

Міністерство освіти і науки України
Український державний університет залізничного транспорту
Факультет інформаційно-керуючих систем та технологій
Кафедра інформаційних технологій

До захисту допустити
Завідувач кафедри ІТ,
д.т.н., проф., Каргін А.О.



24 червня 2026 р.

ПОЯСНЮВАЛЬНА ЗАПИСКА

до кваліфікаційної роботи
здобувача першого (бакалаврського) рівня вищої освіти

тема: Система керування рухом по азимуту колісного робота

Студент 3-го курсу, скороченого терміну навчання
132- ТШІ-Д23 групи
Галузь знань 12 - Інформаційні технології
Спеціальність 126 - Інформаційні системи та технології
Освітньо-професійна програма - Технології штучного інтелекту)

Dob

Довженко В'ячеслав Сергійович
(ПІБ)

Керівник Каргін А.О.
(ПІБ)

д.т.н, професор, професор кафедри ІТ УкрДУЗТ
(науковий ступінь, вчене звання, посада)

Рецензент Ковтун І.В.
(ПІБ)

к.т.н., доцент, доцент кафедри ТЗ УкрДУЗТ
(науковий ступінь, вчене звання, посада)

Національна шкала

відмінно

Кількість балів:

97

Оцінка (ECTS)

A

м. Харків - 2026 рік

Лист оцінок кваліфікаційної роботи здобувача Довженка В. С.

Перший (бакалаврський) рівень вищої освіти

(прізвище в родовому відмінку та ініціали)

Галузь знань 12 - Інформаційні технології

Спеціальність - 126 Інформаційні системи та технології

Освітньо-професійна програма - Технології штучного інтелекту

<i>КРИТЕРІЇ ОЦІНКИ</i>	макс. оцінка	оцінка керівн.	оцінка комісії
ОЦІНКА ЕТАПУ ПІДГОТОВКИ РОБОТИ	8 б.	8 б.	8 б.
дотримання графіка консультацій з керівником кваліфікаційної роботи	2 б.	2 б.	2 б.
дотримання термінів підготовки звіту	2 б.	2 б.	2 б.
дотримання термінів практичної реалізації програм	2 б.	2 б.	2 б.
самостійність і ініціативність	2 б.	2 б.	2 б.
ОЦІНКА ЗА ЗВІТ - ВІДПОВІДНІСТЬ ВИМОГАМ ДО ЗВІТУ	59 б.:	59 б.:	59 б.:
(структура звіту - наявність обов'язкових розділів і підрозділів):	47 б.	47 б.	47 б.
ТИТУЛЬНИЙ ЛИСТ ПОЯСНЮВАЛЬНОЇ ЗАПИСКИ	1	1	1
ЛИСТ ОЦІНОК	1	1	1
ЛИСТ ЗАВДАННЯ	1	1	1
ДЕКЛАРАЦІЯ	1	1	1
РЕФЕРАТ (2 примірника - укр., англ.)	2	2	2
ЗМІСТ	1	1	1
ВСТУП	2	2	2
1 ПОСТАНОВКА ЗАДАЧІ	2	2	2
2 ОПИС ПРЕДМЕТНОЇ ОБЛАСТІ І ОГЛЯД ПІДХОДІВ ДО ВИРІШЕННЯ ЗАВДАННЯ	5	5	5
3 ПРОЄКТУВАННЯ СИСТЕМИ КЕРУВАННЯ РУХОМ ПО АЗИМУТУ КОЛІСНОГО РОБОТА			
3.1 Концептуальна модель системи	4	4	4
3.2 Логічна модель системи. Моделювання в Mental Modeler	5	5	5
3.3 Логічна модель системи. Моделювання в MATLAB	6	6	6
3.4 Фізична модель системи	4	4	4
4 АПАРАТНО-ПРОГРАМНИЙ ПРОТОТИП СИСТЕМИ			
4.1 Топологія та компоненти апаратної частини прототипу системи	3	3	3
4.2 Архітектура та компоненти програмної частини прототипу системи	3	3	3
4.3 Опис даних та бібліотек підтримки компонентів системи	3	3	3
ВИСНОВКИ	2	2	2
СПИСОК ПОСИЛАНЬ	2	2	2
ДОДАТКИ	2	2	2
обсяг роботи (не менше 50 і не більше 70)	3 б.	3 б.	3 б.
оформлення (тексту, малюнків, таблиць, формул, додатків, посилань)	6 б.	6 б.	6 б.
ОЦІНКА ЗА ДОПОВІДЬ:	15 б.	15 б.	15 б.
структура доповіді, оформлення слайдів	5 б.	5 б.	5 б.
знання матеріалу роботи і рівень відповіді на питання	5 б.	5 б.	5 б.
дотримання регламенту, активність	5 б.	5 б.	5 б.
ОЦІНКА ЗА ПРАКТИЧНУ РЕАЛІЗАЦІЮ	15 б.	15 б.	15 б.
Апаратно-програмний прототип системи за темою здобувача	5 б.	5 б.	5 б.
відповідність практичної реалізації постановці завдання	5 б.	5 б.	5 б.
наочність процесу демонстрації (+відео)	5 б.	5 б.	5 б.
ПІДСУМКОВА ОЦІНКА (захист роботи 25.06.26 о 10.00, Zoom)	100 б.	97 б.	97 б.
Кер. (П.І.Б. підпис)		(П.І.Б. підпис)	
(П.І.Б. підпис)		(П.І.Б. підпис)	
(П.І.Б. підпис)		(П.І.Б. підпис)	

Міністерство освіти і науки України
Український державний університет залізничного транспорту
Факультет інформаційно-керуючих систем та технологій
Кафедра інформаційних технологій

Затверджую
Завідувач кафедри ІТ,
д.т.н., проф., Каргін А.О.



18 травня 2026 р.

Завдання

На кваліфікаційну роботу студента 3-го курсу скороченої форми навчання, здобувача першого (бакалаврського) рівня вищої освіти	Довженка В'ячеслава Сергійовича (прізвище ім'я по батькові в родовому відмінку)
Тема кваліфікаційної роботи	Система керування рухом по азимуту колісного робота
Постановка завдання в стислій формі	Виконати проектування апаратно-програмної системи, як фрагменту IoT, виконати аналіз предметної області, сформулювати концептуальну, логічну та фізичну моделі системи та за певними сценаріями виконати моделювання системи як нечіткої когнітивної карти та нечіткої логічної системи.
Джерела, які рекомендує керівник курсової роботи	А. О. Каргін, Р.С. Кузьменко. Модель ймовірно-нечіткого актуатора в контурі м'якого управління автономною безлюдною системою (https://doi.org/10.30837/2522-9818.2025.1.061)

Календарний план роботи

Дата отримання версії звіту керівником	Етап виконання роботи	Виконання (зазначає керівник)
21.05.2026	Уточнення моделей предметної області за темою роботи (UML діаграми, FCM модель). Формування першої версії звіту (включно 3.2)	Виконано
28.05.2026	Моделювання системи як нечіткої системи в Fuzzy Logic Tools (MATLAB). Уточнення фізичної моделі системи. Формування другої версії звіту (включно 4.3)	Виконано
4.06.2026	Тестування моделі в MATLAB. Опис архітектури системи. Формування повної версії звіту, презентації (+відео), доповіді.	Виконано
11.06.2026	Формування остаточних версій звіту, презентації, доповіді та відправлення керівнику	Виконано

Дата видачі завдання 18.05.26
Керівник (ПІБ) професор Каргін А.О.

ДЕКЛАРАЦІЯ

Усвідомлюючи свою відповідальність за надання неправдивої інформації, стверджую, що подана пояснювальна записка до кваліфікаційної роботи на тему «Система керування рухом по азимуту колісного робота» написана мною особисто.

Ця робота подається до захисту вперше та не передавалася іншим особам; не порушує авторських та суміжних прав, що закріплені статтями 21-25 Закону України «Про авторське право та суміжні права»; дані та інформація, які використані в роботі, не отримані в недозволений спосіб.

Я усвідомлюю, що у разі порушення перелічених вимог моя робота може бути відхилена без права її захисту, або під час захисту робота буде оцінена як незадовільна.

Здобувач першого (бакалаврського) рівня вищої освіти,
студент 3-го курсу, скороченого терміну навчання
132- ТШІ-Д23 групи
Галузь знань 12 - Інформаційні технології
Спеціальність 126 - Інформаційні системи та технології
Освітньо-професійна програма - Технології штучного інтелекту

Довженко В'ячеслав Сергійович
(ПІБ)

Дата 12.06.2026

Підпис Dob

РЕФЕРАТ

Довженко В.С. Система керування рухом по азимуту колісного робота. - На правах рукопису.

Пояснювальна записка до кваліфікаційної роботи здобувача першого (бакалаврського) рівня вищої освіти: 50 сторінок, 14 рисунків, 8 таблиць, 32 джерела, 4 додатки.

Об'єктом дослідження є система керування рухом колісного робота по азимуту на основі датчика компаса HMC5883L та п'яти ультразвукових датчиків HC-SR04.

Метою роботи є проектування апаратно-програмної системи керування рухом колісного робота по азимуту з мінімальним відхиленням від курсу під час об'їзду перешкод за допомогою нечіткого керування.

Методами дослідження є аналіз наявних підходів до азимутального керування колісними роботами, моделювання нечіткої когнітивної карти в Mental Modeler та нечіткої системи керування в MATLAB.

В роботі визначено лінгвістичні змінні системи. Вхідними змінними є відхилення від азимуту та показники п'яти ультразвукових датчиків, розташованих спереду, ліворуч, праворуч, під кутом 45° спереду-ліворуч та 45° спереду-праворуч. Вихідними змінними є напрям руху та кут повороту.

Сформовано базу правил нечіткого керування для чотирьох сценаріїв руху: без перешкод, об'їзд перешкоди, рух біля кута та комбінований сценарій. Сформовано концептуальну, логічну та фізичну моделі системи у вигляді UML-діаграм. Апаратний прототип системи включає мікроконтролер Arduino Uno, датчик компаса HMC5883L, п'ять ультразвукових датчиків HC-SR04 та драйвер двигунів L298N.

АЗИМУТ, КОЛІСНИЙ РОБОТ, НЕЧІТКЕ КЕРУВАННЯ, ДАТЧИК КОМПАСУ, УЛЬТРАЗВУКОВИЙ ДАТЧИК, UML, FCM, ARDUINO, MENTAL MODELER, MATLAB

ABSTRACT

Viacheslav Dovzhenko. Azimuth motion control system of a wheeled robot. - Manuscript rights.

Explanatory note to the qualification work of a candidate for the first (bachelor's) level of higher education: 50 pages, 14 figures, 8 tables, 32 sources, 4 appendices.

The object of the study is an azimuth motion control system for a wheeled robot based on the HMC5883L compass sensor and five HC-SR04 ultrasonic distance sensors.

The purpose of the work is to design a hardware-software azimuth motion control system for a wheeled robot with minimum course deviation during obstacle avoidance using fuzzy control methods.

The research methods include analysis of existing approaches to azimuth control of wheeled robots, fuzzy cognitive map modelling in Mental Modeler, and fuzzy logic system simulation in MATLAB.

The system linguistic variables are defined. The input variables are azimuth deviation and readings from five ultrasonic sensors positioned at the front, left, right, front-left at 45° , and front-right at 45° . The output variables are motion direction and steering angle.

A fuzzy rule base is formed for four motion scenarios: obstacle-free motion, obstacle avoidance, motion near a corner, and a combined scenario. Conceptual, logical, and physical models of the system are presented as UML diagrams. The hardware prototype includes an Arduino Uno microcontroller, HMC5883L compass sensor, five HC-SR04 ultrasonic sensors, and L298N motor driver.

AZIMUTH, WHEELED ROBOT, FUZZY CONTROL, COMPASS SENSOR, ULTRASONIC SENSOR, UML, FCM, ARDUINO, MENTAL MODELER, MATLAB

ЗМІСТ

ВСТУП	8
1 ПОСТАНОВКА ЗАДАЧІ	9
2 ОПИС ПРЕДМЕТНОЇ ОБЛАСТІ І ОГЛЯД ПІДХОДІВ ДО ВИРІШЕННЯ ЗАВДАННЯ	10
2.1 Опис предметної області	10
2.2 Огляд підходів до вирішення завдання	12
3 ПРОЄКТУВАННЯ СИСТЕМИ КЕРУВАННЯ РУХОМ ПО АЗИМУТУ КОЛІСНОГО РОБОТА	17
3.1 Концептуальна модель системи	17
3.2 Логічна модель системи. Моделювання в Mental Modeler	18
3.3 Логічна модель системи. Моделювання в MATLAB	20
3.4 Фізична модель системи	27
4 АПАРАТНО-ПРОГРАМНИЙ ПРОТОТИП СИСТЕМИ	30
4.1 Топологія та компоненти апаратної частини прототипу системи	30
4.2 Архітектура та компоненти програмної частини прототипу системи	32
4.3 Опис даних та бібліотек підтримки компонентів системи	34
ВИСНОВКИ	37
СПИСОК ПОСИЛАНЬ	39
ДОДАТОК А ДІАГРАМИ UML	42
ДОДАТОК Б МОДЕЛЮВАННЯ В MENTAL MODELER	44
ДОДАТОК В МОДЕЛЮВАННЯ В MATLAB	48
ДОДАТОК Г МОДЕЛЮВАННЯ АПАРАТНОЇ ЧАСТИНИ ПРОТОТИПУ СИСТЕМИ	51

ВСТУП

У сучасних умовах розвитку робототехніки та систем штучного інтелекту актуальним є розроблення систем керування рухом автономних колісних роботів. Забезпечення точного орієнтування робота у просторі є ключовою задачею для реалізації функцій автономної навігації, патрулювання та виконання цільових маршрутів. Визначення сутності поняття азимуту та методів його вимірювання за допомогою сучасних датчиків компаса в системах вбудованих обчислень є важливими для задач мобільної робототехніки.

Розроблення системи керування рухом по азимуту передбачає розуміння обмеженості можливостей апаратно-програмних платформ на базі мікроконтролерів, зокрема Arduino, та особливостей інтеграції сенсорних пристроїв у реальних умовах експлуатації. Виділення основних концептів предметної області, таких як азимут, система керування, нечіткий регулятор та датчик компаса, і визначення зв'язків між ними є необхідною умовою коректного формування апаратно-програмної системи.

Моделювання системи засобами мови UML (SysML) є одним з ключових етапів розроблення та дозволяє обґрунтувати вибір компонентів системи, визначити її функції та структуру. Побудова нечіткої когнітивної карти (FCM) за допомогою пакета Mental Modeler дає змогу проаналізувати взаємозв'язки між компонентами системи та обґрунтувати правила нечіткого керування.

Моделювання правил нечіткого керування апаратною частиною прототипу системи у середовищі MatLab дозволяє верифікувати обрані методи керування та визначити обмеження системи ще до її фізичної реалізації. Результати моделювання складають концептуальну основу для подальшої розробки та вдосконалення системи керування рухом по азимуту колісного робота.

1 ПОСТАНОВКА ЗАДАЧІ

Метою даної роботи є розроблення апаратно-програмної системи керування рухом колісного робота по азимуту на основі датчика компаса. Система повинна забезпечувати визначення поточного кутового положення робота у горизонтальній площині та автоматичне коригування його курсу відповідно до заданого напрямку руху, а також подолання перешкод, розташованих як на шляху руху робота, так і безпосередньо перед ним. Для реалізації функцій керування передбачається використання мікроконтролерної платформи Arduino у поєднанні з датчиком компаса, ультразвуковими датчиками виявлення перешкод та виконавчими механізмами приводу коліс.

Для досягнення поставленої мети необхідно вирішити такі завдання:

1. Виконати аналіз існуючих підходів до розроблення систем керування рухом колісних роботів по азимуту, розглянути апаратно-програмні прототипи подібних систем та визначити їх переваги і недоліки.
2. Виконати аналіз предметної області, визначити основні концепти системи та зв'язки між ними, сформулювати опис функцій системи та правил нечіткого керування апаратною частиною прототипу.
3. Визначити алгоритм об'їзду перешкод під час руху за заданим азимутом, що забезпечує коректне відновлення курсу після їх подолання.
4. Сформулювати концептуальну модель апаратно-програмної системи у вигляді діаграми варіантів використання засобами мови UML (SysML).
5. Обґрунтувати компоненти системи за допомогою моделі нечіткої когнітивної карти з використанням пакета Mental Modeler.
6. Обґрунтувати правила нечіткого керування апаратною частиною прототипу системи шляхом моделювання у середовищі MatLab.
7. Розробити апаратно-програмний прототип системи.
8. Оформити пояснювальну записку до кваліфікаційної роботи відповідно до встановлених вимог.

2 ОПИС ПРЕДМЕТНОЇ ОБЛАСТІ І ОГЛЯД ПІДХОДІВ ДО ВИРІШЕННЯ ЗАВДАННЯ

2.1 Опис предметної області

Опис предметної області в роботі зроблено відповідно до вимог [1-2].

Автономні колісні роботи сьогодні знаходять застосування у патрулюванні периметрів, доставці вантажів, моніторингу відкритих територій та проведенні пошуково-рятувальних операцій там, де людина діяти не може. Навігація такої платформи передбачає самостійне переміщення від поточної позиції до заданої точки без участі оператора. Спільна вимога до всіх навігаційних задач одна: система повинна витримувати заданий напрям і самостійно обходити перешкоди на шляху.

Азимут є головним навігаційним параметром у цій роботі. Він являє собою кут між напрямком на північ та вектором руху об'єкта, що відраховується у градусах від 0 до 360 за годинниковою стрілкою. Для орієнтиру: 0 градусів відповідає півночі, 90 - сходу, 180 - півдню, 270 - заходу. Точне вимірювання поточного азимуту та його відхилення від цільового є необхідною умовою формування коригувальних сигналів керування [3].

Цільовий азимут руху до заданої географічної точки обчислюється за формулою (2.1):

$$\beta = \text{atan2}(\sin\Delta\lambda \cdot \cos\varphi_2, \cos\varphi_1 \cdot \sin\varphi_2 - \sin\varphi_1 \cdot \cos\varphi_2 \cdot \cos\Delta\lambda), \quad (2.1)$$

де φ_1, λ_1 - широта та довгота поточного положення робота у радіанах; φ_2, λ_2 - широта та довгота цільової точки у радіанах, а $\Delta\lambda = \lambda_2 - \lambda_1$.

Результат переводиться у градуси та нормалізується до діапазону від 0 до 360 за формулою (2.2):

$$\beta_{deg} = (\beta \cdot 180 / \pi + 360) \bmod 360, \quad (2.2)$$

Відхилення поточного азимуту від цільового визначається за формулою (2.3):

$$\Delta\beta = ((\beta_{target} - \beta_{current} + 180) \bmod 360) - 180, \quad (2.3)$$

де β_{target} - цільовий азимут, β_{current} - поточний азимут за показаннями компаса.

Результат нормалізується до діапазону від -180 до +180 градусів: від'ємні значення відповідають відхиленню ліворуч, додатні - праворуч.

Геометричний зміст наведених співвідношень для визначення азимута та відхилення від курсу ілюструє рисунок 2.1.

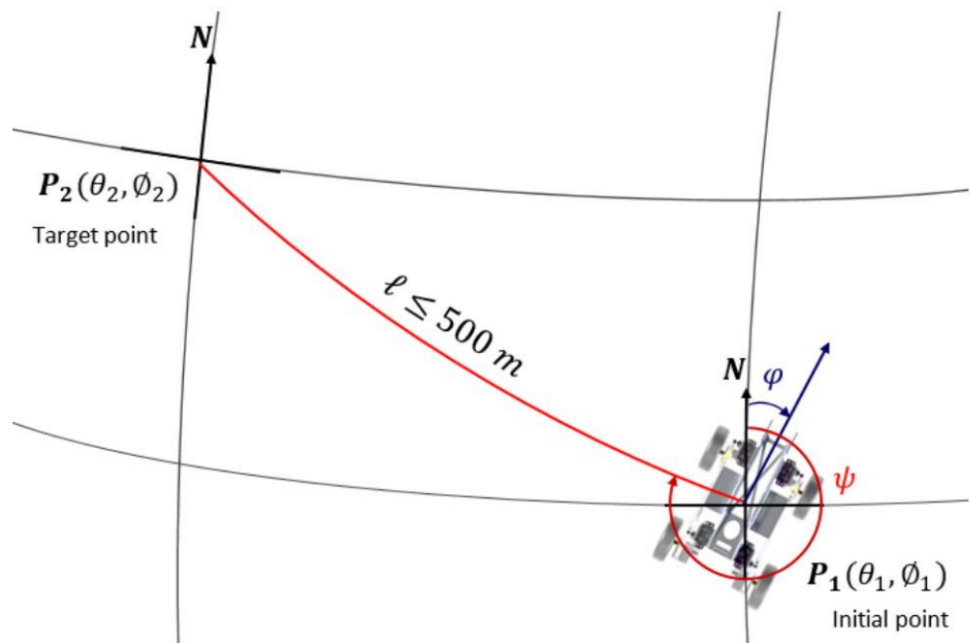


Рисунок 2.1 - Навігаційна геометрія колісного робота у географічній системі координат

Відстань між двома географічними точками обчислюється за формулою гаверсінуса (2.4):

$$d = 2R \cdot \arcsin(\sqrt{\sin^2((\phi_2 - \phi_1) / 2) + \cos\phi_1 \cdot \cos\phi_2 \cdot \sin^2((\lambda_2 - \lambda_1) / 2)}), \quad (2.4)$$

де R - радіус Землі (6371 км). Формула враховує сферичну форму Землі та забезпечує точність обчислення відстані, достатню для задач навігації мобільного робота.

Для отримання координат платформи використовуються супутникові системи позиціонування. Наразі діють чотири основні: американська GPS,

російська ГЛОНАСС, європейська Galileo та китайська BeiDou. Усі вони передають координати у вигляді широти (від -90 до +90 градусів) та довготи (від -180 до +180 градусів). У даній роботі система функціонує у відкритому середовищі зі стабільним сигналом GPS [4-5].

Застосовується GPS-модуль NEO-6M виробництва u-blox. Він підключається до мікроконтролера через UART зі швидкістю 9600 бод і оновлює дані з частотою 1-5 Гц. Точність позиціонування складає близько 2,5 метра за умови прийому сигналу від щонайменше чотирьох супутників. Час холодного старту - до 27 секунд. Дані передаються у форматі NMEA 0183: повідомлення GGA містить координати, висоту та кількість видимих супутників, RMC - координати й швидкість руху.

Поточний азимут вимірюється тривісним цифровим магнітометром HMC5883L виробництва Honeywell з інтерфейсом I2C та роздільною здатністю 2 мГ. Після калібрування і, за потреби, врахування магнітного схилення азимут обчислюється за показаннями осей X та Y за формулою (2.5):

$$\beta_{current} = \text{atan2}(Y, X) \cdot 180 / \pi. \quad (2.5)$$

Калібрування необхідне для компенсації впливу феромагнітних елементів конструкції. Після нього похибка вимірювання становить приблизно 1-2 градуси.

Виявлення перешкод забезпечується ультразвуковими датчиками HC-SR04. Вони вимірюють відстань у діапазоні 2-400 см на основі часу проходження імпульсу частотою 40 кГц. Відстань обчислюється за формулою (2.6):

$$d = t \cdot v_s / 20000, \quad (2.6)$$

де t - час проходження імпульсу у мікросекундах, v_s - швидкість звуку в повітрі.

Для температури 20 градусів Цельсія v_s становить 343 м/с. Похибка вимірювання не перевищує 3 міліметри, кут огляду становить не більше 15 градусів.

2.2 Огляд підходів до вирішення завдання

Керування рухом мобільного робота по азимуту з одночасним уникненням перешкод є комплексним завданням. Воно поєднує навігацію, локалізацію та реактивне керування в єдиний процес. Автономна навігація, по суті, відповідає на три запитання: де перебуває робот, куди він прямує і яким шляхом туди дістатися. У сучасних дослідженнях виокремлюють кілька усталених парадигм керування мобільними платформами.

Ієрархічна парадигма будується на циклі "сприйняття - планування - дія". Спочатку робот збирає дані від датчиків, потім формує модель навколишнього середовища, після чого планує крок і виконує його. Перевага цього підходу полягає в глобальному плануванні маршруту. Водночас він погано реагує на динамічні зміни: поява нової перешкоди змушує систему повністю перераховувати план.

Внутрішню структуру нечіткого регулятора, що застосовується у даній роботі, ілюструє рисунок 2.2.

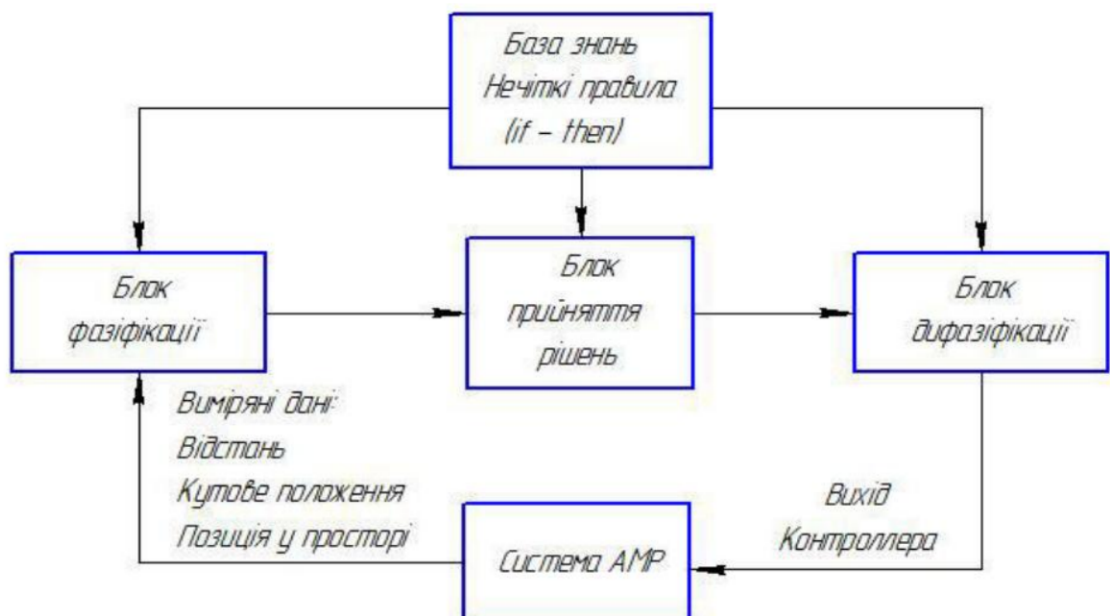


Рисунок 2.2 - Структура нечіткого контролера системи керування мобільним роботом

Реактивна парадигма з'явилася саме як відповідь на цей недолік. Вона безпосередньо пов'язує сигнал датчика з дією приводу, минаючи етап явного планування. Система складається з набору пар "сенсор - актуатор", сумарна реакція яких і визначає поведінку робота. Підхід швидкий і стійкий до локальних змін середовища, але не придатний там, де потрібен глобальний маршрут [6].

Окремий клас утворюють алгоритми типу Bug. Bug0 веде платформу прямо до цілі, а при виявленні перешкоди переключається на рух вздовж її контуру до відновлення вільного напрямку. Bug1 гарантує завершення об'їзду через повний обхід контуру і вибір найближчої точки до мети. TangentBug застосовує локальні вектори дотичних для скорочення траєкторії.

Спільна слабкість усіх трьох алгоритмів - суттєве відхилення від заданого курсу під час об'їзду. Для систем азимутального керування це критично, оскільки головна мета - досягти цільової точки з мінімальним відступом від заданого азимуту.

Для наочного уявлення про взаємодію компонентів навігаційної системи на рисунку 2.2 наведено узагальнену архітектуру, що відображає повний ланцюг обробки інформації - від зчитування сенсорних даних до формування керувальних команд для приводів.

З рисунка 2.2 видно, що навігаційний процес розпочинається зі збору сенсорних даних. Далі виконується їх обробка, визначення положення робота та формування моделі навколишнього простору. На підставі зібраної інформації система будує маршрут і генерує команди для виконавчих механізмів. Така архітектура дає платформі змогу самостійно рухатися у незнайомому або частково відомому середовищі.

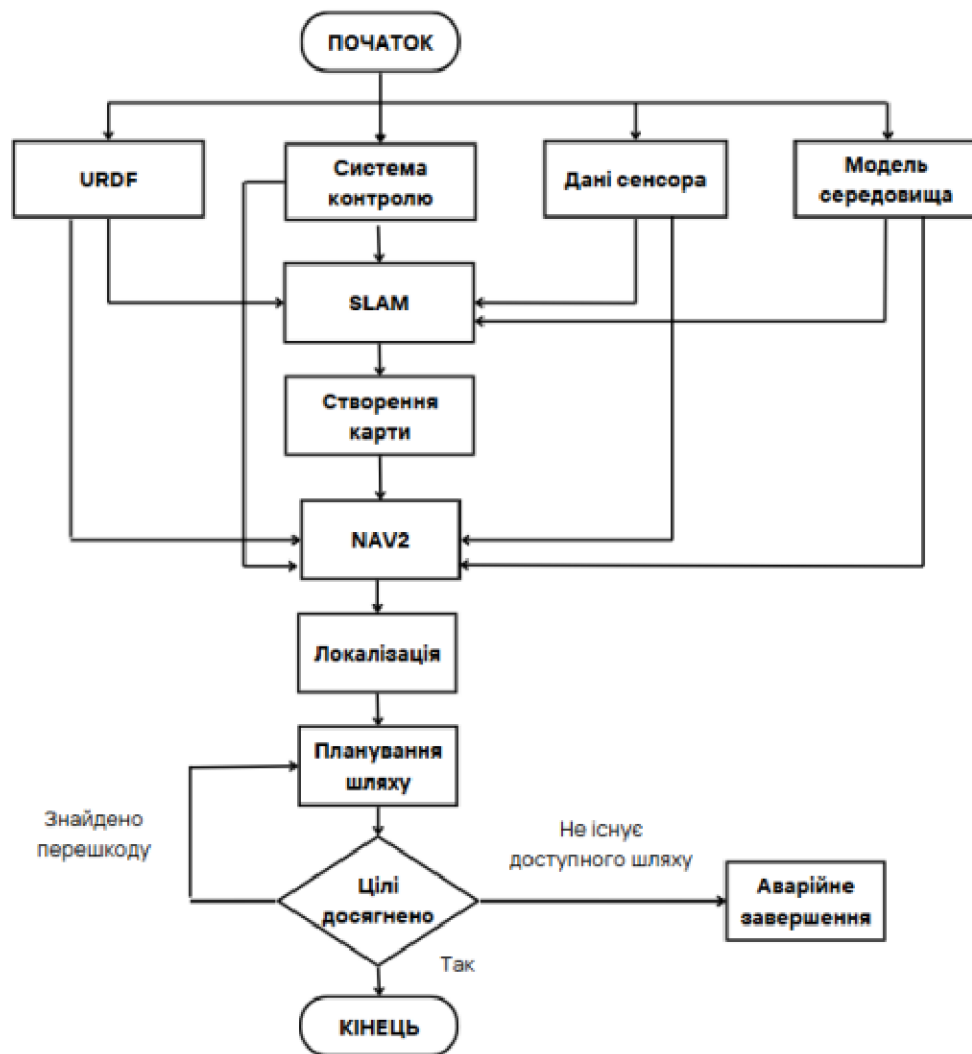


Рисунок 2.3 - Узагальнена архітектура навігаційної системи мобільного робота

Системи на основі нечіткої логіки належать до гібридного класу підходів [7]. Вони описують поведінку робота через набір лінгвістичних правил виду "якщо умова, то дія". Нечіткий регулятор не потребує точної математичної моделі об'єкта і добре витримує шуми вимірювань - а це особливо важливо при роботі з ультразвуковими датчиками. Саме тому нечітке керування успішно застосовується для корекції курсу мобільних роботів на основі відхилення від заданого азимута та даних про перешкоди.

Головна ідея системи керування полягає у тому, щоб робот рухався за заданим азимутом з мінімальним відхиленням навіть під час об'їзду перешкод. На відміну від алгоритмів типу Bug, що обходять контур перешкоди повністю,

нечіткий регулятор обирає напрямок об'їзду, який дає найменше відхилення від цільового азимута. Після подолання перешкоди система повертається до початкового курсу відповідно до поточного значення $\Delta\beta$.

Для охоплення всіх можливих ситуацій виділено чотири сценарії. Перший - рух без перешкод: усі датчики фіксують відстань вище граничного значення, система лише коригує відхилення від азимута. Другий - об'їзд перешкоди: D_{front} або D_{fl} , D_{fr} виявляють об'єкт у небезпечній зоні, регулятор обирає поворот на основі порівняння D_{left} , D_{right} , D_{fl} і D_{fr} . Третій - рух біля кута: одночасно спрацьовують датчики з різних боків, наприклад, D_{front} і D_{left} , система формує траєкторію відходу з урахуванням $\Delta\beta$. Четвертий - комбінований: перешкоди з кількох напрямків, пріоритет безпеки перевищує пріоритет мінімізації відхилення від азимута.

База правил нечіткого керування охоплює два класи дій: корекцію курсу та об'їзд перешкоди [8-9]. Правило корекції виглядає так: якщо $\Delta\beta$ є "позитивним великим" і перешкод немає, то поворот ліворуч є "великим". Правило об'їзду враховує комбінації датчиків: якщо D_{front} є "малим" і D_{right} є "великим", то поворот праворуч є "малим". Повний перелік правил наведено у підрозділі 3.3.

За підсумками аналізу для реалізації системи обрано метод нечіткого керування. Вибір обґрунтований трьома чинниками: відповідністю апаратним обмеженням платформи, стійкістю до шумів датчиків і можливістю формалізувати правила об'їзду перешкод без побудови точної моделі динаміки робота.

3 ПРОЄКТУВАННЯ СИСТЕМИ КЕРУВАННЯ РУХОМ ПО АЗИМУТУ КОЛІСНОГО РОБОТА

3.1 Концептуальна модель системи

Концептуальна модель системи керування рухом по азимуту колісного робота визначає основні функції, які система повинна виконувати, та описує взаємодію між користувачами (акторами) і самою апаратно-програмною системою. Головною метою розроблюваної системи є забезпечення точного переміщення колісного робота в заданому напрямку з мінімізацією відхилень від заданого курсу.

Для визначення вимог до системи та моделювання її функціональності розроблено діаграму варіантів використання мовою UML [10-11]. Вона дозволяє наочно представити прецеденти, які є важливими для користувачів системи. Основними акторами розроблюваної системи є оператор, який задає цільові координати й контролює виконання місії, та інженер, який відповідає за первинне налаштування і технічне обслуговування. Безпосереднє керування рухом здійснюється системою автономно, без участі оператора.

Призначення акторів системи наведено у таблиці 3.1.

Таблиця 3.1 - Актори системи

Ім'я актора системи (англ. мова)	Пояснення призначення (укр. мова)
Operator	Користувач, який задає цільовий азимут або координати точки призначення, ініціює автономний рух робота та здійснює моніторинг його поточного стану під час виконання завдання.
Engineer	Технічний спеціаліст, який виконує налаштування параметрів системи, калібрування датчиків орієнтації та діагностику апаратної частини.

Основні функції системи представлені у вигляді варіантів використання, що описують реакцію системи на запити акторів. Детальний опис прецедентів наведено у таблиці 3.2.

Таблиця 3.2 - Варіанти використання системи

Назва варіанта використання системи (англ. мова)	Пояснення призначення (укр. мова)
Set target azimuth	Встановлення бажаного кута напрямку руху (азимута) для колісного робота відносно початкової системи координат.
Start movement	Ініціалізація процесу руху колісного робота за встановленим азимутом після підтвердження параметрів оператором.
Stop movement	Примусова або штатна зупинка двигунів робота за командою оператора.
Monitor system status	Відстеження поточних параметрів робота (поточний азимут, кут відхилення, швидкість) у реальному часі.
Calibrate sensors	Процедура налаштування датчиків орієнтації (наприклад, магнітометра або гіроскопа) для забезпечення точності визначення курсу.

Зазначені актори та прецеденти пов'язані між собою відношеннями асоціації, які визначають, який саме актор ініціює той чи інший варіант використання. Назви акторів та прецедентів англійською мовою наведено на діаграмі варіантів використання, як і назву системи в межах діаграми. Діаграму варіантів використання, побудовану засобами Visual Paradigm, наведено як рисунок А.1 [12]. Запропонована концептуальна модель є основою для подальшої розробки логічної та фізичної моделей системи керування.

3.2 Логічна модель системи. Моделювання в Mental Modeler

Для побудови логічної моделі системи керування рухом по азимуту колісного робота було обрано метод нечітких когнітивних карт (Fuzzy Cognitive Map, FCM) [13]. Цей метод дає змогу формалізувати причинно-наслідкові зв'язки між основними параметрами системи та змодельовати її поведінку в умовах невизначеності навколишнього середовища. Розробку моделі виконано у спеціалізованому середовищі Mental Modeler [14] із послідовним визначенням концептів системи, встановленням зв'язків між ними, заданням ваг впливу для кожного зв'язку та проведенням сценарного аналізу типу "що-якщо".

Сенс моделювання за темою роботи полягає у створенні логіки прийняття рішень колісним роботом під час руху. Робот повинен одночасно виконувати два завдання: дотримуватися заданого курсу, тобто азимута, та уникати зіткнень з перешкодами. Модель FCM дає змогу наочно показати, як показники з різних сенсорів впливають на зміну швидкості лівого та правого двигунів, що забезпечує коригування траєкторії руху.

У процесі проектування було визначено набір концептів, які описують стан системи.

До вхідних концептів віднесено:

1. відхилення від азимута;
2. напрям азимута до точки;
3. перешкода спереду (показник центрального сенсора);
4. перешкода зліва (показник лівого бокового сенсора);
5. перешкода справа (показник правого бокового сенсора);
6. перешкода під кутом 45 градусів зліва;
7. перешкода під кутом 45 градусів справа.

До вихідних концептів, які формують керуючий вплив на апаратну частину, віднесено швидкість лівого двигуна та швидкість правого двигуна. Зміна співвідношення між цими двома вихідними змінними забезпечує повороти робота на потрібний кут.

Основні компоненти розробленої нечіткої когнітивної карти наведено у таблиці 3.3.

Таблиця 3.3 - Компоненти FCM

Назва компоненту (англ. мова)	Пояснення призначення (укр. мова)
Azimuth Deviation	Вхідна змінна, що відображає поточну кутову похибку між заданим та фактичним напрямком руху робота.
Target Azimuth Direction	Вхідна змінна, що задає глобальний напрям до кінцевої точки маршруту.
Obstacle Front	Вхідна змінна від центрального сенсора, що фіксує наявність перепони прямо по курсу руху.

Obstacle Left	Вхідна змінна від лівого бокового сенсора для виявлення перешкод зліва.
Obstacle Right	Вхідна змінна від правого бокового сенсора для виявлення перешкод справа.
Obstacle Front Left	Вхідна змінна від сенсора, спрямованого під кутом 45 градусів ліворуч.
Obstacle Front Right	Вхідна змінна від сенсора, спрямованого під кутом 45 градусів праворуч.
Left Motor Speed	Вихідна змінна, що визначає рівень керуючого сигналу для лівого приводу.
Right Motor Speed	Вихідна змінна, що визначає рівень керуючого сигналу для правого приводу.
Navigation Accuracy	Системна метрика оцінки, що характеризує загальну точність дотримання курсу роботом після обробки всіх сигналів.

Під час перевірки моделі в Mental Modeler було уточнено зв'язки між концептами, що відповідають за виявлення перешкод і формування впливів на двигуни. Зокрема, було враховано взаємний вплив передніх та бокових датчиків на швидкість лівого і правого двигунів, що підвищує чутливість системи до зміни ситуації перед роботом. У результаті модель стала краще відображати сценарії обходу перешкод із подальшим поверненням на цільовий азимут.

Результати моделювання логічної структури системи у вигляді скріншотів робочого простору середовища Mental Modeler під час виконання різних сценаріїв наведено в додатку Б (рис. Б.1, рис. Б.2 та рис. Б.3). Таблицю з оцінкою характеристик створеної FCM-моделі, яка відображає розраховані показники щільності та центральності вузлів, також розміщено в додатку Б (табл. Б.1).

3.3 Логічна модель системи. Моделювання в MATLAB

Для практичної реалізації логіки прийняття рішень та забезпечення точного керування рухом колісного робота розроблено нечітку систему керування (Fuzzy control system) у середовищі MATLAB [15-16]. На відміну від нечіткої когнітивної карти, яка відображає загальні причинно-наслідкові зв'язки між концептами, система нечіткої логіки в MATLAB дає змогу отримати конкретні сигнали для двигунів на основі даних від сенсорів.

На першому етапі проєктування системи нечіткого керування виконано визначення вхідних і вихідних змінних, а також їхніх терм-множин. Вхідними змінними є відхилення від цільового курсу, напрямок до цільової точки та дані з п'яти ультразвукових датчиків відстані, які контролюють простір навколо робота. Вихідними змінними є сигнали керування швидкістю лівого та правого двигунів.

Опис вхідних і вихідних лінгвістичних змінних, а також їхніх термів наведено в таблиці 3.4.

Таблиця 3.4 - Вхідні та вихідні змінні нечіткої логічної системи

Назва нечіткої вхідної змінної або терму (англ.мова)	Терми лінгвістичної змінної	Пояснення призначення українською мовою
Azimuth_Deviation	Left, Zero, Right	Кутове відхилення від заданого азимуту. Визначає похибку курсу: відхилення ліворуч, рух точно за курсом або відхилення праворуч.
Target_Direction	Left, Straight, Right	Напрямок до цільової точки маршруту. Вказує географічне положення цілі відносно робота.
Obstacle_Front	Close, Medium, Far	Центральний датчик відстані до перешкоди. Фіксує об'єкти безпосередньо за курсом руху робота.
Obstacle_Left	Close, Far	Лівий боковий датчик відстані. Визначає наявність перешкод або стін зліва від платформи.
Obstacle_Right	Close, Far	Правий боковий датчик відстані. Визначає наявність перешкод або стін справа від платформи.
Obstacle_FL	Close, Far	Кутовий датчик, спрямований спереду ліворуч під кутом 45 градусів. Попереджає про наближення до перешкод під час повороту ліворуч.
Obstacle_FR	Close, Far	Кутовий датчик, спрямований спереду праворуч під кутом 45 градусів. Попереджає про наближення до перешкод під час повороту праворуч.
Left_Motor_Speed	Stop, Slow, Normal	Швидкість лівого двигуна. Вихідний сигнал керування

		(рівень ШІМ) для лівого приводного колеса.
Right_Motor_Speed	Stop, Slow, Normal	Швидкість правого двигуна. Вихідний сигнал керування (рівень ШІМ) для правого приводного колеса.

Для побудови системи нечіткого виведення було обрано алгоритм Мамдані (Mamdani). Вибір цього алгоритму обґрунтований його інтуїтивністю, можливістю опису поведінки системи за допомогою лінгвістичних правил та зручністю налаштування функцій належності. Дефазифікацію виконано методом центру ваги (Centroid), що забезпечує плавну зміну швидкостей двигунів колісного робота під час корекції азимута та об'їзду перешкод [17].

Взаємодія між вхідними параметрами та вихідними сигналами керування описується базою нечітких правил. Правила систематизовано за чотири ситуаційними сценаріями системи: нормальний рух за азимутом без перешкод, уникнення фронтальних перешкод, рух навколо кутових перешкод та комбінована критична ситуація. Базу знань сформовано з 22 логічних правил.

База правил нечіткого керування азимутальним рухом:

Сценарій 1. Нормальний рух за азимутом (відсутність перешкод)

1. ЯКІЩО Azimuth_Deviation є Zero ТА Target_Direction є Straight ТА Obstacle_Front є Far, ТО Left_Motor_Speed є Normal, Right_Motor_Speed є Normal.
2. ЯКІЩО Azimuth_Deviation є Left ТА Obstacle_Front є Far, ТО Left_Motor_Speed є Normal, Right_Motor_Speed є Slow.
3. ЯКІЩО Azimuth_Deviation є Right ТА Obstacle_Front є Far, ТО Left_Motor_Speed є Slow, Right_Motor_Speed є Normal.
4. ЯКІЩО Azimuth_Deviation є Zero ТА Target_Direction є Left ТА Obstacle_Front є Far, ТО Left_Motor_Speed є Slow, Right_Motor_Speed є Normal.
5. ЯКІЩО Azimuth_Deviation є Zero ТА Target_Direction є Right ТА Obstacle_Front є Far, ТО Left_Motor_Speed є Normal, Right_Motor_Speed є Slow.
6. ЯКІЩО Azimuth_Deviation є Left ТА Target_Direction є Left ТА Obstacle_Front є Far, ТО Left_Motor_Speed є Normal, Right_Motor_Speed є Stop.

Сценарій 2. Уникнення фронтальних перешкод

7. ЯКЩО Obstacle_Front є Medium ТА Obstacle_Left є Far ТА Obstacle_Right є Far, TO Left_Motor_Speed є Slow, Right_Motor_Speed є Slow.

8. ЯКЩО Obstacle_Front є Medium ТА Obstacle_Left є Close, TO Left_Motor_Speed є Normal, Right_Motor_Speed є Slow.

9. ЯКЩО Obstacle_Front є Medium ТА Obstacle_Right є Close, TO Left_Motor_Speed є Slow, Right_Motor_Speed є Normal.

10. ЯКЩО Obstacle_Front є Close ТА Obstacle_Left є Close, TO Left_Motor_Speed є Normal, Right_Motor_Speed є Stop.

11. ЯКЩО Obstacle_Front є Close ТА Obstacle_Right є Close, TO Left_Motor_Speed є Stop, Right_Motor_Speed є Normal.

12. ЯКЩО Obstacle_Front є Close ТА Obstacle_Left є Far ТА Obstacle_Right є Far, TO Left_Motor_Speed є Slow, Right_Motor_Speed є Stop.

Сценарій 3. Рух навколо кутових перешкод (реагування на кутові датчики)

13. ЯКЩО Obstacle_FL є Close ТА Obstacle_Front є Far, TO Left_Motor_Speed є Normal, Right_Motor_Speed є Slow.

14. ЯКЩО Obstacle_FR є Close ТА Obstacle_Front є Far, TO Left_Motor_Speed є Slow, Right_Motor_Speed є Normal.

15. ЯКЩО Obstacle_FL є Close ТА Obstacle_Front є Medium, TO Left_Motor_Speed є Normal, Right_Motor_Speed є Slow.

16. ЯКЩО Obstacle_FR є Close ТА Obstacle_Front є Medium, TO Left_Motor_Speed є Slow, Right_Motor_Speed є Normal.

17. ЯКЩО Obstacle_Front є Close ТА Obstacle_Left є Close ТА Obstacle_FL є Close, TO Left_Motor_Speed є Normal, Right_Motor_Speed є Stop.

18. ЯКЩО Obstacle_Front є Close ТА Obstacle_Right є Close ТА Obstacle_FR є Close, TO Left_Motor_Speed є Stop, Right_Motor_Speed є Normal.

Сценарій 4. Комбінована критична ситуація

19. ЯКЩО Obstacle_Front є Close ТА Obstacle_Left є Close ТА Obstacle_Right є Close, TO Left_Motor_Speed є Stop, Right_Motor_Speed є Stop.

20. ЯКЩО $Obstacle_FL \in Close$ $TA\ Obstacle_FR \in Close$ $TA\ Obstacle_Front \in Close$, $TO\ Left_Motor_Speed \in Stop$, $Right_Motor_Speed \in Stop$.

21. ЯКЩО $Obstacle_Front \in Close$ $TA\ Obstacle_FL \in Close$ $TA\ Obstacle_Right \in Far$, $TO\ Left_Motor_Speed \in Normal$, $Right_Motor_Speed \in Stop$.

22. ЯКЩО $Obstacle_Front \in Close$ $TA\ Obstacle_FR \in Close$ $TA\ Obstacle_Left \in Far$, $TO\ Left_Motor_Speed \in Stop$, $Right_Motor_Speed \in Normal$.

Узагальнені маршрути руху колісного робота за чотирма ситуаційними сценаріями наведено на рисунку 3.1.

МАРШРУТИ РОБОТА ДО ЦІЛЬОВОЇ ТОЧКИ ЗА ЧОТИРМА СЦЕНАРІЯМИ

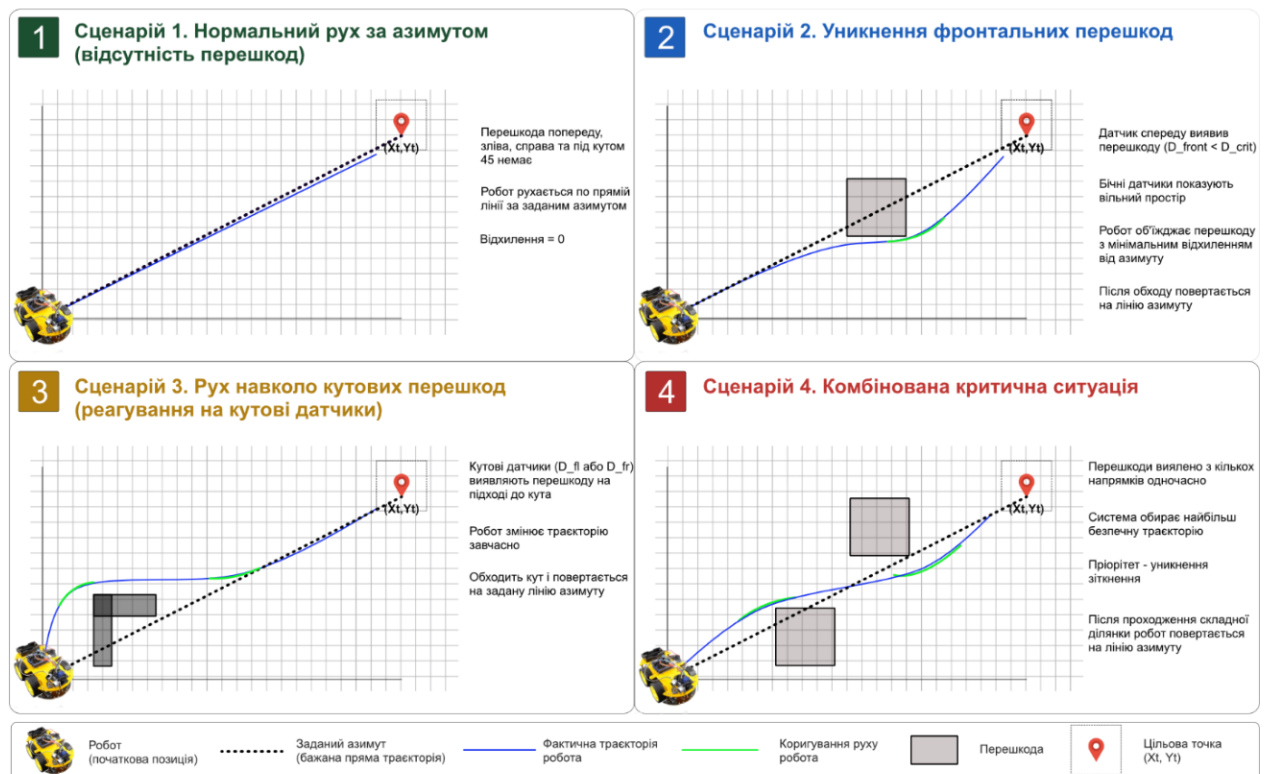


Рисунок 3.1 - Маршрути руху колісного робота до цільової точки за чотирма ситуаційними сценаріями.

Аналіз спрацьовування бази нечітких правил для сценарію 4:

Сценарій 4, тобто критична комбінована ситуація, відповідає траєкторії, позначеній на рисунку 3.1, де колісний робот потрапляє в умови одночасного

блокування курсу з кількох напрямків. У межах цього сценарію послідовно активуються такі правила бази знань:

1. Етап наближення до пастки. Робот рухається у напрямку цільової точки, проте виявляє перед собою тупиковий контур, у якому одночасно спрацьовують фронтальний датчик, а також обидва кутові датчики під 45 градусів. Система активує правило 20, що ініціює повну зупинку обох двигунів для запобігання зіткненню та подальшого перерахунку маневру.
2. Етап локалізації вільного коридору. Після зупинки система оцінює бічний простір. Якщо лівий і фронтальний напрями заблоковані, а з правого боку виявлено вільну зону, активується правило 22 або правило 21 залежно від конфігурації перешкод. Це дає змогу виконати маневр виходу з небезпечної зони з пріоритетом безпеки над точністю дотримання азимута.
3. Етап виходу на безпечну траєкторію та відновлення курсу. Після подолання небезпечної ділянки робот переходить до правил першого сценарію, які забезпечують плавне вирівнювання платформи на заданий курс і повернення до руху за цільовим азимутом.

Для верифікації розробленої бази знань та аналізу динаміки зміни сигналів проведено серію симуляційних експериментів у середовищі MATLAB Fuzzy Logic Toolbox . Вхідні вектори моделювали зміну просторового стану робота під час виконання маневрів. Протокол тестування логічної моделі системи за ключовими точками чотирьох сценаріїв представлено в таблиці 3.6.

Таблиця 3.6 - Протокол тестування системи нечіткого керування

№ експерименту	Azimuth_Deviation, град.	Target_Direction	Obstacle_Front, см	Obstacle_FL, см	Obstacle_FR, см	Left_Motor_Speed, ШИМ	Right_Motor_Speed, ШИМ	Активоване правило БЗ
1	0	Straight	400 (Far)	400 (Far)	400 (Far)	255 (Normal)	255 (Normal)	Правило 1
2	-35	Straight	400 (Far)	400 (Far)	400 (Far)	255 (Normal)	120 (Slow)	Правило 2
3	+40	Straight	400 (Far)	400 (Far)	400 (Far)	120 (Slow)	255 (Normal)	Правило 3

4	0	Straight	150 (Medium)	400 (Far)	400 (Far)	120 (Slow)	120 (Slow)	Правило 7
5	-15	Straight	120 (Medium)	30 (Close)	400 (Far)	120 (Slow)	255 (Normal)	Правило 8
6	0	Straight	20 (Close)	400 (Far)	400 (Far)	120 (Slow)	0 (Stop)	Правило 12
7	+10	Straight	350 (Far)	25 (Close)	400 (Far)	255 (Normal)	120 (Slow)	Правило 13
8	0	Straight	110 (Medium)	400 (Far)	35 (Close)	120 (Slow)	255 (Normal)	Правило 16
9	-45	Left	15 (Close)	20 (Close)	22 (Close)	0 (Stop)	0 (Stop)	Правило 20
10	+85	Right	18 (Close)	15 (Close)	400 (Far)	255 (Normal)	0 (Stop)	Правило 21

Протокол тестування охоплює всі чотири ситуаційні сценарії в порядку зростання складності.

Експерименти 1-3 відповідають Сценарію 1 (вільний простір). Значення всіх датчиків відстані дорівнюють 400 см (терм Far), тому система реагує лише на Azimuth_Deviation. За нульового відхилення (експ. 1) обидва двигуни отримують ШІМ 255 (Normal) згідно з правилом 1. Відхилення ліворуч на -35° (експ. 2) активує правило 2: правий двигун уповільнюється до 120 (Slow), платформа довертає ліворуч. Відхилення праворуч на $+40^{\circ}$ (експ. 3) активує правило 3 із дзеркальним ефектом.

Експерименти 4-6 моделюють Сценарій 2 (фронтальні перешкоди). За фронтальної перешкоди 150 см (Medium) без бічних перешкод (експ. 4) правило 7 симетрично уповільнює обидва двигуни. Поява близької перешкоди зліва 30 см (експ. 5) активує правило 8: правий двигун прискорюється до Normal - робот уникає перешкоди праворуч. За критично близької фронтальної перешкоди 20 см (експ. 6) правило 12 зупиняє правий двигун - платформа розвертається ліворуч.

Експерименти 7-8 ілюструють Сценарій 3 (кутові перешкоди). У експерименті 7 фронтальний простір вільний (350 см, Far), але Obstacle_FL

фіксує 25 см (Close). Правило 13 уповільнює правий двигун - об'їзд праворуч. У експерименті 8 Obstacle_FR = 35 см (Close) за фронтальної перешкоди 110 см (Medium) активує правило 16 - об'їзд ліворуч.

Експерименти 9-10 відтворюють Сценарій 4 (критична комбінація). У експерименті 9 всі три напрямки заблоковано (15-22 см, Close) - правило 20 зупиняє обидва двигуни (ШИМ = 0). У експерименті 10 фронтальний і лівий кутовий датчики фіксують перешкоди, а правий кутовий - вільний (400 см). Правило 21 вмикає лівий двигун на Normal, правий зупиняє - вихід у вільний коридор.

У всіх десяти тестових векторах система формує коректні сигнали без конфліктів правил. Значення ШИМ змінюються плавно (0, 120, 255) завдяки дефазифікації методом центру ваги.

Аналіз результатів тестування свідчить про адаптивність логічної моделі та її здатність формувати керувальні сигнали залежно від поточного стану навколишнього середовища. Отримані результати моделювання, а також графічні поверхні відгуку системи наведено в додатку В.

3.4 Фізична модель системи

Фізична модель системи керування рухом по азимуту колісного робота описує розміщення програмних компонентів на апаратних вузлах та фізичні зв'язки між компонентами системи. Для подання фізичної моделі використано діаграму розгортання мови UML [19], наведену в додатку А (рис. А.3). Діаграма розгортання дає змогу наочно відобразити топологію апаратного забезпечення, фізичні вузли системи, канали зв'язку між ними та програмні компоненти, розгорнуті на відповідних вузлах.

Центральним обчислювальним вузлом системи є мікроконтролер Arduino Uno на основі ATmega328P з тактовою частотою 16 МГц [20]. Він забезпечує збір і обробку даних від сенсорів, виконання алгоритму нечіткого керування та формування керувальних сигналів для виконавчих механізмів. До

мікроконтролера підключено такі компоненти: тривісний цифровий магнітометр HMC5883L через двопровідний послідовний інтерфейс I²C, GPS-модуль NEO-6M через послідовний інтерфейс UART зі швидкістю обміну 9600 бод, п'ять ультразвукових датчиків відстані HC-SR04 через цифрові виводи загального призначення [21], а також драйвер двигунів L298N через виходи широтно-імпульсної модуляції та цифрові виводи керування напрямком обертання. Драйвер L298N, у свою чергу, керує двома двигунами постійного струму, що забезпечують привід лівого та правого коліс платформи.

Живлення системи здійснюється від зовнішнього акумулятора напругою 7,4-12 В. Мікроконтролер та логічна частина драйвера L298N отримують напругу 5 В через вбудований лінійний стабілізатор плати Arduino Uno. Силова частина драйвера L298N живиться безпосередньо від акумулятора, що дає змогу забезпечити необхідний рівень струму для приводу двигунів без перевантаження стабілізатора мікроконтролера. Датчики HMC5883L та HC-SR04 живляться від виводів 3,3 В та 5 В відповідно.

Таблиця 3.5 - Компоненти апаратного забезпечення

Назва компоненту (англ. мова)	Пояснення призначення (укр. мова)
Arduino Uno	Мікроконтролерна плата на основі ATmega328P з тактовою частотою 16 МГц; виконує функції центрального обчислювального вузла: збирає та обробляє дані від усіх сенсорів, виконує алгоритм нечіткого керування, формує керуючі сигнали для виконавчих механізмів.
HMC5883L Compass Sensor	Тривісний цифровий магнітометр виробництва Honeywell; підключається за протоколом I ² C; вимірює складові вектора магнітного поля Землі та забезпечує визначення поточного азимуту робота з похибкою не більше 1-2 градусів після калібрування.
HC-SR04 Ultrasonic Sensor (x5)	Ультразвуковий датчик відстані; вимірює відстань до перешкод у діапазоні від 2 до 400 сантиметрів на основі часу розповсюдження ультразвукового імпульсу частотою 40 кГц; п'ять датчиків розміщено у напрямках: прямо, ліворуч, праворуч, під кутом 45 градусів спереду-ліворуч та 45 градусів спереду-праворуч, що забезпечує огляд сектору 180 градусів перед роботом.
L298N Motor Driver	Двоканальний драйвер двигунів постійного струму; приймає сигнали ШІМ від мікроконтролера та забезпечує

	необхідний рівень струму для незалежного керування швидкістю і напрямком обертання лівого та правого приводних двигунів.
Neo-6M GPS Module	GPS-модуль виробництва u-blox; підключається за інтерфейсом UART; передає поточні координати у форматі протоколу NMEA 0183 з частотою оновлення до 5 Гц та точністю позиціонування 2,5 метри за наявності сигналу від не менше чотирьох супутників.
DC Motor (x2)	Двигун постійного струму; слугує виконавчим механізмом для приводу ведучих коліс платформи; швидкість та напрямок обертання кожного двигуна незалежно задаються через драйвер L298N сигналами ШІМ від мікроконтролера.
Wheeled Robot Chassis	Механічна платформа колісного робота; є основою для монтажу всіх апаратних компонентів системи; включає два ведучі колеса з двигунами та одне пасивне колесо для забезпечення стійкості під час руху.
Li-Ion Battery Pack	Акумуляторна батарея літій-іонного типу; забезпечує автономне живлення всіх апаратних компонентів системи; номінальна напруга становить 7,4-12 В.

Після аналізу фізичної моделі можна зробити висновок, що апаратна структура системи є компактною, технологічно простою та придатною для реалізації на базі мікроконтролерної платформи Arduino Uno. Сукупність сенсорів забезпечує визначення напрямку руху та виявлення перешкод, а драйвер L298N і двигуни постійного струму дають змогу реалізувати керування траєкторією руху відповідно до обраної логіки нечіткого виведення.

4 АПАРАТНО-ПРОГРАМНИЙ ПРОТОТИП СИСТЕМИ

4.1 Топологія та компоненти апаратної частини прототипу системи

Апаратна частина прототипу системи керування рухом по азимуту колісного робота побудована за централізованою топологією, у якій єдиним обчислювальним вузлом є мікроконтролерна плата Arduino Uno. Загальну структуру системи, що відображає склад апаратних вузлів та фізичні зв'язки між ними, наведено на рисунку 4.1. Детальний склад апаратних компонентів описано у таблиці 3.5.

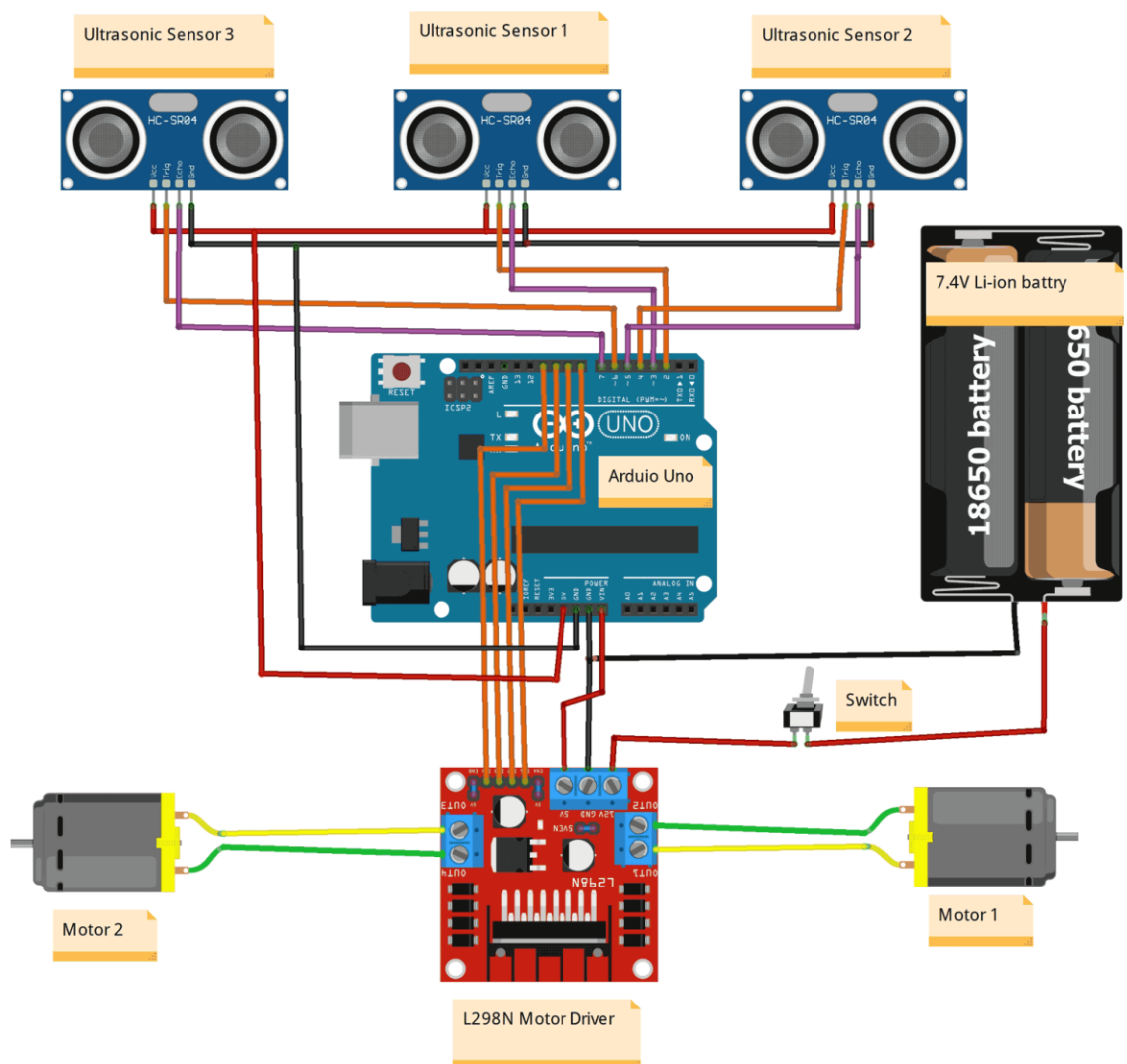


Рисунок 4.1 - Топологія апаратної частини прототипу системи

Датчик компаса HMC5883L підключається до мікроконтролера за двопровідним послідовним інтерфейсом I²C: вивід SDA датчика з'єднано з аналоговим виводом A4 плати Arduino Uno, що виконує функцію лінії даних, а вивід SCL - з аналоговим виводом A5, що виконує функцію лінії тактування. Живлення датчика здійснюється напругою 3,3 В від відповідного виводу плати. Для коректного функціонування шини I²C на лініях SDA та SCL встановлюються підтягувальні резистори номіналом 4,7 кОм.

GPS-модуль NEO-6M підключається до мікроконтролера через програмний послідовний інтерфейс SoftwareSerial: вивід TX модуля з'єднано з цифровим виводом D4 плати Arduino Uno, а вивід RX - з цифровим виводом D3. Живлення модуля здійснюється напругою 3,3 В від відповідного виводу плати.

П'ять ультразвукових датчиків відстані HC-SR04 підключаються до цифрових виводів мікроконтролера: кожен датчик має вивід Trig для подачі імпульсу запуску та вивід Echo для приймання відповідного відлуння. Виводи Trig та Echo датчиків підключаються до цифрових виводів D2, D5, D6, D7, D8, D9, D10, D11, D12, D13 мікроконтролера відповідно до фізичного розташування датчиків на корпусі робота. Датчики живляться напругою 5 В від відповідного виводу плати Arduino Uno.

Драйвер двигунів L298N підключається до мікроконтролера через чотири виводи: ENA та ENB для керування швидкістю лівого та правого двигунів відповідно за допомогою сигналів ШІМ, а IN1, IN2, IN3, IN4 - для визначення напрямку обертання кожного двигуна. Силові виводи OUT1 та OUT2 драйвера підключено до лівого двигуна постійного струму, а OUT3 та OUT4 - до правого. Живлення логічної частини драйвера L298N здійснюється напругою 5 В від мікроконтролера, а силовій частини - безпосередньо від акумулятора.

Датчики HC-SR04 розміщено на передній частині корпусу робота у п'яти фіксованих напрямках: один датчик спрямовано прямо по курсу, два датчики - під кутом 90 градусів ліворуч та праворуч від осі руху, ще два - під кутом 45 градусів спереду-ліворуч та спереду-праворуч. Таке розташування формує зону спостереження шириною 180 градусів перед роботом, що є достатнім для

реактивного уникнення перешкод під час руху по азимуту. Датчик компаса HMC5883L та GPS-модуль NEO-6M встановлено на достатній відстані від джерел електромагнітних завад, зокрема від двигунів та драйвера, для мінімізації похибок вимірювань. Схему підключення компонентів апаратного прототипу до мікроконтролера, виконану у середовищі Fritzing, наведено в додатку Г.

4.2 Архітектура та компоненти програмної частини прототипу системи

Програмна частина прототипу системи керування рухом по азимуту колісного робота розгорнута на єдиному обчислювальному вузлі - мікроконтролері Arduino Uno. Програму реалізовано у вигляді одного скетчу мовою C/C++ у середовищі Arduino IDE, який працює в циклічному режимі без використання операційної системи.

Архітектура програмної частини є реактивною та циклічною. Кожна ітерація головного циклу `loop()` містить послідовне виконання чотирьох функціональних блоків: зчитування даних від сенсорів, обчислення відхилення від заданого азимута, виконання нечіткого виводу та формування сигналів для виконавчих механізмів. Тривалість одного циклу визначається переважно часом вимірювання ультразвукових датчиків HC-SR04 і не перевищує 100 мілісекунд, що забезпечує частоту оновлення сигналів не менше 10 Гц.

Функція `setup()` виконується одноразово під час запуску мікроконтролера та забезпечує налаштування інтерфейсів I²C і UART, ініціалізацію виводів GPIO для підключення ультразвукових датчиків та виводів ШІМ для керування драйвером двигунів, а також виконання процедури калібрування магнітометра HMC5883L. Склад програмних компонентів системи наведено у таблиці 4.1.

Таблиця 4.1 - Компоненти програмної частини прототипу системи

Назва компоненту (англ.мова)	Обчислювальний вузол (англ.мова)	Пояснення призначення (укр.мова)
Sensor Reading Module	Arduino Uno	Модуль зчитування даних від п'яти ультразвукових датчиків HC-SR04,

		магнітометра HMC5883L та GPS-модуля Neo-6M; реалізований у вигляді функцій <code>readUltrasonic()</code> , <code>readCompass()</code> та <code>readGPS()</code> .
Azimuth Processing Module	Arduino Uno	Модуль обчислення поточного азимуту за показаннями магнітометра за формулою (2.5), цільового азимуту за координатами GPS за формулою (2.1) та відхилення $\Delta\beta$ за формулою (2.3); реалізований у вигляді функцій <code>computeAzimuth()</code> та <code>computeDeviation()</code> .
Fuzzy Inference Module	Arduino Uno	Модуль нечіткого виводу, що реалізує фазифікацію входних змінних, обчислення ступенів активації правил нечіткої бази знань та дефазифікацію вихідної змінної методом центроїда; реалізований у вигляді функції <code>fuzzyControl()</code> .
Motor Control Module	Arduino Uno	Модуль формування керуючих сигналів ШІМ для драйвера двигунів L298N та цифрових сигналів керування напрямком обертання на основі результату нечіткого виводу; реалізований у вигляді функції <code>setMotors()</code> .

Взаємодія програмних компонентів з апаратними здійснюється через бібліотеки підтримки периферійних пристроїв. Бібліотека `Wire.h` забезпечує обмін даними з магнітометром HMC5883L за протоколом I2C. Бібліотека `SoftwareSerial.h` реалізує програмний UART-інтерфейс для обміну з GPS-модулем NEO-6M на виводах D3 та D4. Бібліотека `TinyGPS++` використовується для розбору повідомлень протоколу NMEA 0183, що надходять від GPS-модуля, та отримання числових значень координат широти і довготи. Вимірювання відстані ультразвуковими датчиками HC-SR04 реалізовано безпосередньо засобами мови C++ через функцію `pulseIn()`, що вимірює тривалість імпульсу Echo у мікросекундах без застосування зовнішніх бібліотек.

4.3 Опис даних та бібліотек підтримки компонентів системи

Прототип системи керування рухом по азимуту колісного робота обробляє два типи даних: дані, що надходять від апаратних компонентів у реальному часі, та проміжні обчислені величини, які використовуються алгоритмом нечіткого керування для формування вихідних сигналів. Опис змінних, що обробляються програмною частиною прототипу, наведено у таблиці 4.2.

Таблиця 4.2 - Опис даних програмної частини прототипу системи

Назва змінної	Тип даних (C++)	Діапазон значень	Джерело	Призначення
magX, magY, magZ	int16_t	-4096 ... +4096	HMC5883L (I2C)	Сирі значення проєкцій вектора магнітного поля Землі на осі X, Y, Z у одиницях АЦП магнітометра.
azimuthCurrent	float	0,0 ... 360,0	Обчислюється з magX, magY	Поточний азимут робота у градусах, обчислений за формулою (2.5) з урахуванням поправки деклінації.
latitude, longitude	double	-90,0 ... +90,0 та -180,0 ... +180,0	Neo-6M (UART)	Поточні географічні координати робота у градусах, отримані з повідомлень NMEA 0183 типу GGA та RMC.
azimuthTarget	float	0,0 ... 360,0	Обчислюється з latitude, longitude	Цільовий азимут до кінцевої точки маршруту у градусах, обчислений за формулою (2.1).
azimuthDeviation	float	-180,0 ... +180,0	Обчислюється з azimuthTarget, azimuthCurrent	Відхилення $\Delta\beta$ поточного азимуту від цільового у градусах відповідно до формули (2.3); від'ємні значення відповідають відхиленню ліворуч, додатні - праворуч.
distFront, distLeft,	float	2,0 ... 400,0	HC-SR04 x5 (GPIO)	Відстані до перешкод у сантиметрах у п'яти

distRight, distFL, distFR				напрямах спостереження, обчислені за формулою (2.6) з тривалості імпульсу Echo.
fuzzyOutput	float	-1,0 ... +1,0	Модуль нечіткого виводу	Нормалізований вихідний сигнал нечіткого регулятора: від'ємні значення відповідають повороту ліворуч, додатні - праворуч, нульове значення відповідає прямолінійному руху.
pwmLeft, pwmRight	int	0 ... 255	Обчислюється з fuzzyOutput	Значення ШІМ-сигналу для лівого та правого каналів драйвера двигунів L298N, що визначають швидкість обертання відповідних приводних двигунів.
satCount	uint8_t	0 ... 12	Neo-6M (UART)	Кількість видимих супутників GPS; використовується для перевірки достовірності координат перед обчисленням цільового азимуту.

Для забезпечення роботи апаратних компонентів системи у програмі використовуються бібліотеки підтримки, перелік і призначення яких наведено у таблиці 4.3.

Таблиця 4.3 - Бібліотеки підтримки компонентів системи

Назва бібліотеки	Тип	Версія	Призначення
Wire.h	Стандартна бібліотека Arduino IDE	вбудована	Реалізує протокол I2C на рівні апаратного інтерфейсу TWI мікроконтролера ATmega328P; забезпечує обмін даними з магнітометром HMC5883L: запис адреси регістра та зчитування шести байт сирих вимірювань осей X, Y, Z.

SoftwareSerial.h	Стандартна бібліотека Arduino IDE	вбудована	Реалізує програмний послідовний інтерфейс UART на будь-яких цифрових виводах мікроконтролера; використовується для прийому потоку символів від GPS-модуля Neo-6M на виводах D3 та D4 зі швидкістю 9600 бод.
TinyGPS++.h	Стороння бібліотека (Mikal Hart)	1.0.3	Виконує синтаксичний аналіз повідомлень протоколу NMEA 0183, що надходять від GPS-модуля, та надає числові значення широти, довготи, кількості супутників та ознаки достовірності фіксації позиції.

Алгоритм нечіткого виводу реалізовано безпосередньо у програмному коді без застосування зовнішніх бібліотек нечіткої логіки. Таке рішення зумовлене обмеженим обсягом оперативної пам'яті мікроконтролера ATmega328P, а також необхідністю зменшити накладні витрати на зберігання функцій належності та правил нечіткого виводу. Параметри функцій належності зберігаються у флеш-пам'яті мікроконтролера за допомогою директиви PROGMEM, що дає змогу не перевантажувати оперативну пам'ять.

ВИСНОВКИ

Метою даної кваліфікаційної роботи було розроблення апаратно-програмної системи керування рухом колісного робота по азимуту на основі датчика компасу HMC5883L та п'яти ультразвукових датчиків HC-SR04 з реалізацією нечіткого регулятора на мікроконтролерній платформі Arduino Uno.

У ході виконання роботи отримано такі результати.

Виконано аналіз предметної області та огляд існуючих підходів до азимутального керування колісними роботами. Встановлено, що використання нечіткого керування є доцільним для реалізації на мікроконтролерних платформах класу Arduino, оскільки такий підхід не потребує побудови повної математичної моделі динаміки робота, є стійким до похибок сенсорів і дає змогу формалізувати правила об'їзду перешкод.

Сформовано концептуальну модель системи у вигляді діаграми варіантів використання, логічну модель у вигляді нечіткої когнітивної карти в середовищі Mental Modeler та нечіткої логічної системи в середовищі MATLAB, а також фізичну модель у вигляді діаграми розгортання. Для основних сценаріїв руху сформовано базу правил нечіткого керування, що охоплює рух без перешкод, об'їзд перешкоди, рух біля кута та комбінований сценарій.

Розроблено апаратно-програмний прототип системи, що включає мікроконтролер Arduino Uno, датчик компасу HMC5883L, п'ять ультразвукових датчиків HC-SR04 та драйвер двигунів L298N. Запропоновано прототип програмного забезпечення мовою C++ у середовищі Arduino IDE, який забезпечує циклічне зчитування даних із сенсорів, визначення відхилення від заданого азимуту та формування керуючих сигналів для двигунів на основі нечіткого виводу.

Отже, поставлену в роботі мету досягнуто, а результати дослідження підтвердили можливість побудови працездатної системи керування рухом колісного робота по азимуту засобами нечіткого керування.

Подальший розвиток системи може бути спрямований на вдосконалення алгоритмів фільтрації показань датчиків, розширення бази правил нечіткого керування та впровадження засобів дистанційного моніторингу стану системи.

СПИСОК ПОСИЛАНЬ

- 1) Освітньо-професійна програма «Технології штучного інтелекту» першого (бакалаврського) рівня освіти за спеціальністю 126 Інформаційні системи та технології галузі знань 12 Інформаційні технології. URL: https://kart.edu.ua/wp-content/uploads/2022/08/opp_bak_126_tshi_23-24_pidpisi-1.pdf (Дата звернення: 21.05.2026)
- 2) Петренко Т.Г. Методичні матеріали до порядку та змісту підготовки, оформлення та захисту пояснювальної записки кваліфікаційної роботи. Харків: УкрДУЗТ, 2020. - 15 с.
- 3) А. О. Каргін, Р.С. Кузьменко. Модель ймовірісно-нечіткого актуатора в контурі м'якого управління автономною безлюдною системою. / Сучасний стан наукових досліджень та технологій в промисловості. 2025. № 1 (31), с.61-72, DOI: <https://doi.org/10.30837/2522-9818.2025.1.061>
- 4) Abueejela Y. Moh., Ali H. A. Wheeled Mobile Robot Obstacle Avoidance Using Compass and Ultrasonic. Universal Journal of Control and Automation. 2018. Vol. 6, No. 1. P. 13-18. DOI: 10.13189/ujca.2018.060102. URL: https://www.hrpub.org/journals/article_info.php?aid=6926 (Last accessed: 10.06.2026).
- 5) Tracking control of 3-wheels omni-directional mobile robot using fuzzy azimuth estimator. ACM Digital Library. 2010. URL: <https://dl.acm.org/doi/10.5555/1844600.1844608> (Last accessed: 10.06.2026).
- 6) Tzafestas S. G. Mobile Robot Control and Navigation: A Global Overview. Journal of Intelligent & Robotic Systems. 2018. Vol. 91, No. 1. P. 35-58. DOI: 10.1007/s10846-018-0805-9. URL: <https://link.springer.com/article/10.1007/s10846-018-0805-9> (Last accessed: 10.06.2026).
- 7) Monte Carlo localisation of a mobile robot using a Doppler-Azimuth radar. ScienceDirect. URL:

<https://www.sciencedirect.com/science/article/abs/pii/S0005109818304023>
(Last accessed: 10.06.2026).

- 8) Smart Navigation Robot Using Arduino. URL: <https://tijer.org/tijer/papers/TIJERC001024.pdf> (Last accessed: 10.06.2026).
- 9) Obstacle Avoiding Autonomous Robot. URL: <https://tijer.org/tijer/papers/TIJER2304394.pdf> (Last accessed: 10.06.2026).
- 10) Understand Anything. URL: <https://notebooklm.google/> (Last accessed: 21.05.2026).
- 11) Unified Modeling Language. URL: <https://www.uml.org/> (Last accessed: 21.05.2026).
- 12) UML Use Case Diagrams. URL: <https://www.uml-diagrams.org/use-case-diagrams.html> (Last accessed: 21.05.2026).
- 13) Kosko B. Fuzzy cognitive map. Int. J. Man-Machine Studies. 1986, pp. 65-75.
- 14) Fuzzy Cognitive Map. Mental Modeler. URL: <https://www.mentalmodeler.com> (Last accessed: 21.05.2026).
- 15) Welcome to MATLAB. URL: <https://matlab.mathworks.com/> (Last accessed: 21.05.2026).
- 16) Get Started with MATLAB. URL: <https://www.mathworks.com/help/matlab/getting-started-with-matlab.html> (Last accessed: 21.05.2026).
- 17) Fuzzy Logic Toolbox. URL: <https://www.mathworks.com/help/fuzzy/index.html> (Last accessed: 21.05.2026).
- 18) How to Draw UML Use Case Diagram with Free Online Software? URL: <https://online.visual-paradigm.com/knowledge/use-case-diagram/free-use-case-diagram-software/> (Last accessed: 21.05.2026).
- 19) What is a Deployment Diagram? URL: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-deployment-diagram/> (Last accessed: 21.05.2026).
- 20) Fritzing. URL: <https://fritzing.org/> (Last accessed: 21.05.2026).

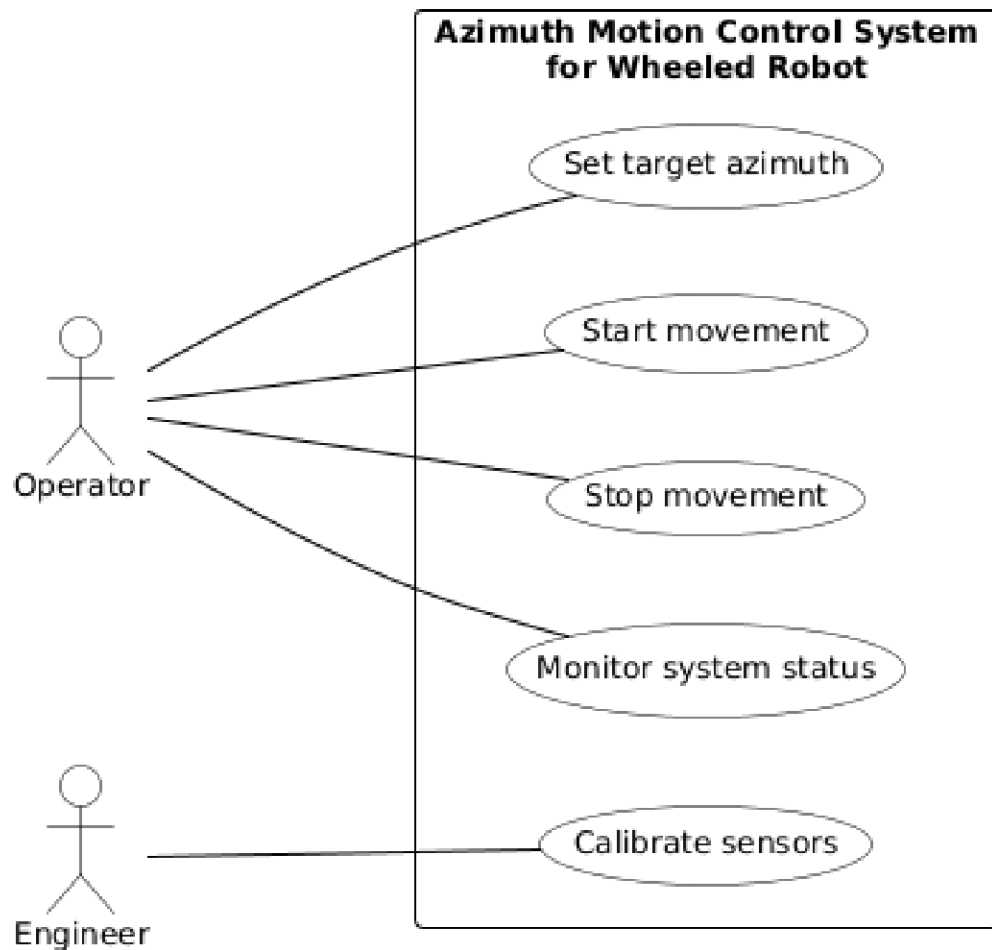


Рисунок А.1 - Діаграма варіантів використання (Use Case)

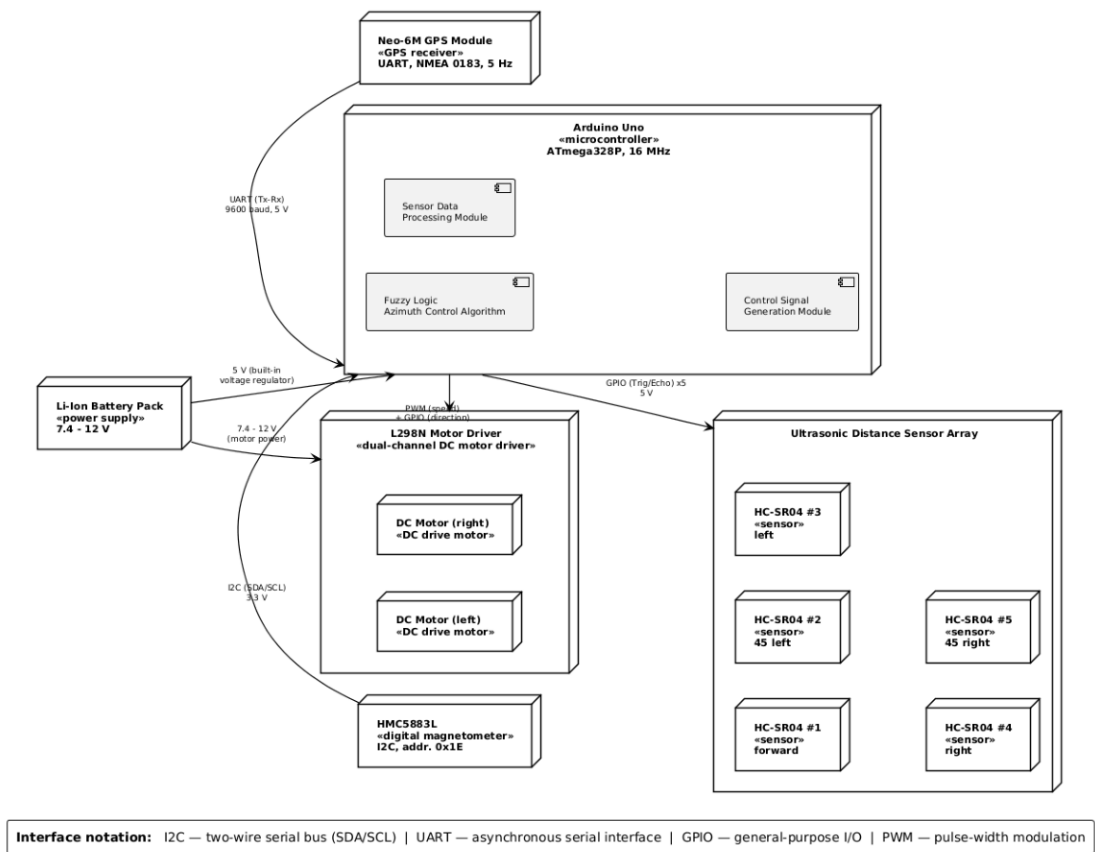


Рисунок А.3 - Діаграма розгортання (Deployment Diagram)

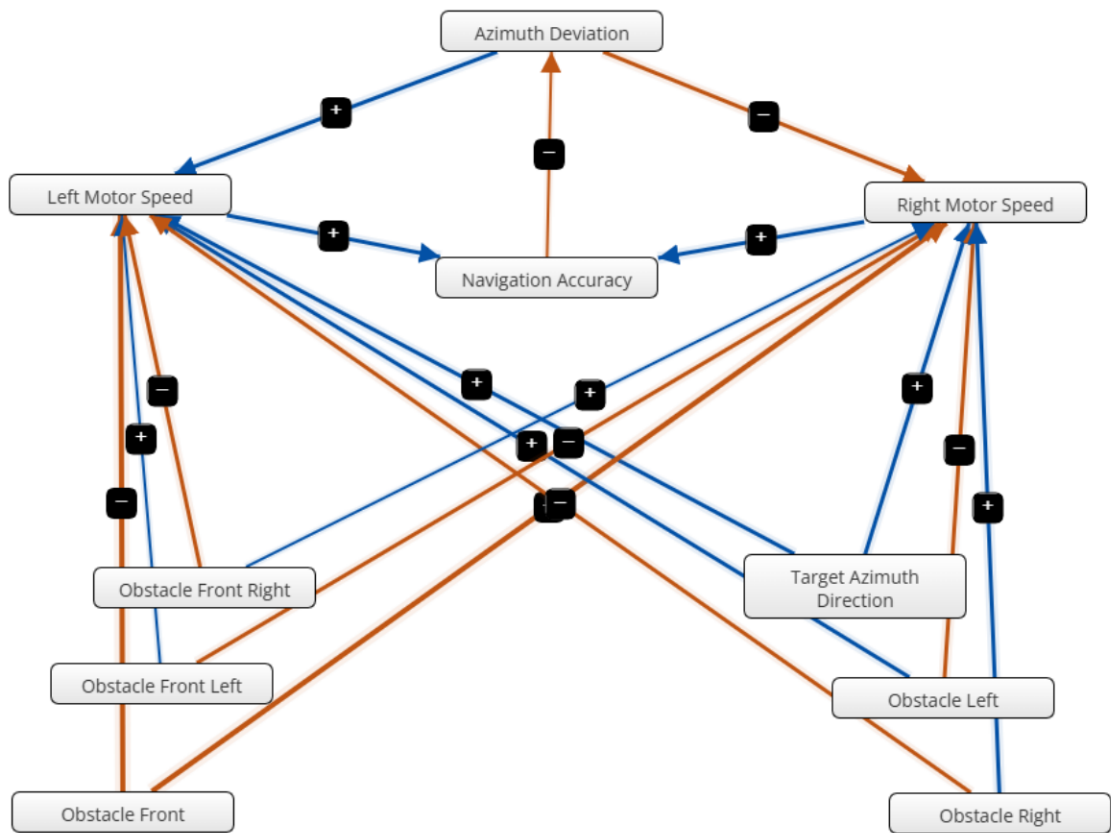


Рисунок Б.1 - Нечітка когнітивна карта системи керування рухом по азимуту колісного робота в середовищі Mental Modeler

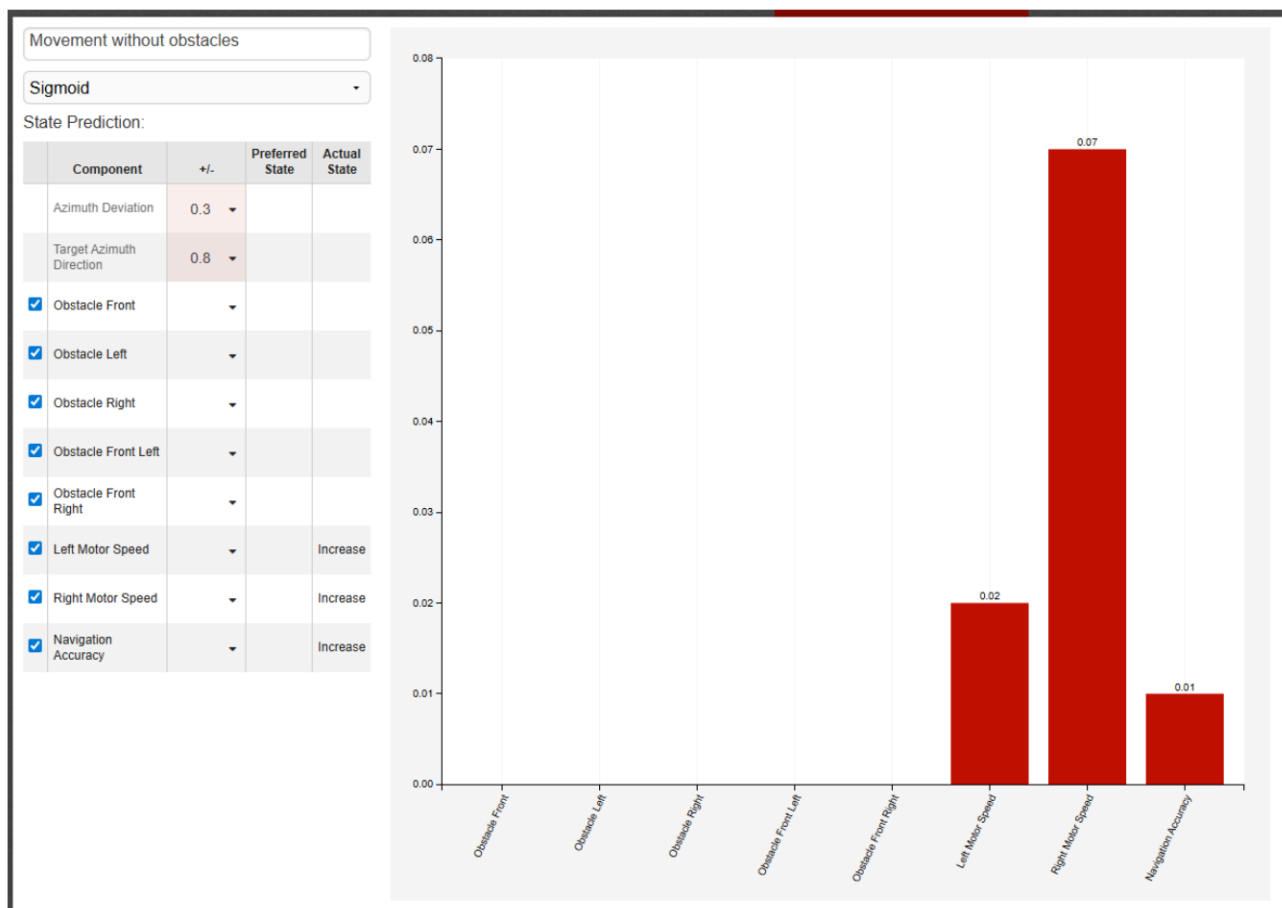


Рисунок Б.2 - Результат моделювання сценарію руху без перешкод у середовищі Mental Modeler

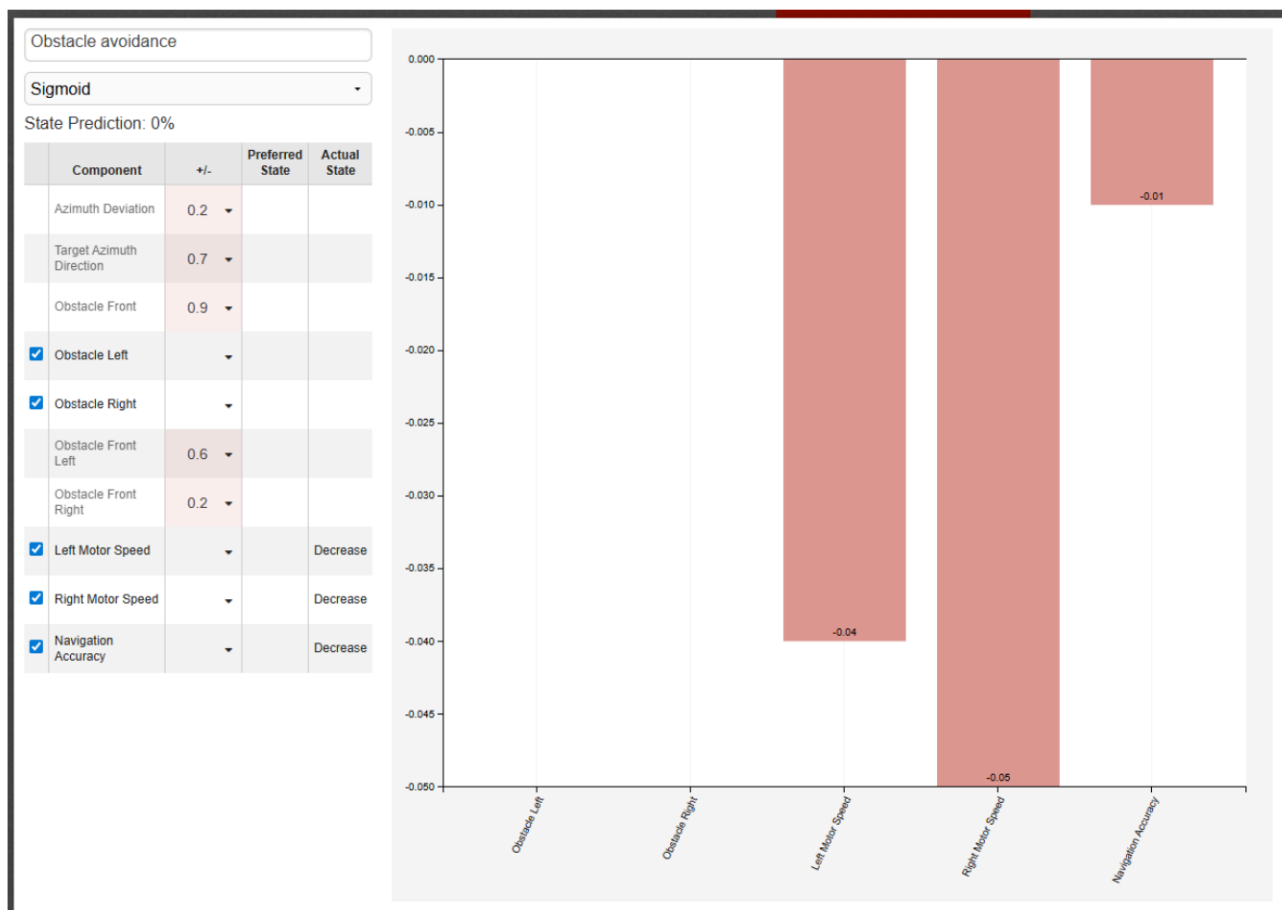


Рисунок Б.3 - Результат моделювання сценарію об'їзду перешкоди у середовищі
Mental Modeler

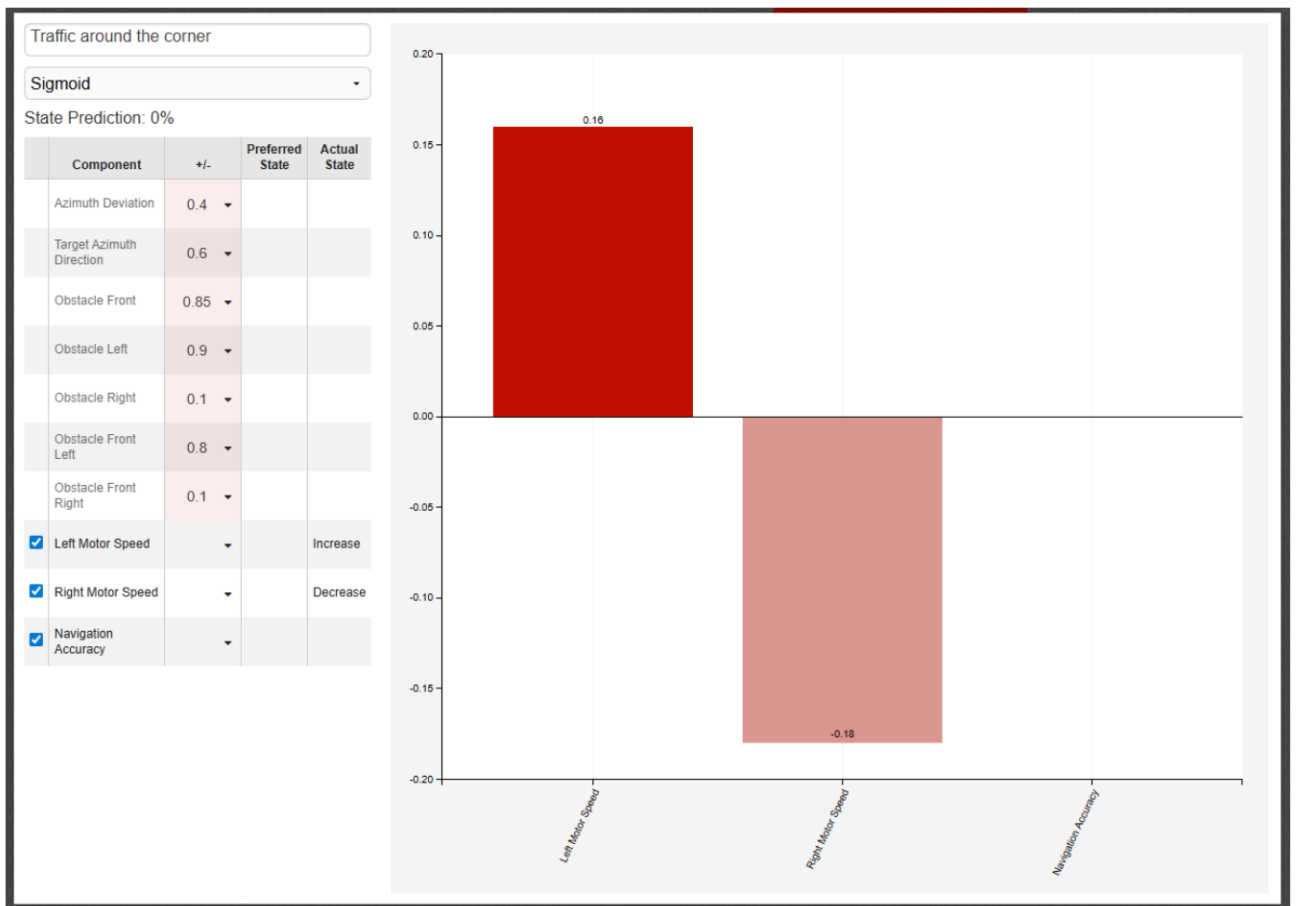


Рисунок Б.4 - Результат моделювання сценарію руху біля кута у Mental Modeler

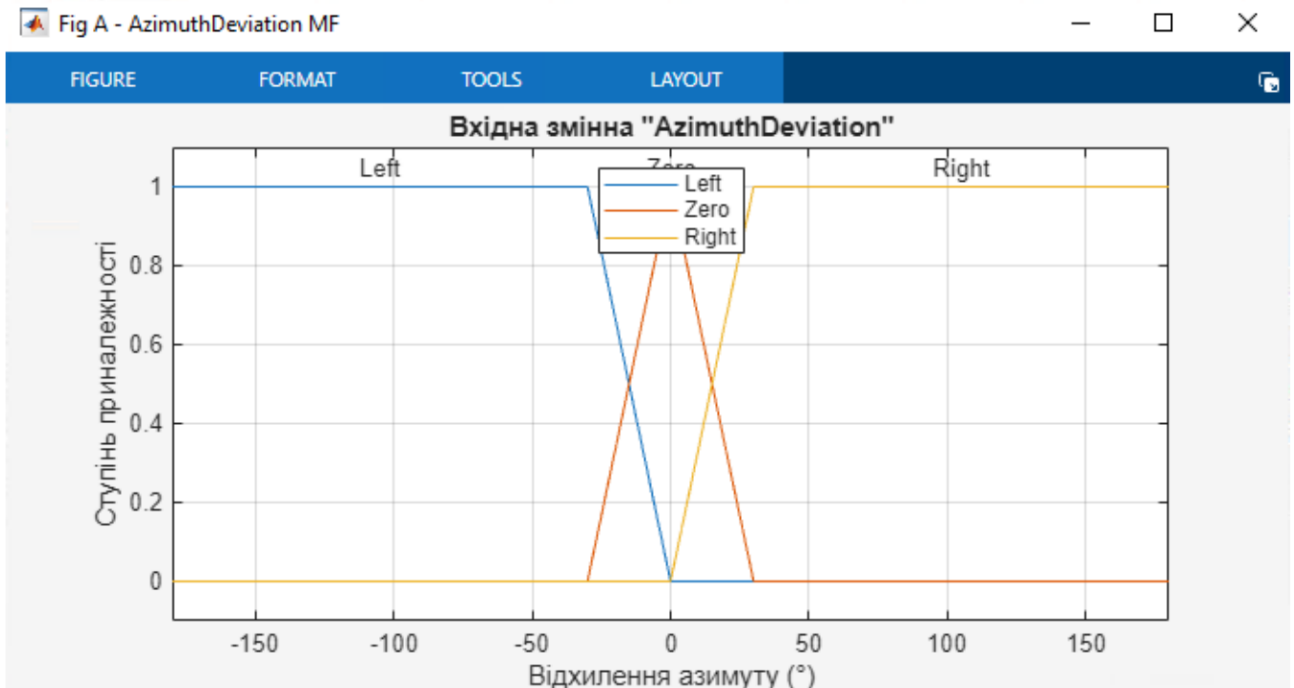


Рисунок Г.1 - Функції приналежності вхідної змінної AzimuthDeviation

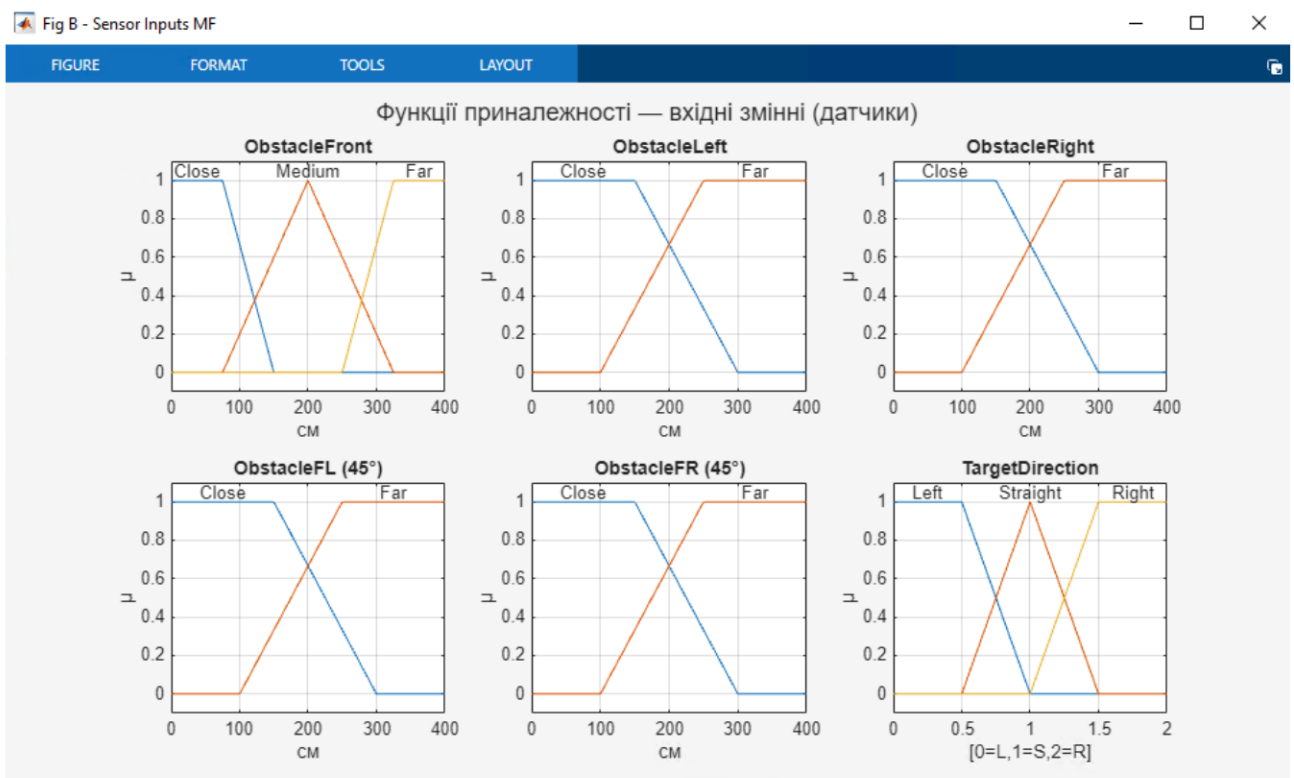


Рисунок Г.2 - Функції приналежності вхідних змінних датчиків

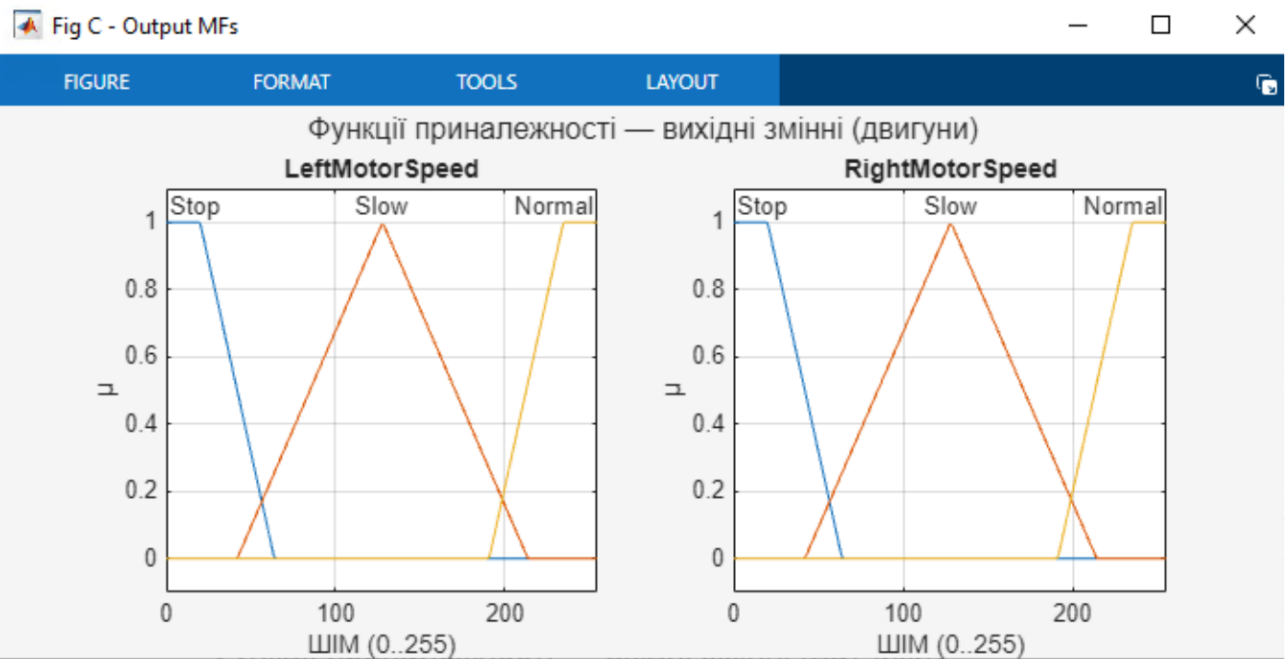


Рисунок Г.3 - Функції приналежності вихідних змінних

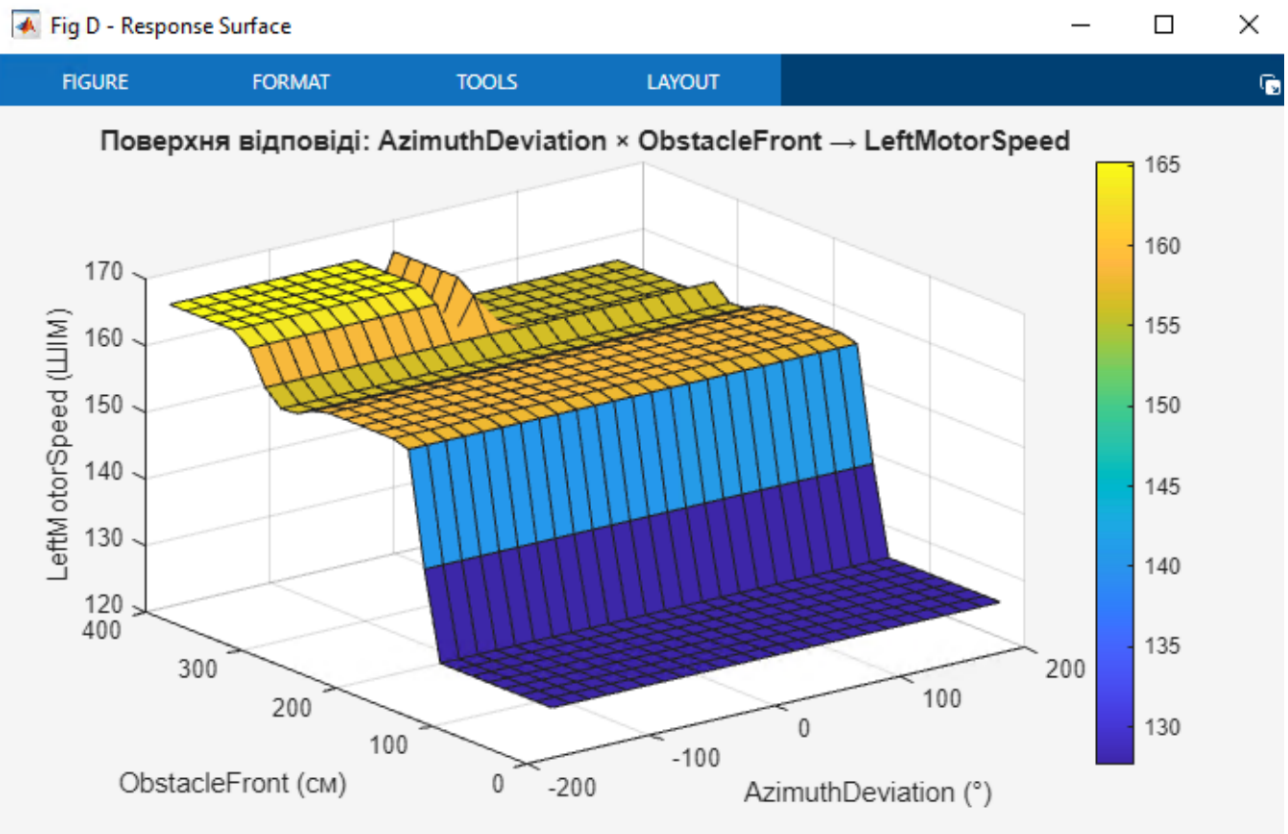


Рисунок Г.4 - Поверхня відповіді

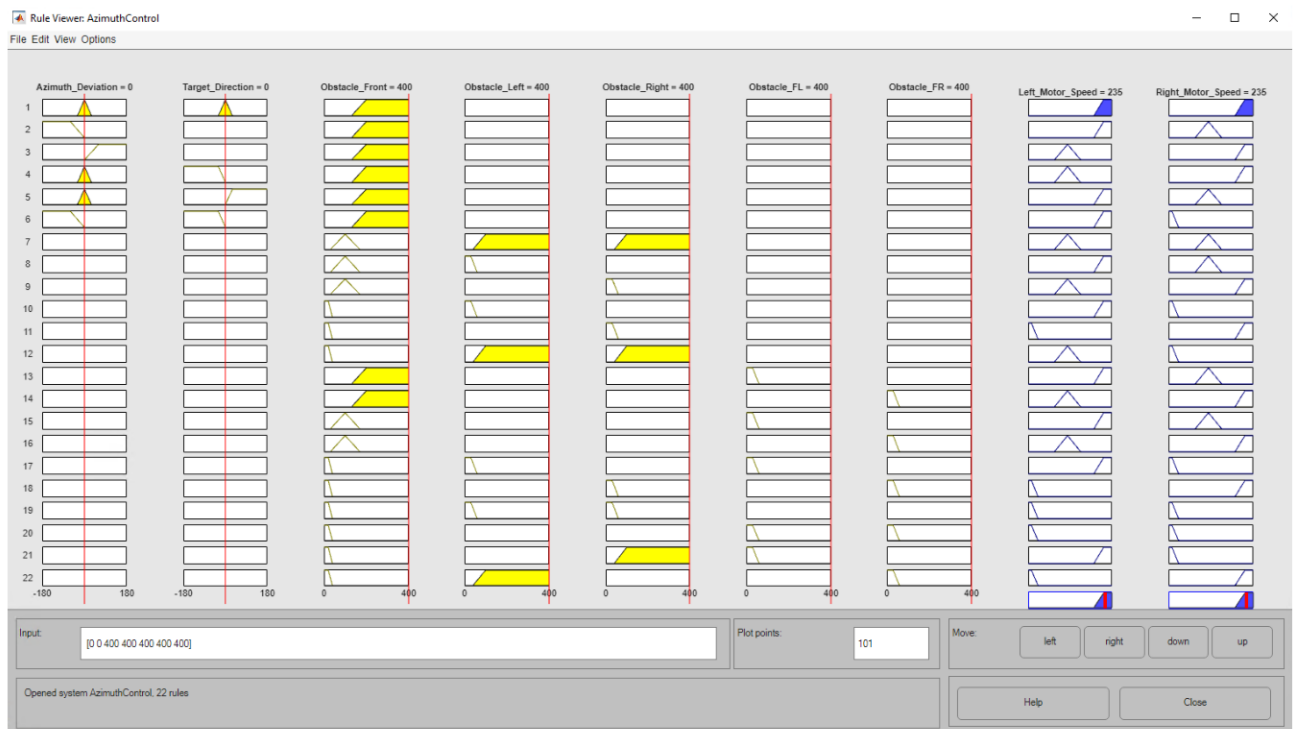


Рисунок Г.5 - Rule Viewer: нормальный рух (без перешкод)

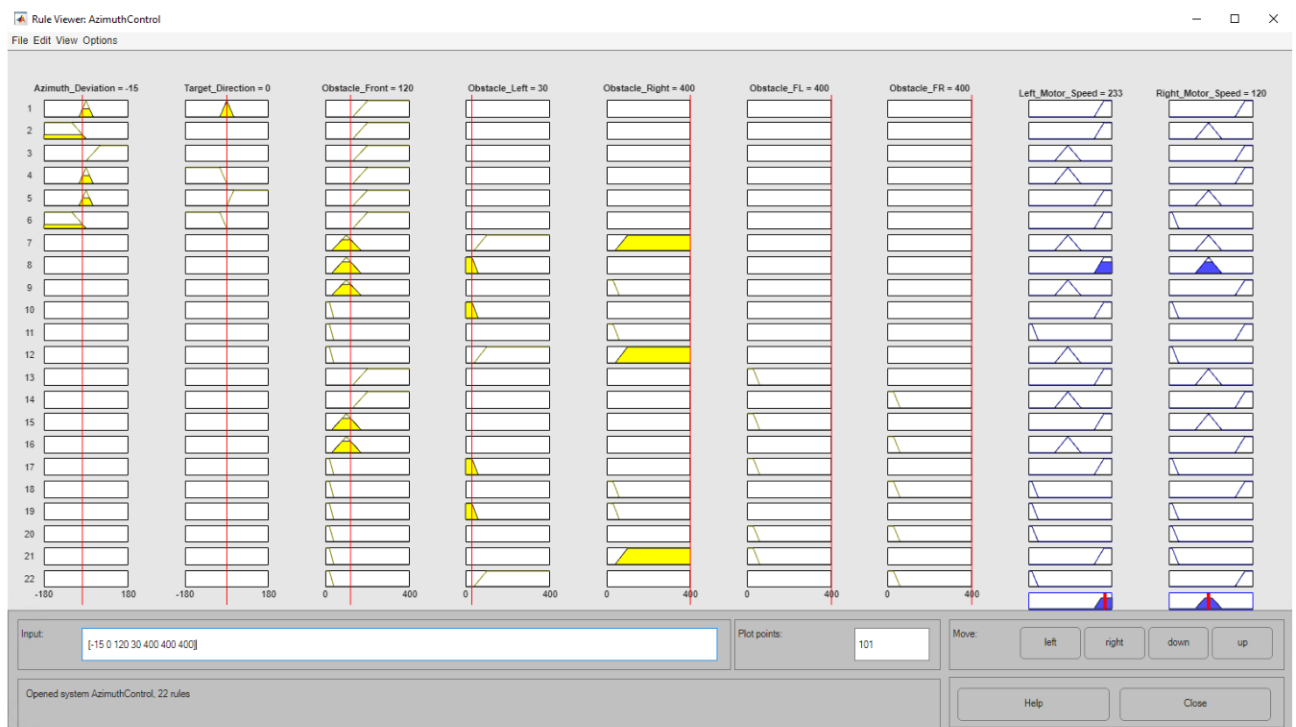


Рисунок Г.6 - Rule Viewer: уникнення фронтальних перешкод

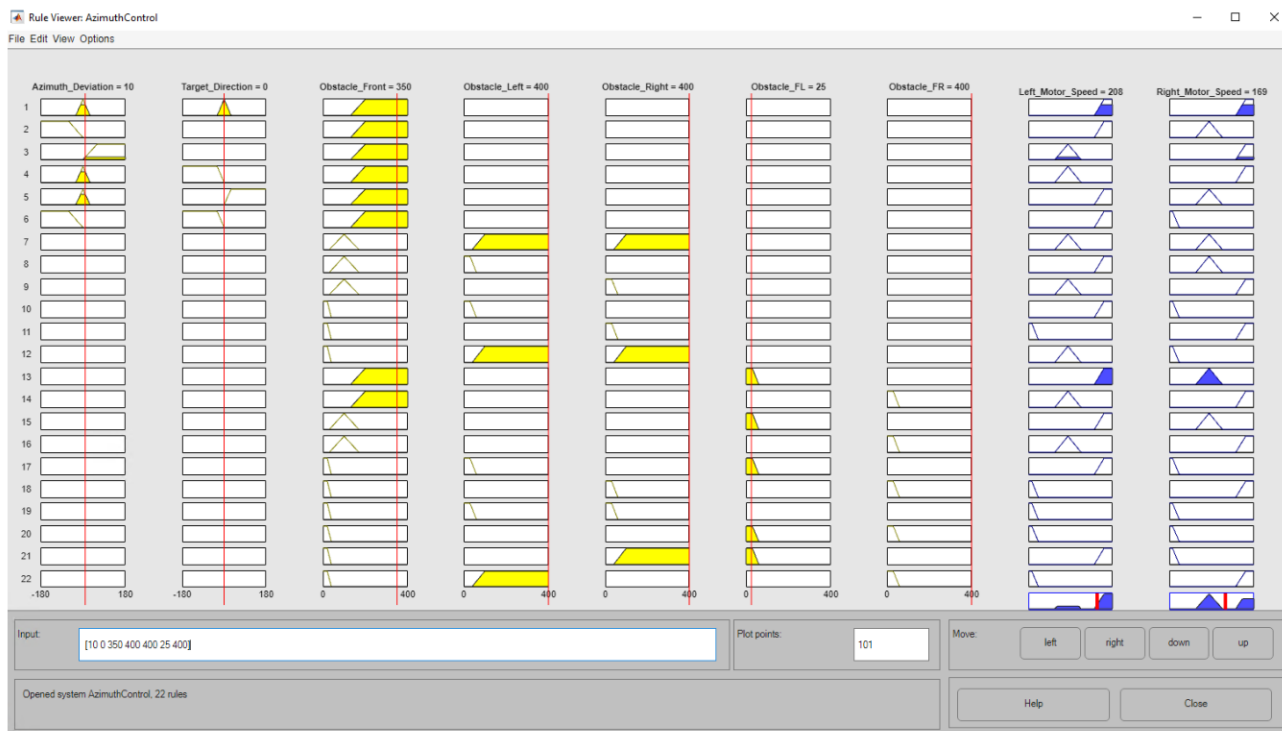


Рисунок Г.7 - Rule Viewer: рух навколо кутових перешкод

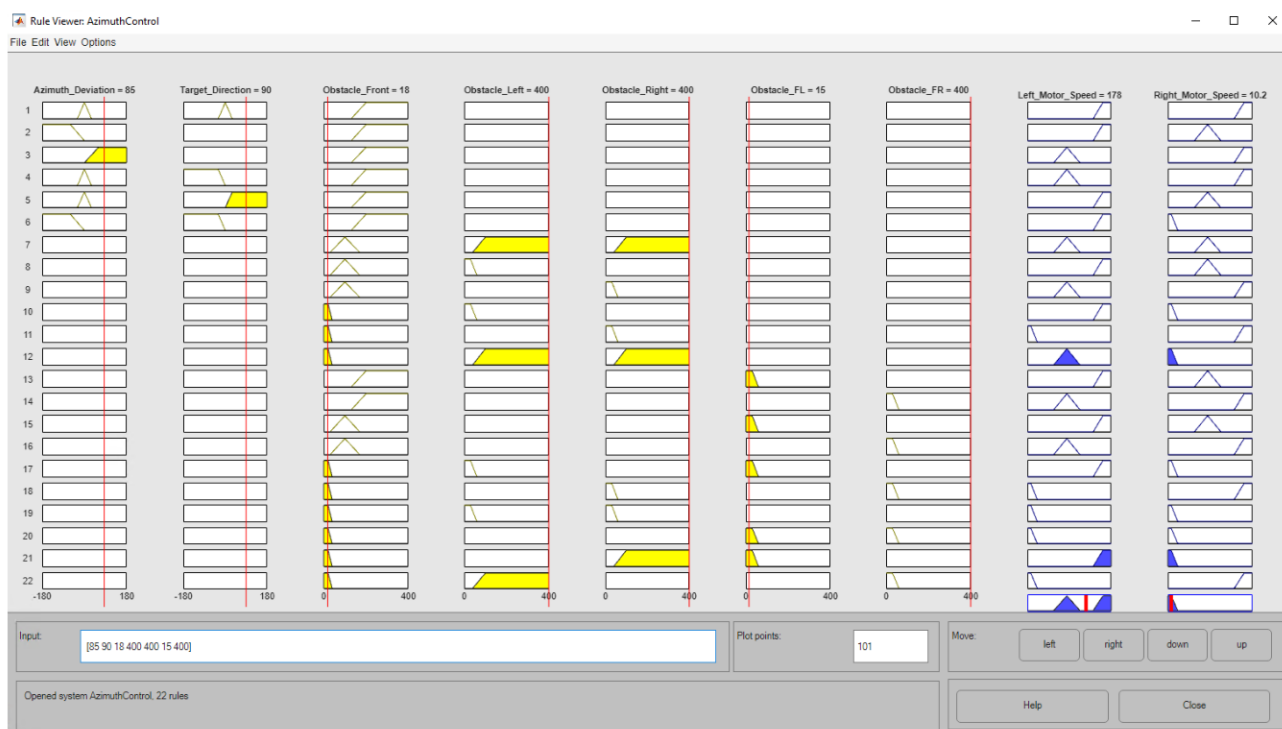


Рисунок Г.8 - Rule Viewer: комбінована критична ситуація

ДОДАТОК Г

МОДЕЛЮВАННЯ АПАРАТНОЇ ЧАСТИНИ ПРОТОТИПУ СИСТЕМИ

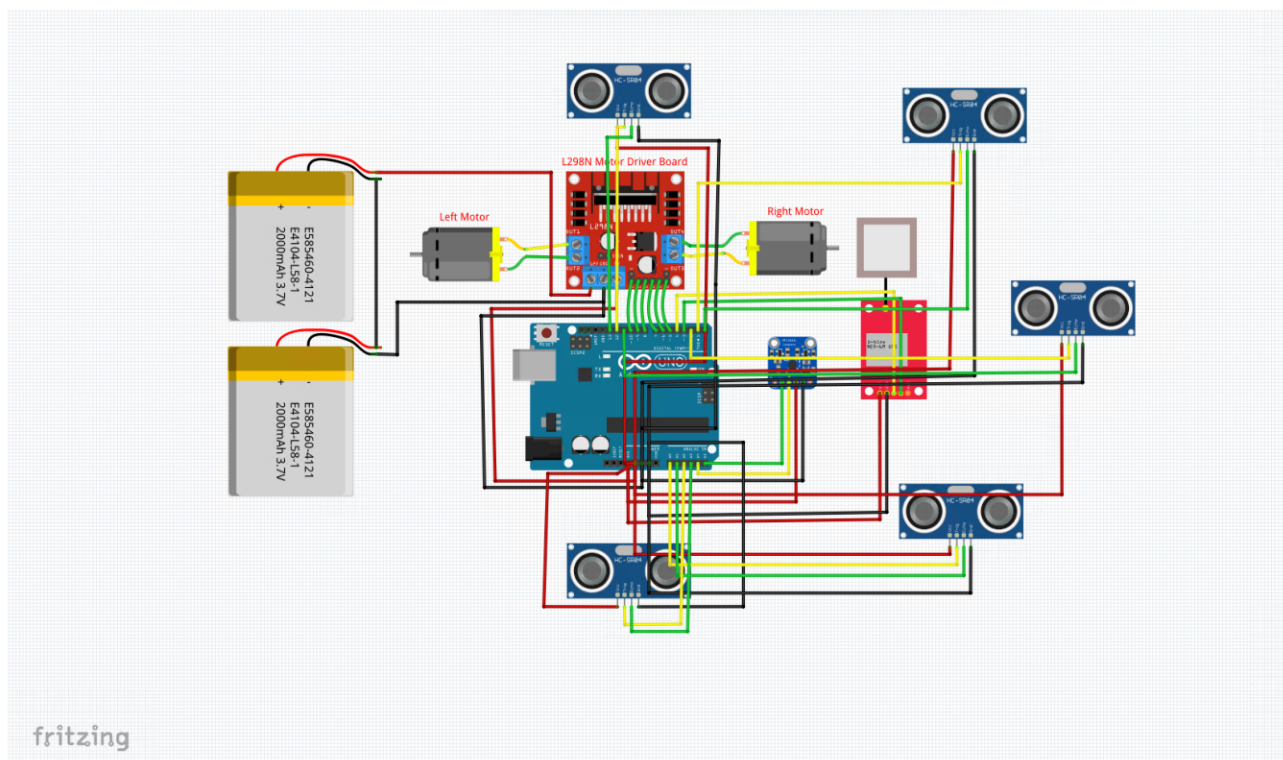


Рисунок Г.1 - Загальна схема апаратного прототипу системи

ДОВІДКА ПРО ОЗНАКИ ПЛАГІАТУ
(збігів/ідентичності/схожості в текстах)
В ПОЯСНЮВАЛЬНІЙ ЗАПИСЦІ ДО КВАЛІФІКАЦІЙНОЇ РОБОТИ
ЗДОБУВАЧА ПЕРШОГО (БАКАЛАВРАТСЬКОГО) РІВНЯ ВИЩОЇ ОСВІТИ

ПІБ студента

Довженко В'ячеслав Сергійович

Тема роботи	Система керування рухом по азимуту колісного робота
Сервіс перевірки на плагіат	StrikePlagiarism
Робота містить авторського тексту	99,25%
Перевірку виконала (ПІБ)	к.т.н, доцент, доцент кафедри інформаційних технологій УкрДУЗТ Петренко Тетяна Григорівна
Дата, підпис	19.06.26 

StrikePlagiarism.com

☰ ✓ Документ прийнятий

i

Метадані

📄 ДОКУМЕНТ

Заголовок
Dovzhenko_QW

Автор
Довженко В'ячеслав Сергійович

Науковий керівник / Експерт
Petrenko

ІД документа
334356582

🏢 ОРГАНІЗАЦІЯ

Назва організації
Ukrainian State University of Railway Transport

Підрозділ
Кафедра "Інформаційні технології"

📄 ЗВІТ

Дата звіту
2026-06-17

Дата редагування
2026-07-02

Обсяг знайдених подібностей

Коефіцієнт подібності визначає, який відсоток тексту по відношенню до загального обсягу тексту було знайдено в різних джерелах. зверніть увагу, що високі значення коефіцієнта не автоматично означають плагіат. Звіт має аналізувати компетентна / уповноважена особа.

0.75%

0.75%

КП 1