що розглядаються, вказані в другому і третьому рядках таблиці 4.7.

Таблиця 4.7 – Значення функції приналежності

ψ	0	$\frac{\pi}{8}$	$\frac{2\pi}{8}$	$\frac{3\pi}{8}$	$\frac{4\pi}{8}$	$\frac{5\pi}{8}$	$\frac{6\pi}{8}$	$\frac{7\pi}{8}$	$8\frac{\pi}{8}$
$\mu_{224}\left(\frac{\pi}{8}, 40, \psi\right)$	0,00	0,50	0,00	0,00	0,00	0,00	0,00	0,00	0,00
$\mu_{215}\left(\frac{\pi}{8}, 40, \psi\right)$	0,00	0,00	0,30	0,30	0,00	0,00	0,00	0,00	0,00
$\mu_D\left(\frac{\pi}{8}, 40, \psi\right)$	0,00	0,50	0,30	0,30	0,00	0,00	0,00	0,00	0,00

Через $\mu_D(\varphi, d, \psi)$ позначимо функцію приналежності лінгвістичної змінної «кермовий кут» за умови, що значенням лінгвістичної змінної «напрямок» є кут, рівний φ , а значенням лінгвістичної змінної «дистанція» є d :

$$\mu_D(\varphi, d, \psi) = \max_{(i,j,k)} \mu_{ijk}(\varphi, d, \psi).$$

Значення цієї функції вказані у четвертому рядку таблиці 4.4, вони дорівнюють максимальному числу з чисел відповідної графі.

Перший та четвертий рядки таблиці 4.7 задають функцію приналежності нечіткої множини:

$$\text{«кермовий кут за умови, що } \varphi = \frac{\pi}{8} \text{ і } d = 40\text{»}.$$

Залишилося здійснити дефазифікацію цієї нечіткої множини, тобто перетворити їх у чітке число (визначити, який же кут треба повернути). Дефазифікація методом «центр ваги» дає:

$$\frac{0 \cdot 0 + \frac{\pi}{8} \cdot 0,50 + \frac{2\pi}{8} \cdot 0,30 + \frac{3\pi}{8} \cdot 0,30 + \frac{4\pi}{8} \cdot 0 + \frac{5\pi}{8} \cdot 0 + \frac{6\pi}{8} \cdot 0 + \frac{7\pi}{8} \cdot 0 + \frac{8\pi}{8} \cdot 0}{0 + 0,50 + 0,30 + 0,30 + 0 + 0 + 0 + 0 + 0} =$$

$$= \frac{2}{1,1} \cdot \frac{\pi}{8} \cong 1,82 \cdot \frac{\pi}{8}.$$

Отже, повернути треба на кут $1,82 \cdot \frac{\pi}{8}$.

Висновки за розділом 4

У четвертому розділі наведено обчислювальні експерименти для різних вхідних даних. На практиці переконуємось в простоті і доступності методу Мамдані, що дозволяє нам визначити точний кут, під яким робот зможе обійти перешкоду попереду. Розрахунки базуються на базі даних, що дозволяє роботу швидко приймати рішення в умовах реального часу.

ВИСНОВКИ

Підсумовуючи, можна сказати, що керування мобільним роботом представляє собою динамічний і багатогранний процес, який потребує комплексного підходу для навігації у складнощах сучасної робототехніки. Як ми досліджували в цій роботі, такі фактори, як відстань, аналіз отриманих даних і адаптивність, відіграють ключову роль в ефективному функціонуванні мобільних роботів у різних програмах. Удосконалення сенсорних технологій, штучного інтелекту та автономних систем відкрили нові шляхи для розширення можливостей цих роботів, зробивши їх цінними активами в різних галузях промисловості, від виробництва та логістики до охорони здоров'я та дослідження.

Незважаючи на досягнутий прогрес, все ще є проблеми, які потрібно подолати, зокрема проблеми, пов'язані з безпекою, етичними міркуваннями та інтеграцією роботів у існуючі робочі процеси. Крім того, постійний розвиток технологій вимагає гнучкої та адаптивної системи управління, яка може йти в ногу з інноваціями.

Наукова та академічна важливість цієї роботи полягає в її внеску у зростаючий обсяг знань у сфері керування робототехнікою. Результати дослідження створюють основу для майбутніх досліджень, заохочуючи до глибшого вивчення нових технологій і методологій у цій галузі. Виявлені напрямки вдосконалення та подальших досліджень підкреслюють постійну актуальність і доцільність продовження досліджень теми. Швидкий розвиток технологій вимагає постійного вивчення практик керування мобільними роботами, щоб бути в курсі прогресу та вирішувати нові виклики.

Управління мобільними роботами – це не просто технічне завдання, а багатовимірне дослідження, яке передбачає гармонійну інтеграцію передових технологій із глибоким усвідомленням ширших наслідків. Успішне управління мобільними роботами обіцяє революцію в промисловості та покращення якості нашого життя, але воно вимагає постійної відданості, співпраці та ретельного розгляду етичних і соціальних норм.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Kalinichenko A. Control of the mobile robot. *Conference Proceedings of II International Scientific & Practical Conference «Learning & Teaching: after War and during Peace»*. Kharkiv : H. S. Skovoroda Kharkiv National Pedagogical University, 2023. P. 200.
2. Klir G. J., Yuan B. Fuzzy sets and fuzzy logic theory and applications. Prentice Hall P T R, New Jersey. 1995. 591 p.
3. Python. URL: <https://docs.python.org> (дата звернення: 20.12.2023)
4. Зак Ю. О. Принятие решений в условиях нечетких и размытых данных: Fuzzy-технологии. Москва : Книжный дім «ЛИБРОКОМ», 2013. 352 с.
5. Калініченко А. В. Управління мобільним роботом. *27-й Міжнародний молодіжний форум «Радіоелектроніка і молодь у XXI столітті»* : зб. матеріалів форуму. Т. 7. Харків : ХНУРЕ, 2023. С. 143-144.
6. Костигов А. М., Даденков Д. А., Каверин А. А. Нечеткая система управления мобильным роботом. *Современные проблемы науки и образования*. 2019 № 5. С. 136-142.
7. Мазорчук М. С., Симонова К. А., Греков Л. Д. Применение методов и моделей нечеткой логики для моделирования экономических процессов. *Системи обробки інформації*. 2007. №9. С.159-162.
8. Поліщук М. М., Писаренко С. Г. Система управління мобільним роботом з підвищеним ступенем інтелектуальності. *ЛОГОΣ. Мистецтво наукової думки*. 2019. №3. С. 101-103.
9. Програмування числових методів мовою Python / Анісімов А. В., Дорошенко А. Ю., Погорілий С. Д., Дорогий Я. Ю.; за ред. Анісімова А. В. Київ : Видавничо-поліграфічний центр "Київський університет", 2014. 640 с.
10. Силич В. А., Силич М. П., Аксенов С. В. Алгоритм построения нечеткой системы логического вывода Мамдани, основанный на анализе плотности обучающих примеров. *Доклады ТУСУРа. Серия: Управление, вычислительная техника и информатика*. 2013. № 3. С. 76-82.

11. Флегонтов А. В. Вилков В. Б., Черных А. К. Моделирование задач принятия решений при нечетких исходных данных. Москва : Лань, 2020. 332 с.