

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Тернопільський національний економічний університет
Факультет комп'ютерних інформаційних технологій
Кафедра комп'ютерної інженерії

Радь Назарій Михайлович

**Програмне забезпечення управління мобільним
роботом на основі клієнт-серверної архітектури /
The software application for mobile robot control
based on client server architecture**

напрямок підготовки: 123 Комп'ютерна інженерія
фахове спрямування - Комп'ютерна інженерія
Бакалаврська робота

Виконав студент групи КСМ-43/2
Назарій Михайлович Радь

Науковий керівник:
к.т.н. , О.П. Адамів

Тернопіль - 2018

РЕЗЮМЕ

Бакалаврська робота містить 68 сторінок пояснюючої записки, 32 рисунок, 12 таблиць, 2 додатки та 2 аркуші формату А3.

Метою баклаврської роботи є розроблення програмного забезпечення управління мобільним роботом на основі клієнт-серверної архітектури

З метою дослідження розробленого методу навігації використано програмний симулятор мобільних роботів компанії ActivMedia MobileSim, що дозволяє проводити дослідження методів навігації з різними базами мобільних роботів Pioneer 1, Pioneer AT, Pioneer 2™ -DX, -DXe, -DXf, -CE, -AT, Pioneer 2™-DX8/Dx8 Plus і -AT8/AT8 Plus

Для побудови карти середовища використано також продукт компанії ActivMedia Mapper3, що дозволяє побудувати статичне середовище, з визначенням початкового положення робота та координатів цілі

Для здійснення управління мобільним роботом реалізовано програмне забезпечення отримання інформації від одометричних двигунів АМР та проведення співставлення реального місця знаходження робота та інформації від системи управління автономним мобільним роботом АМР. Для програмної реалізації і експериментальних досліджень методу управління в якості програмного засобу використано пакет Matlab.

Ключові слова: **МОБІЛЬНИЙ РОБОТ, НАВІГАЦІЯ, КАРТА СЕРЕДОВИЩА.**

RESUME

The bachelor's thesis contains 68 pages of explanatory note, 32 figures, 12 tables, 2 appendices and 2 A3 spreadsheets.

The purpose of the baccalaureate work is to develop mobile robot control software based on client-server architecture

In order to study the developed navigation method, the software simulator of mobile robots from ActivMedia MobileSim was used, which allows to study navigation methods with different bases of mobile robots Pioneer 1, Pioneer AT, Pioneer 2™ -DX, -DXe, -DXf, -CE, -AT, Pioneer 2™ -DX8 / Dx8 Plus and -AT8 / AT8 Plus

ActivMedia Mapper3's product was also used to build the environment map, which allows you to build a static environment, with the definition of the initial position of the robot and the coordinates of the target

To control the mobile robot, the software for obtaining information from ODM motors and comparing the real location of the robot and information from the control system of the autonomous mobile robot AMP was implemented. Matlab package was used as a software tool for software implementation and experimental research of the control method.

Keywords: MOBILE WORK, NAVIGATION, ENVIRONMENT MAP.

ЗМІСТ

Вступ.....	10
1 Аналіз методів управління та планування.....	11
рухом мобільного робота	11
1.1 Огляд методів управління мобільним роботом	11
1.2 Методи планування руху мобільного робота.....	16
1.3 Узагальнена структура системи управління	26
2 Алгоритмічне та інформаційне забезпечення системи управління мобільним роботом	33
2.1 Алгоритм планування рухом мобільного робота для неструктурованого середовища.....	33
2.2. Архітектура системи управління мобільного робота.....	42
2.3 Програмне забезпечення клієнт-серверної взаємодії	49
3 Моделювання та тестування програмної системи управління.....	56
3.1 Вибір технології та програмного середовища	56
3.2 Імітаційне моделювання програмного забезпечення управління.....	60
3.3 Результати експериментальних досліджень на базі мобільного робота Amigo Bot.....	62
Висновки	68
Список використаних джерел.....	69

					ДП.КСМ. 07254/16.00.00.000.ПЗ			
Змн.	Лист	№ докум.	Підпис	Дата				
Розробив	Радь Н.М..				ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ УПРАВЛІННЯ МОБІЛЬНИМ РОБОТОМ НА ОСНОВІ КЛІЄНТ- СЕРВЕРНОЇ АРХІТЕКТУРИ	Літ.	Арк.	Акрушів
Перевір.	Адамів О. П.					8	72	
Консульт.	Паздрій І.Р.					ТНЕУ,ФКІТ, КСМ-43/2		
Н. Контр.	Гураль І.В.							
Затвердив	Березький О.М.							

ПЕРЕЛІК ПОЗНАЧЕНЬ

АРМ - автономний мобільний робот

ООП - об'єктно-орієнтоване програмування

СОМ - Component Object Model

ПЗ – програмне забезпечення

ПК – персональний комп'ютер

МЗД – модуль збору даних

ПЗП – постійно запам'ятовуючий пристрій

ОЗП – оперативно-запам'ятовуючий пристрій

ЦП – центральний процесор

ЛКС – локальна карта середовища

ГКС – глобальна карта середовища

					ДП.КСМ. 07254/16.00.00.000 ПЗ	9
Змн.	Арк.	№ докум.	Підпис	Дат		

ВСТУП

На даний час застосування робототехнічних систем в різних галузях науки і виробництва зростає з великою швидкістю. Особливе місце серед даних систем займають мобільні роботи, що здатні рухатись і виконувати чітко визначені задачі. При цьому виникає проблема адаптації мобільного робота до середовища, що його оточує. Процес формування карти середовища дозволяє сформувати інформацію про середовище з певною невизначеністю. При цьому виникає задача коректного управління мобільним роботом при певній ентропії.

Основна увага тут приділяється системі навігації робота. Побудова ефективної системи управління при цьому дозволяє скоротити затрати часу на орієнтування в середовищі.

					ДП.КСМ. 07254/16.00.00.000 ПЗ	10
Змн.	Арк.	№ докум.	Підпис	Дат		

Для визначення місцеположення наземних об'єктів широко застосовуються одометричні навігаційні системи (ОНС), де швидкість руху на кожній прямолінійній ділянці шляху вимірюється за кількістю обертів коліс (трансмисії) транспортного засобу. Курсовий кут на такій ділянці визначається за допомогою гірокомпаса. Як правило, до складу ОНС входять гірокомпаси, гірокурсказівники, обчислювальні блоки; одометричні сенсори. До таких систем відносяться німецька ОНС FNA-615, англійська ОНС LNS-202, російські ОНС ТНА-4, 15Ш55 і інші.

Сучасні одометричні навігаційні системи мають граничну відносну похибку визначення координат порядку 1.5% від пройденого шляху за 7 годин роботи (ТНА-3) або за 5 годин руху (ТНА-4) [25].

Для зняття помилок визначення місця положення, що нагромаджуються, здійснюється прийом контрольного орієнтування. Його проводять на опорних пунктах, координати яких відомі. Цей спосіб дозволяє визначити координати будь - якого об'єкту і дирекційний кут напрямку руху (прицілювання) з досить високою точністю (25-30м і 0-04 п.к. відповідно), але він має ряд недоліків. Зазначимо деякі з них.

Опорних пунктів може не бути в зоні прямої видимості. Вночі їх не видно, вони можуть бути приховані пилом, туманом, димом, снігом. Під'їзд до них інколи неможливий. Також при задачі дослідження невідомої території чи при дослідженні планет опорних пунктів може і не існувати.

Вище вказане переконує в тому, що одометричні навігаційні системи не можуть використовуватися як основні системи навігації.

В навігації рухомих об'єктів (морських, повітряних, наземних) продовжують експлуатуватися наземні РНС "LORAN-C", "LORAN-D", "Тропик-2", "Тропик-2П" і інші.

На думку фахівців наземні РНС типу LORAN-3, LORAN-D будуть знаходитися в експлуатації "досить тривалий час" [23].

В Україні ще не стоїть питання про припинення експлуатації систем типу "Тропик-2", "Тропик-2П".

Супутникові радіонавігаційні системи (СРНС) (Глобальні системи точного визначення місцеположення) - інтерес до СРНС викликаний їх універсальністю. У рамках однієї системи можливе рішення великого комплексу різних задач.

Найбільш перспективними є СРНС "NAVSTAR" (США) і "ГЛОНАСС" (Росія) [24].

Потреба в оперативній високоточній навігації сухопутних, морських, повітряних об'єктів обумовила створення в 80-90-ті роки середньоорбітальних СРНС GPS "NAVSTAR" у США і ГЛОНАСС в Росії.

Основне призначення супутникових радіонавігаційних систем другого покоління - глобальна, оперативна навігація сухопутних, морських, повітряних об'єктів, забезпечення можливості в будь-якій точці земної поверхні, у будь-який час року і доби, за будь-якої погоди визначити (уточнити) параметри рухомого об'єкту - три координати і три складові вектора швидкості.

Хоча на ринку є багато комерційно доступних GPS-приймачів, їх застосування в глобальній навігації поки обмежене завданнями дотримання загального курсу. Пов'язано це з очевидною вимогою законів робототехніки про точність такої навігації - помилка у визначенні власних координат не може перевищувати розміру автономного апарату (інакше можливі зіткнення з пристроями такого ж або меншого розмірів і інші конфлікти з середовищем). Типовий самохідний комерційний робот звичайно не перевищує в довжину одного-двох метрів і може віддалятися від місця старту на 10 км, а ось GPS-сигнал дає точність близько 100 м, і цивільним організаціям найближчими роками будуть доступні оновлені GPS-приймачі з точністю 20-30 м [23]. Тому як базова GPS-навігація застосовується переважно в автопілотах великих літаків або океанських лайнерів. Крім того,

в різних регіонах Землі, на місцевості з складним рельєфом і в будівлях GPS-сигнал може прийматися нестійкий і з перешкодами. Таким чином, ця система ще досить довго не зможе використовуватися як основна в завданнях глобальної навігації мобільних роботів.

Результатом роботи будь-якої навігаційної системи є представлення середовища функціонування АМР, виділяють два основних методи:

1) Топологічний метод - топологічна карта представляє середовище у формі графу, в якому кожна вершина відповідає різним позиціям і дуги представляють сусідство між ними [56]. Основною перевагою топологічного представлення є його компактність. Топологічна карта може будуватись на основі сітко-базованої карти. Основна ідея проста: не зайняті положення сітко-базованої карти поділяються на невелике число областей, розділених критичними лініями, які відповідають вузьким проходам середовища, наприклад, дверні проходи та ін. Таким чином карта середовища є ізоморфним графом. Компресія інформації є надзвичайно великою. Наприклад, топографічний граф містить 67 вершин, беручи до уваги те, що сітко-базована карта середовища містить 27280 зайнятих комірок. Зауважимо, що критичні лінії є досить зручними для декомпозиції метричної карти. По-перше, коли слідувати через критичні лінії, то робот буде переміщуватись у відносно малій області. Тому, втрата в продуктивності є меншою, ніж в сітково-базованій карті. По-друге, вузькі області блокуються перешкодами з більшою імовірністю (наприклад, двері, які можуть бути зачинені або відчинені);

2) Метод представлення карти середовища в полярній системі координат - вказує у визначеному кутовому діапазоні відстані до перешкод [123]. Перевагами такого методу є невеликий розмір і простота представлення перешкод, що важливо при функціонування АМР у реальному часі.

Проведений аналіз показав, що в АМР доцільно використовувати ОСН, а також в залежності від задач наземні чи супутникові радіонавігаційні системи, та формувати карту середовища мобільного робота для локальної навігації в полярній системі координат, а для глобальної навігації використовувати топологічний граф.

1.2 Методи планування руху мобільного робота

Середовище функціонування мобільного робота можна розділити на два типи: структуроване (напер відоме) і неструктуроване (складне, наперед невідоме середовище). В залежності від середовища функціонування виділяють глобальні та локальні методи навігації. Якщо середовище є наперед відоме і задана ціль лежить в межах відомого середовища, то для навігації мобільного робота застосовують глобальні методи навігації. Якщо ж середовище є наперед невідомим або перед роботом ставиться задача дослідження середовища, в даному випадку застосовуються локальні методи навігації до цілі, що використовують лише локальну інформацію про середовище функціонування отриману з допомогою методів побудови локальної карти середовища, що розглядалися у попередніх параграфах.

Існує багато реалізацій систем навігації, що використовують глобальні методи [8-14]. В даному параграфі розглядаються основні методи глобальної навігації. Алгоритм роботи глобальних методів навігації в цілому складається з двох етапів:

- 1). планування траєкторії руху мобільного робота;
- 2). безпосереднє переміщення до цілі, використовуючи інформацію про необхідну траєкторію, отриману на першому етапі.

Етап планування глобального методу навігації Видимий граф (Visibility Graph) [9,10] складається з трьох підетапів, що забезпечують побудову

оптимальної траєкторії переміщення до цілі, де за критерій оптимальності взято мінімальний шлях:

- 1) встановлення вершин периметру перешкод (рис. 1.2а);
- 2) встановлення всіх можливих шляхів(рис. 1.2 б);
- 3) пошук мінімального шляху переміщення(рис. 1.2 в).

Одним із модифікацій методу Видимий Граф є метод, що забезпечує зменшення кількості ребер графу. При цьому зменшується обчислювальна складність методу. На рисунку 1.3 редставлено порівняння методу Видимий Граф та методу Tangent Видимий Граф [76]

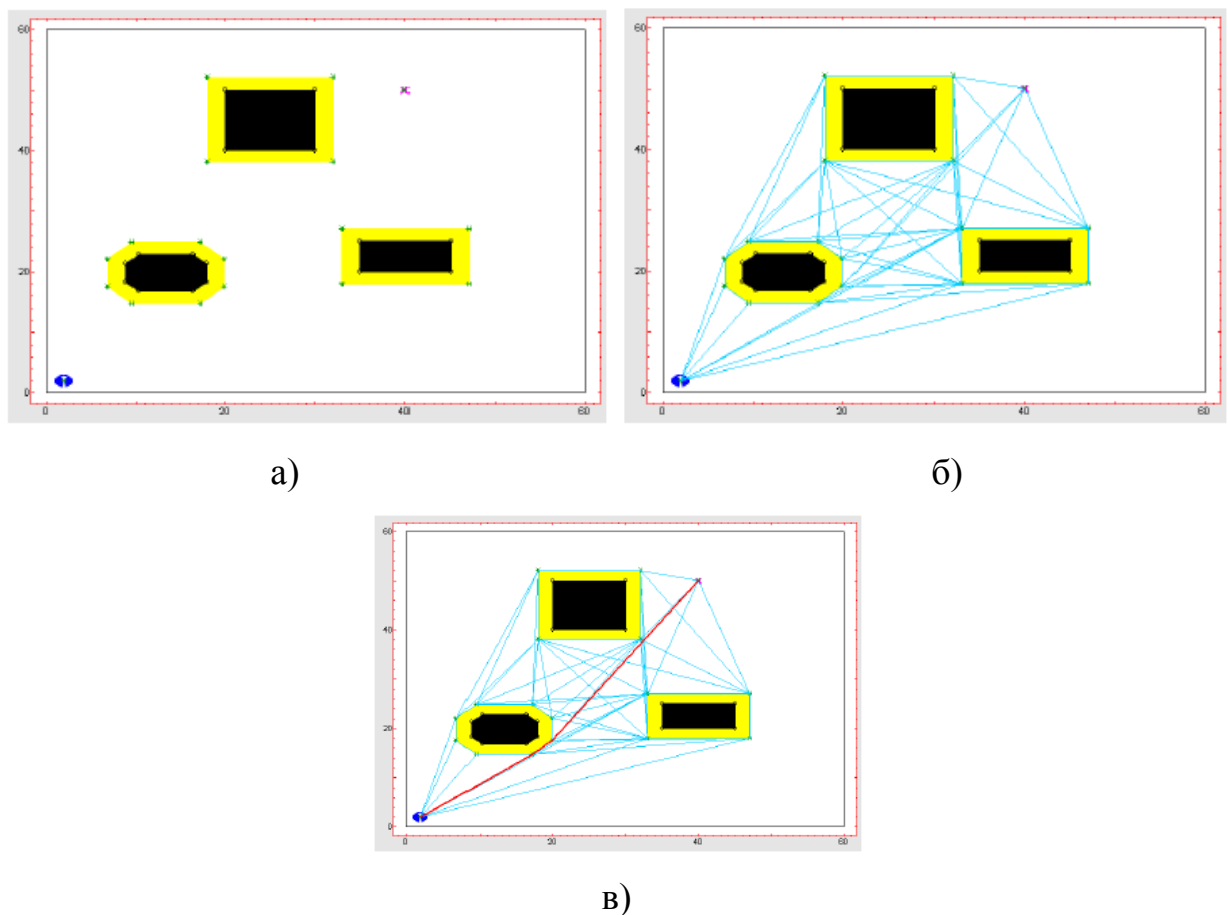
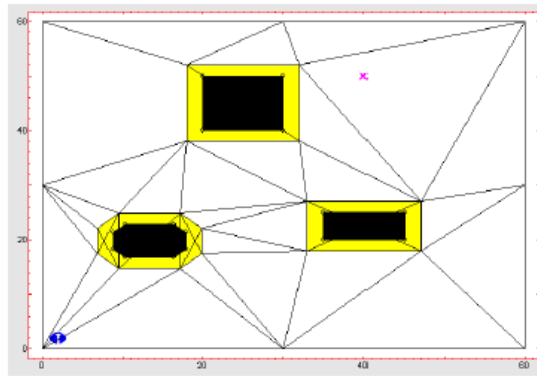
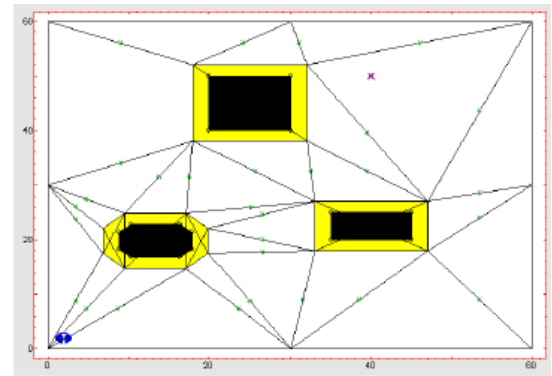


Рисунок 1.2 - Графічне представлення підетапів планування методу глобальної навігації Видимий граф:

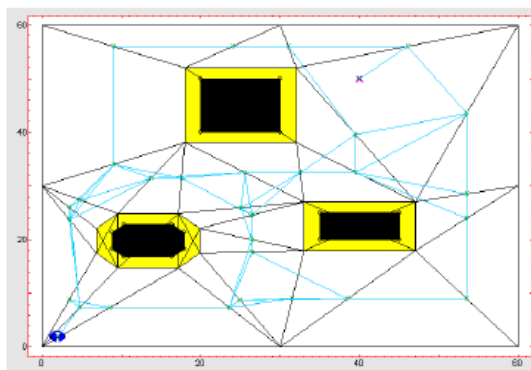
- а) встановлення вершин периметру перешкод; б) встановлення всіх можливих шляхів; в) пошук мінімального шляху переміщення



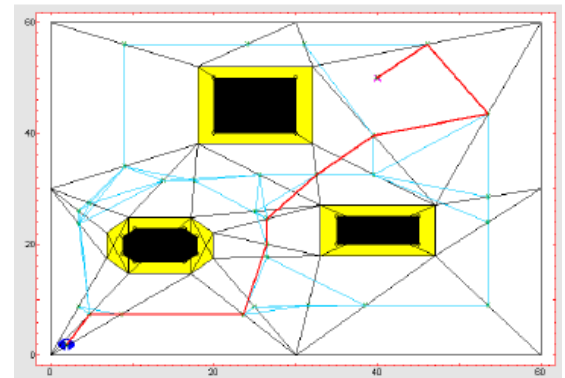
а)



б)



в)



г)

Рисунок 1.4 - Графічне представлення підетапів планування методу глобальної навігації Діаграм Вороного:

- а) триангуляція Делона; б) визначення середини відрізків; в) всі можливі шляхи; г) мінімальний шлях.

Інший глобальний метод навігації, що є дещо схожим до методу Діаграм Вороного є Дерева Квадратів (Quadtrees) [13, 14], що базується на використанні методів представлення великих середовищ з допомогою квадратів. Цей підхід складається також із чотирьох підетапів планування:

- 1). поділ середовища на дерево квадратів, при цьому враховуються периметри об'єктів, що розміщені в структурованому середовищі функціонування мобільного робота (рис. 1.5 а);
- 2). визначення центру отриманих квадратів (рис. 1.5 б);

Змн.	Арк.	№ докум.	Підпис	Дат

3). побудова графа, що зв'язує всі вершини графа (центри квадратів), а також в даний повнозв'язний граф вводяться як вершини початкове положення робота і ціль переміщення (рис.1.5 в);

4). Пошук мінімальної траєкторії руху від початкового положення робота до цілі, використовуючи граф, що отриманий на підетапі 3 (рис.1.5 г).

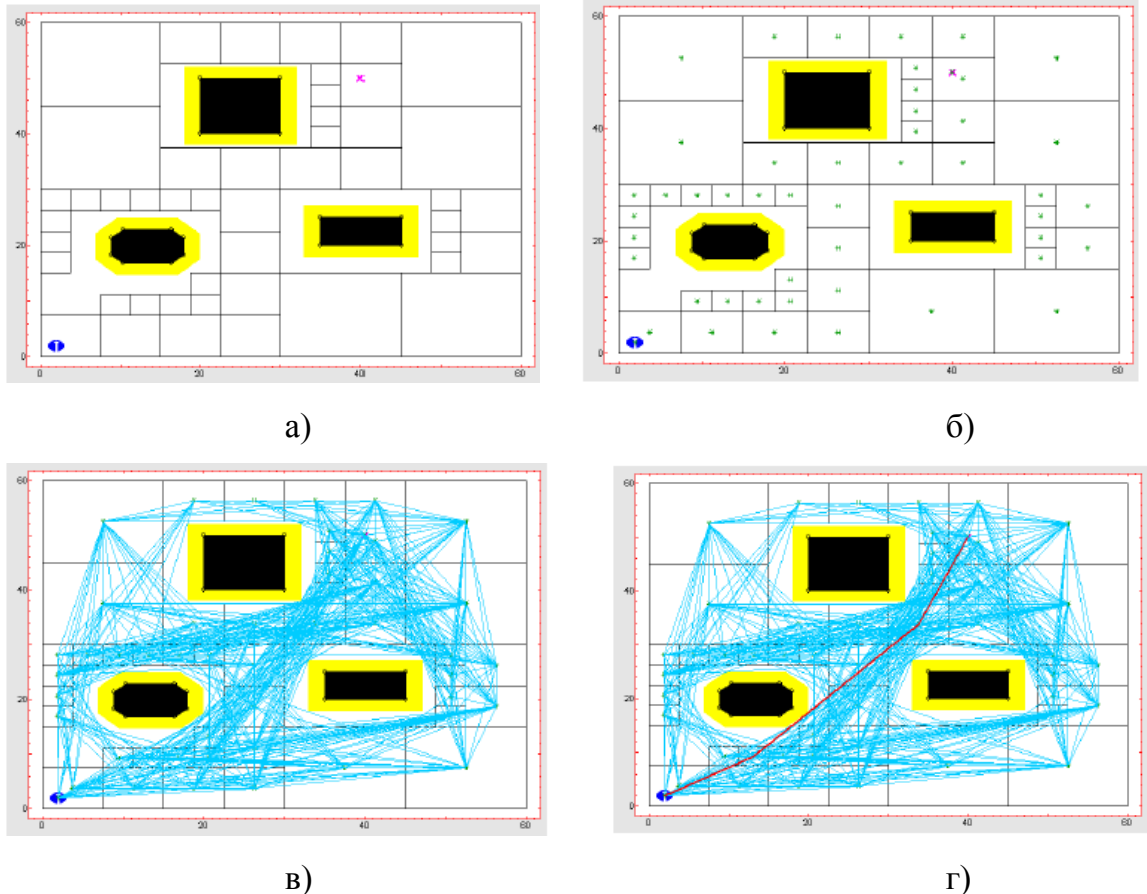


Рисунок 1.5 - Графічне представлення підетапів планування методу глобальної навігації Дерева квадратів:

- а) дерево квадратів; б) визначення центру квадратів; в) граф всіх можливих шляхів; г) мінімальний шлях.

Великого поширення набув метод Фронт хвилі (Wave Front) для навігації у структурованому середовищі, різні комбінації цього підходу розглядаються в [8]. Метод передбачає наступні підетапи планування траєкторії переміщення мобільного робота:

1). поділ структурованого середовища функціонування мобільного робота на окремі комірки, причому чим більше комірок буде представляти

одиницю площі, тим точніше буде представлено середовище, однак при цьому зростає складність обчислень (рис. 1.6 а)

2). Встановлення значень кожної комірки, що були отримані на підетапі планування 2 (рис. 1.6 б). Логічна одиниця комірки, що містить ціль переміщення, встановлюється в «0». Логічні одиниці комірок, що знаходяться по периметру біля комірки цілі встановлюються в «1». Логічні одиниці наступних комірок встановлюються за правилом:

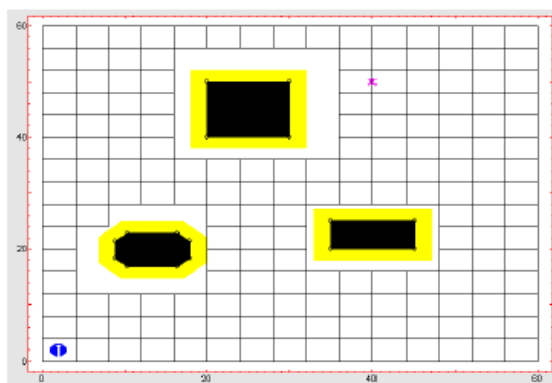
$$LV_{i+1} = LV_i + 1,$$

де LV_i - логічне значення попередньої комірки.

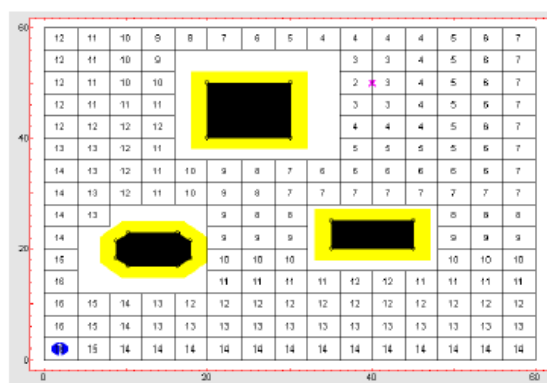
Значення комірок, що зайняті перешкодами, не враховуються. Якщо не можливо продовжити присвоєння значень комірок до комірки початкового положення робота, то на етапі планування буде прийнято рішення, що використовуючи відому інформацію про середовище функціонування неможливо побудувати траєкторії досягнення цілі роботом.

3). Наступний підетап використовується для побудови всіх можливих траєкторій руху від початкового положення мобільного робота до цілі. При цьому значення кожної наступної комірки повинно зменшуватись на «1» (рис. 1.6 в).

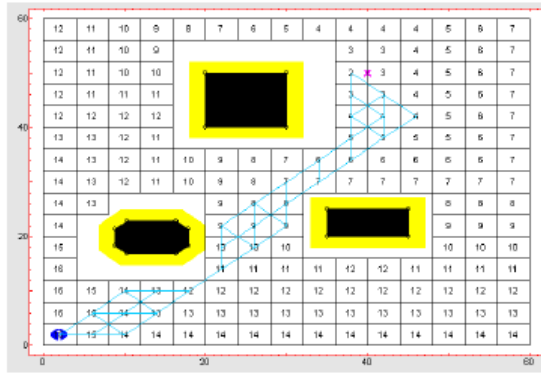
4). Кінцевий підетап використовується для знаходження мінімальної траєкторії переміщення (рис. 1.6 г.).



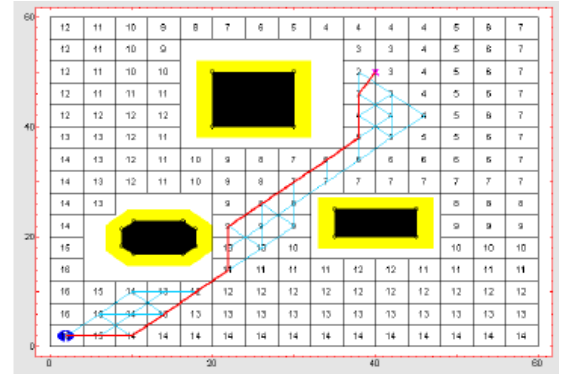
а)



б)



в)



г)

Рисунок 1.6 - Графічне представлення підетапів планування методу глобальної навігації Фронт Хвилі:

- а) поділ середовища на окремі комірки; б) встановлення значень комірок; в) всі можливі шляхи; г) мінімальний шлях

Якщо на етапі планування глобальних методів визначено оптимальну траєкторію переміщення до цілі, то наступний етап передбачає безпосереднє переміщення мобільного робота.

Проаналізувавши відомі методи глобальної навігації можна виділити ряд недоліків:

- 1) велика обчислювальна складність методів при великих картах середовища та при наявності великої кількості перешкод;
- 2) необхідність досить часто проводити процедуру локалізації, що зумовлено неточностями сенсорної та одометричної системи робота, але це є окрема задача, що в цілому приводить до ще більшої обчислювальної складності.

Тому слід використати більш прості з обчислювальної точки зору методи, як наприклад методи локальної навігації, що розглянуті в наступному параграфі.

На противагу методам глобальної навігації методи локальної навігації для переміщення до заданої цілі використовують інформацію отриману з допомогою сенсорів. В цьому випадку глобальна карта середовища є не доступною або ж середовище є достатньо неструктурованим чи в ньому

присутня велика кількість динамічних перешкод, при цьому застосування методів глобальної навігації стає неможливим.

Найбільш відомий метод локальної навігації, що базується на використанні інформації про середовище отриманої від сенсорів, є BUG. На даний час існує велика кількість його модифікацій PolarBUG [15], VisBUG [16], FuzzyBUG [17].

Основна суть методу полягає в тому, що напрямок руху робота визначається по градієнту, тобто вектору, що забезпечує мінімальний шлях слідування. Обхід перешкод здійснюється по периметру за годинниковою стрілкою чи проти неї, при цьому робот повертається на траєкторію, що визначена градієнтом на початку слідування до цілі.

Графічне представлення роботи методу локальної навігації BUG показано на рисунку 1.7.

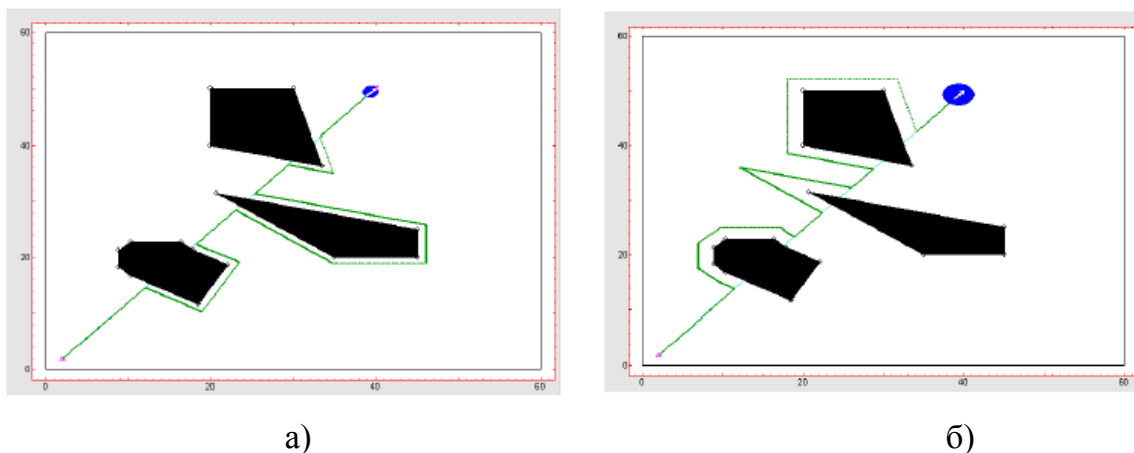


Рисунок 1.7 - Графічне представлення методу локальної навігації BUG
а) обхід перешкоди проти годинникової стрілки; б) обхід перешкоди за годинниковою стрілкою.

Метод VisBUG передбачає використання інформації від сенсорів, що забезпечують визначення перешкод на певній відстані від мобільного робота. Метод BUG при цьому використовує тактильні сенсори, коли VisBUG для побудови локальної карти середовища використовує ультразвукові, інфрачервоні та лазерні сенсори. Наприклад, ультразвукові сенсори

мобільного робота AmigoBot [70] спроможні визначити перешкоду на відстані 5 метрів. В такому випадку робот оперує локальною картою середовища радіусом 5 метрів. Приклад роботи методу VisBUG представлений на рисунку 1.8.

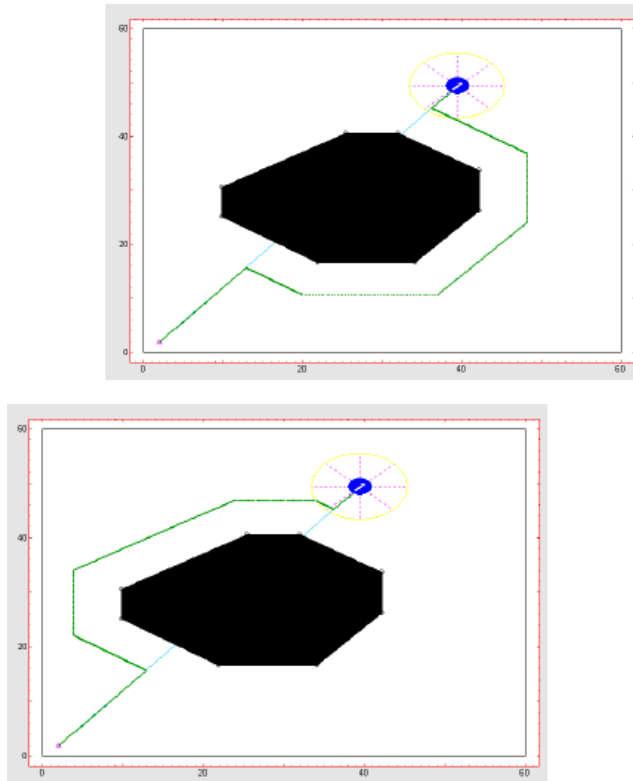


Рисунок 1.8 - Графічне представлення роботи методу локальної навігації VisBug:

- а) обхід перешкоди проти годинникової стрілки; б) обхід перешкоди за годинниковою стрілкою

Ще одним підходом до побудови методів локальної навігації є використання потенційних полів (Potential Fields) об'єктів в середовищі [28]. Ціль переміщення використовує власне потенційне поле, що забезпечує рух робота до цілі на кожному кроці переміщення, оскільки наближення до цілі забезпечує збільшення потенційного поля цілі. Значення потенційних полів перешкод збільшуються при наближенні до перешкоди, що приводить до «відштовхування» робота від перешкоди. На кожному кроці руху

визначається вплив потенційних полів цілі та перешкод на рух робота.

Пошук максимуму вказує на наступну траєкторію руху мобільного робота:

$$\max (K_{\text{пр.}} - \sum K_{\text{від.}}),$$

де $K_{\text{пр.}}$ - коефіцієнт притягання до цілі, $K_{\text{від.}}$ - коефіцієнт відштовхування від перешкод.

Графічне представлення роботи методу представлено на рисунку 1.9.

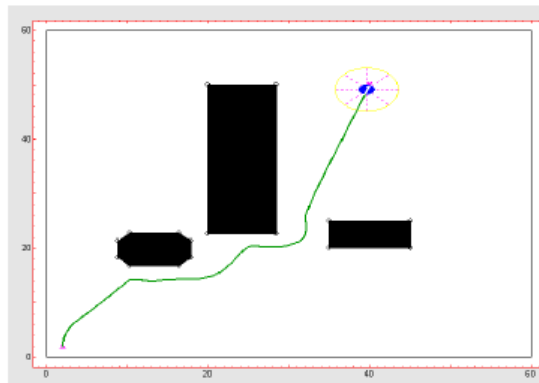


Рисунок 1.9 - Графічне представлення роботи методу локальної навігації
Потенційних полів

Провівши аналіз методів навігації можна виділити наступні недоліки їхнього використання:

- більша проблема локалізації робота у порівнянні з методами глобальної навігації;
- відхилення від оптимального маршруту;
- досягнення локального мінімуму (блокуючі перешкоди);
- зациклення (кружляння по одній і тій же траєкторії) при спробі вийти із локального мінімуму.

1.3 Узагальнена структура системи управління

Аналіз переваг та недоліків згаданих вище засобів та способів управління показав, що функції робота, умовно можна поділити по рівнях комплексності. До основних функцій мобільного робота відносять: виконання рухів в робочій зоні оточення, виявлення об'єктів оточення та їх лінійних параметрів, планування шляху та траєкторій руху робота, керування засобами, що знаходяться на платформі мобільного робота (маніпуляторами), тощо. Одночасно вирішуються навігаційні задачі: локалізації, досягнення кінцевої точки руху, уникнення перешкод та ін. Виконання вище наведених дій в автоматичному режимі забезпечують автономність мобільного робота [7,29,30].

Під автономним мобільним роботом слід вважати систему керування і технічну систему, що здійснює цілеспрямований рух в оточенні для досягнення поставленої цілі[30]. До технічних засобів АМР включають силові пристрої, що підтримують фактичне переміщення АМР в оточенні і системи давачів, які можуть бачити перешкоди, які містяться в напрямку цілі об'єкту мобільного робота. Можна вважати, що АМР рухається в стаціонарному оточенні, в якому розташування основних перешкод-об'єктів є наперед відомим, а також з динамічними перешкодами, про які АМР дізнається лише після його.

Глобальна ціль переміщення АМР - направлений рух робота в неструктурованому середовищі, оминаючи перешкоди для досягнення цільового пункту переміщення найоптимальнішим методом.

При побудові системи управління АМР виникає багато технічних нюансів, зокрема, можна виділити наступні три:

1. Щоб переміщуватись до цілі, АМР повинен побудувати досить точну структуру середовища, що його оточує.

модулів із використанням інтелектуальних засобів (нейронних мереж, еволюційних алгоритмів тощо). Така ситуація надає можливість швидко адаптуватись до змін в оточенні і забезпечувати досягнення поставленої цілі.

На основі проведеного аналізу відомих систем управління мобільними роботами [30,32,33], узагальнену модель забезпечення функціонування АРМ і забезпечення його переміщення можна представити у вигляді сукупності наступних функціональних блоків, що представлено на рисунку 1.10.

У блоці оброблення сенсорних даних робота (2) здійснюється перетворення фізичних показів давачів в форму, що придатна для подальшого оброблення та аналізу, також в цьому блоці застосовуються методи усунення шумів та підвищення достовірності отриманої інформації. Блок побудови локальної карти середовища (3) на основі отриманої інформації та бази знань, що визначає тип оточення функціонування АРМ, формує локальну карту. Блок локалізації (4) призначений для коригування місцезнаходження мобільного робота під час переміщення, оскільки покази системи управління та реальне місце робота в середовищі з часом може змінюватись, що зумовлено похибками у одометричній системі, а також проковзуванням коліс. Блок навігації (5) застосовується у прийнятті рішення про напрямок переміщення до цілі на стратегічному етапі, на тактичному етапі блок маневрів (7), використовуючи дані про середовище, що оточує робота, безпосередньо передає управляючі дані до апаратних засобів руху (1). У базі знань (8) встановлені правила поведінки системи управління, а також кожного блоку окремо, на складних ділянках руху та при переміщенні в неструктурованому динамічному оточенні. Блок побудови глобальної карти середовища (6) використовується при плануванні руху в невідомій місцевості. При цьому використовуються методи, що будують глобальну карту використовуючи локальну карту середовища на кожному етапі руху.

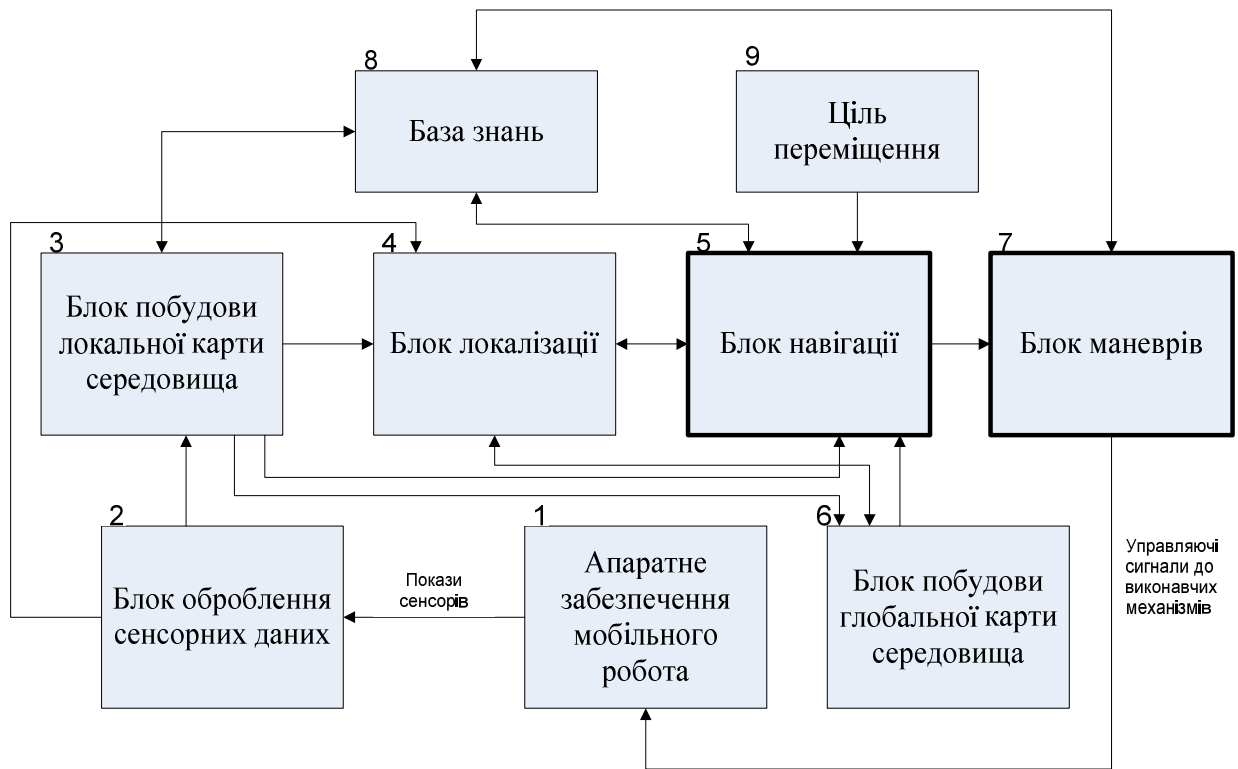


Рисунок 1.10 - Узагальнена структура управління АРМ

Згідно наведеної структури управління автономним мобільним роботом, задача планування руху в місцевості із стаціонарними та динамічними об'єктами вимагає прийняття швидких дій, при цьому слід враховувати конструктивні особливості робота, поточне розміщення, а також особливості оточуючої місцевості. Це завдання вирішується з допомогою застосування навігаційних засобів, що розглядалися вище. Для переміщення в середовищі без наземних чи супутникових радіонавігаційних систем мобільний робот використовує різні типи давачів [26], відеокамери чи стереопари, інфрачервоні, ультразвукові і/або лазерні дальноміри тощо. В цій ситуації при переміщенні робот детектує різні моделі середовища та накопичує знання про нього, планує оптимальні маневри та рухається з допомогою підсистеми маневрів, що формує керуючі сигнали на двигуни.

2 АЛГОРИТМІЧНЕ ТА ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ СИСТЕМИ УПРАВЛІННЯ МОБІЛЬНИМ РОБОТОМ

2.1 Алгоритм планування рухом мобільного робота для неструктурованого середовища

Методи глобальної навігації, що розглядались у першому розділі, мають ряд обмежень, та передусім основним недоліком стає не можливість проводити навігацію у випадку незначних змін середовища, чи у середовищі, що є неструктурованим з наявністю динамічних перешкод. Основною перевагою локальних методів навігації є простота їх реалізації, однак вони також не забезпечують вихід на ціль при існуванні блокуючих перешкод, що утворюють локальні мінімуми. Таким чином актуальною є задача розроблення методу навігації в складному неструктурованому середовищі з можливістю функціонування АМР при появі динамічних перешкод. Тому запропоновано та розроблено регулярний ітеративний метод градієнтного пошуку з локальною адаптивною оцінкою обмежень другого роду, для визначення шляху переміщення АМР та метод побудови траєкторії руху АМР для підсистеми маневрів, які дозволяють досягнути ціль в складному неструктурованому середовищі [37].

Суть запропонованого градієнтного методу навігації полягає в тому, що переміщення здійснюється з використанням потенційних полів об'єктів в середовищі функціонування АМР, при цьому використовується обчислення значення функції вартості. Значення функції вартості по відношенню до перешкод, а також по відношенню до заданої цілі переміщення. Навігація до цілі відбувається по градієнту до цілі – характеристика, що показує напрямок найшвидшого зростання/спадання деякої величини, значення якої змінюється від однієї точки простору до іншої. В разі появи блокуючих перешкод (попадання в локальний мінімум) переміщення здійснюється адаптацією до

встановлюється потенційне поле та функція вартості відстані до перешкоди. З наближенням до перешкоди значення функції зростає. Також, потрібно встановити критичне значення мінімальної відстані до перешкоди $obst_dist_critical$, що визначається технічними специфікаціями робота, та відстані non_dang_dist , на якій перешкода не впливає на рух АМР до цілі. Такий підхід дозволить АМР безпечно рухатись до цілі, не зіштовхуючись з перешкодами, а також мати достатній простір для проведення маневрів.

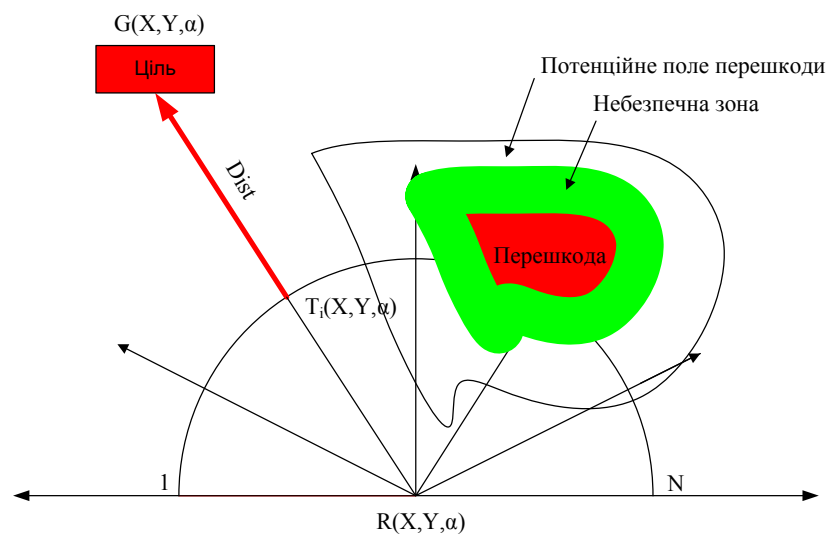


Рисунок 2.2 - Представлення перешкоди на локальній карті середовища АМР

Формалізацію обчислення функції вартості перешкоди представлено у 2.2.

$$g(obst_dist) = \begin{cases} 1000, & \text{якщо } obst_dist < obst_dist_critical \\ k * obst_dist + b, & \text{якщо } obst_dist_critical \leq obst_dist \leq non_dang_dist \\ 0, & \text{якщо } dist_obst > non_dang_dist, \end{cases} \quad (2.2)$$

де k і b – коефіцієнти рівняння прямої, що визначаються технічними характеристиками АМР.

Якщо АМР попадає у області дії потенційних полів перешкоди чи декількох перешкод, то напрямок руху вибирається по максимальному значенню функції вартості (рис.2.3).

Для визначення напрямку переміщення на кожному кроці при впливі на рух АМР двох і більше перешкод знаходимо сумарне значення функції вартості:

$$y_{Sum_{T_i}} = f(dist_{T_i})^2 - \sum_{j=1}^M g(obst_dist_m^{T_i})^2, \quad (2.3)$$

де M – загальна кількість перешкод, що впливають свої потенційним полем на альтернативну точку наступного переміщення T_i .

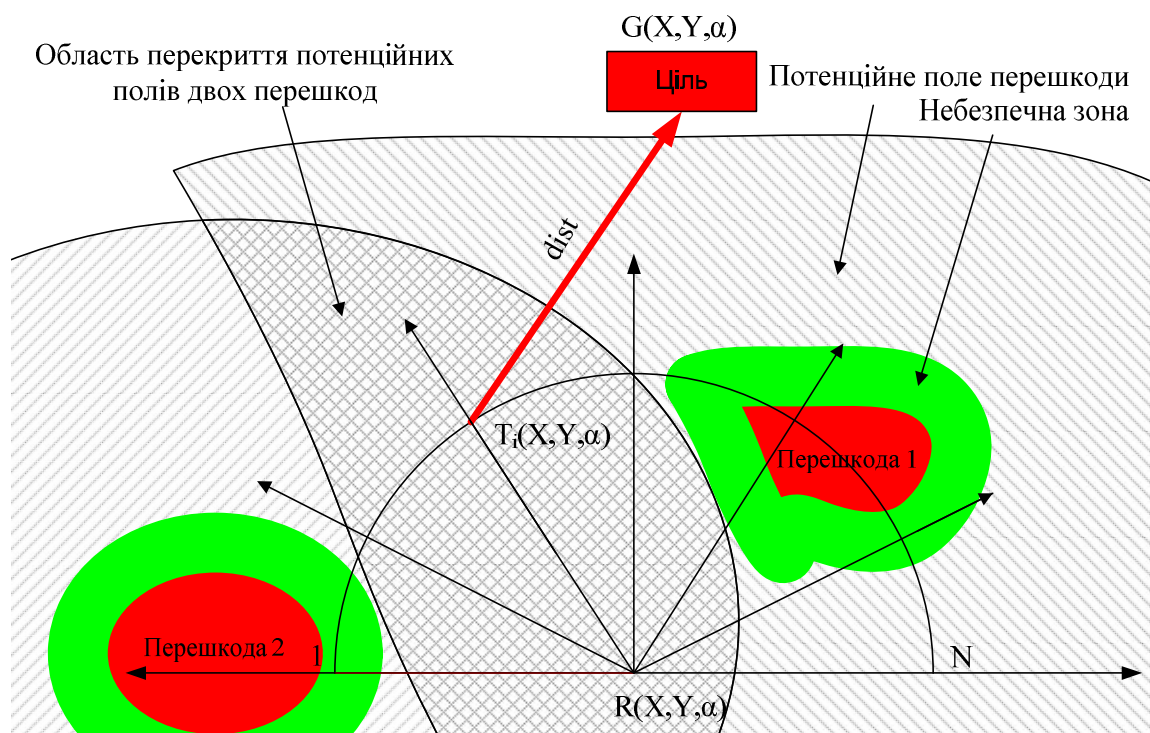


Рисунок 2.3 - Графічне представлення впливу потенційних полів двох перешкод

Напрямок руху АМР вибирається за максимальним значенням сумарної функції вартості з усіх альтернативних точок переміщення:

$$\max(y_{Sum_{T_i}}). \quad (2.4)$$

Змн.	Арк.	№ докум.	Підпис	Дат

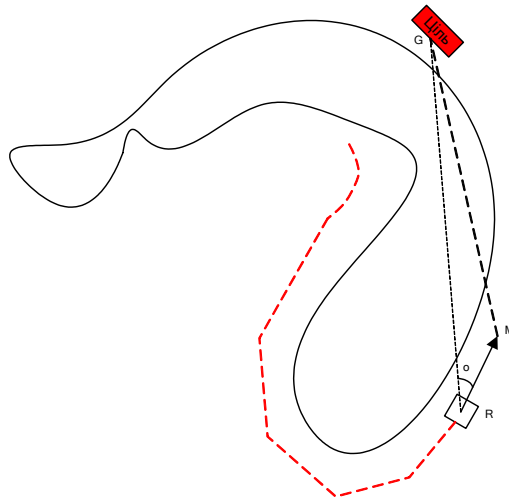


Рисунок 2.6 - Обхід перешкоди по периметру (умова (2.5) виходу з етапу не виконується)

Можлива ситуація, коли АМР може потрапити в попередні позиції (рис.2.7), при цьому доцільно зберігати координати попереднього етапу обходу перешкоди. При цьому, якщо наступний крок переміщення попадає в позицію, через яку робот проходив на попередньому етапі обходу перешкод $P \in M$, тоді напрямок обходу перешкод змінюється на протилежний, що забезпечує вихід на ціль.

У випадку коли АМР, змінивши напрямок руху обходу перешкод у точці P_1 потрапляє у точку P_2 , що належить траєкторії попереднього обходу, переміщення припиняється оскільки MP попадає у закрите середовище, з якого немає виходу (рис.2.8).

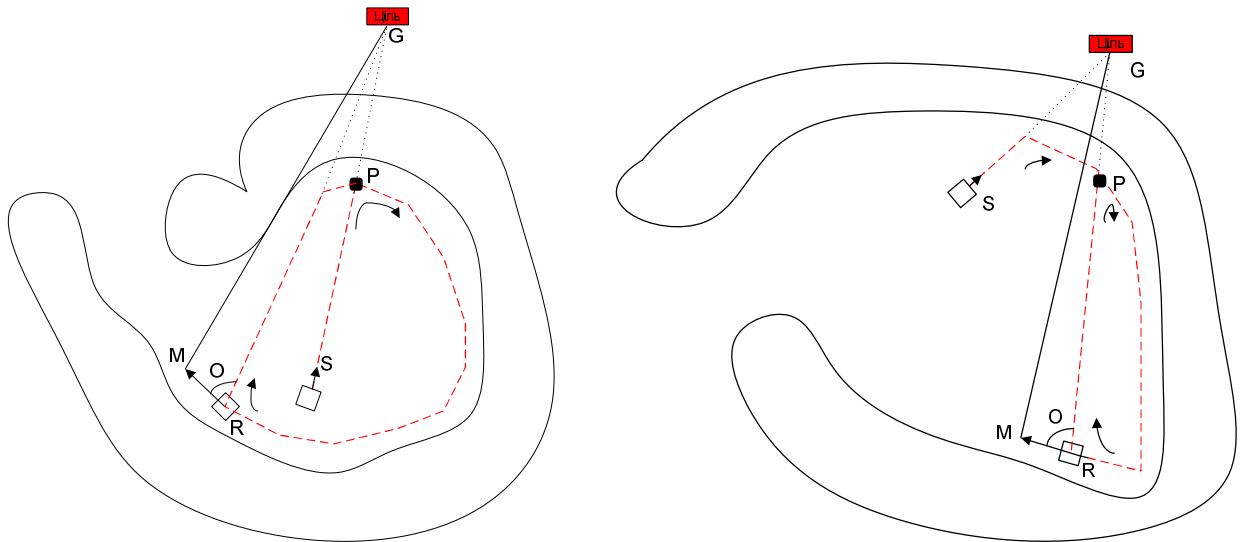


Рисунок 2.7 - Повторне попадання на траєкторію руху

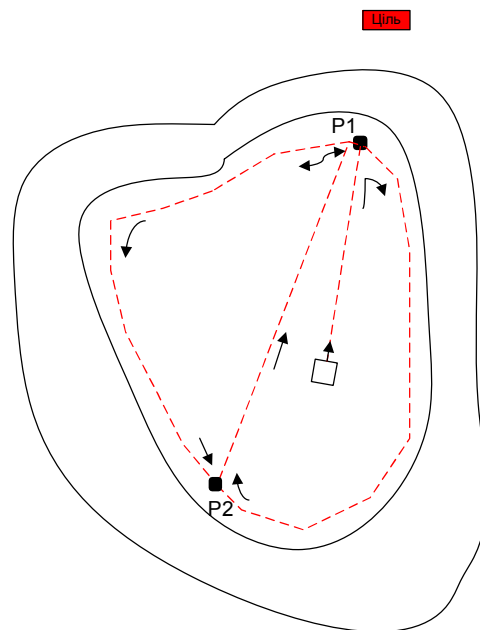


Рисунок 2.8 - Тупикова ситуація при навігації АМР

Таким чином реалізація методу навігації складається з двох основних етапів:

- слідування до цілі, використовуючи градієнт напрямку до цілі, та значення функції вартості до перешкод;
- обхід перешкоди по периметру.

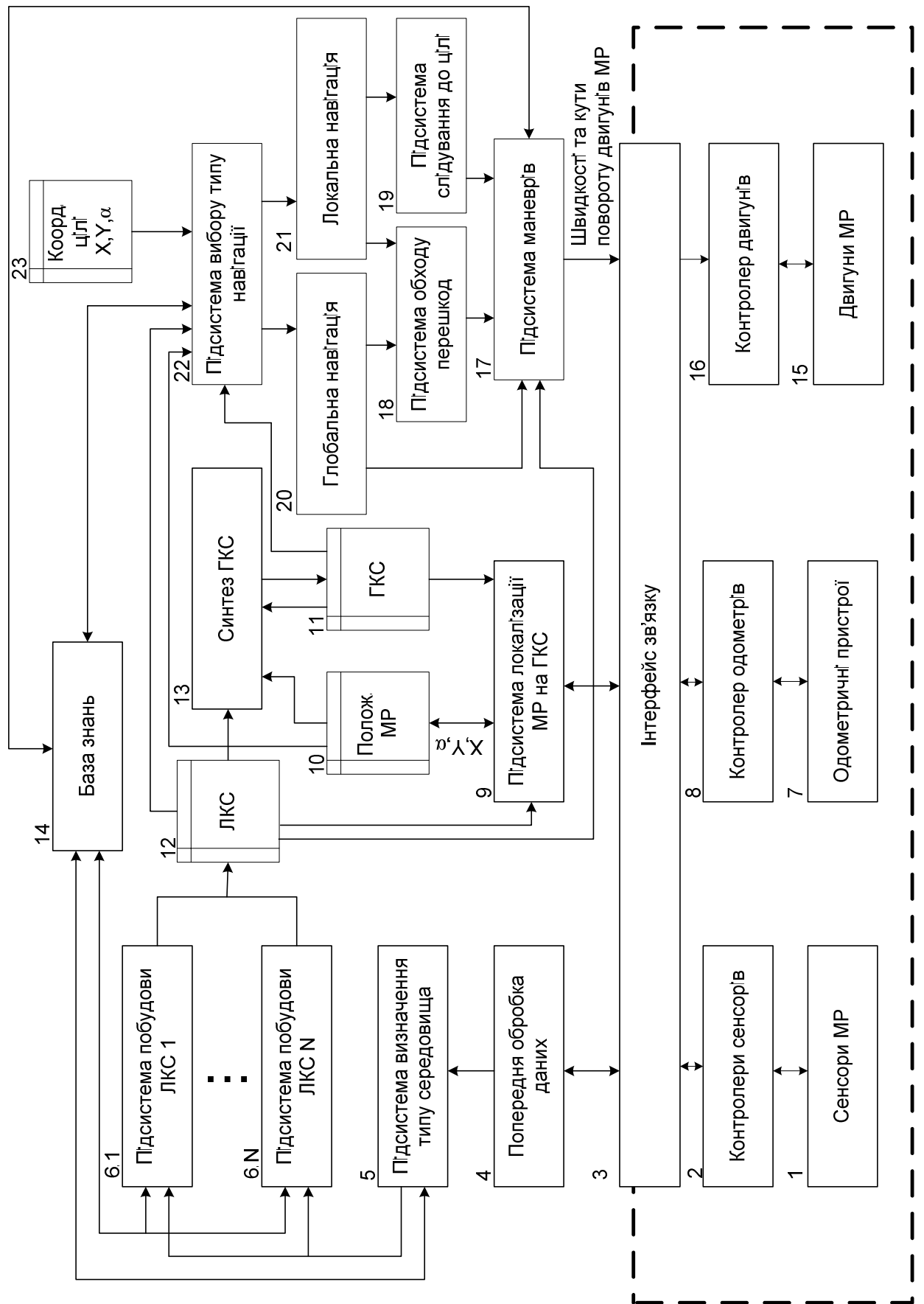


Рисунок 2.9 - Модель системи управління мобільним роботом

Змн.	Арк.	№ докум.	Підпис	Дат

інформації в БД для можливості переміщення робота у попередні положення у випадку тупикових ситуацій, коли потрібно провести рух робота на позицію, яку він проходив на попередніх етапах руху, і вибору іншого напрямку руху.

Модуль візуалізації підтримує моніторинг функціональної активності робота на екрані монітора. Такий режим є потрібним під час відлагоджування основних завдань функціонування мобільного робота. В реальних ситуаціях модуль моніторингу не застосовується для збільшення загальної продуктивності обчислювальних засобів.

Модулі відтворення ЛКС 1..N, побудови ГКС, локалізації та навігації забезпечують функціонування алгоритмів відповідно до етапу загального алгоритму управління роботом. Для функціонування таких алгоритмів диспетчер виділяє процесорний час, який необхідний для їх роботи. Функціонування кожного з алгоритмів закінчується викликом переривання процесу і повернення до диспетчера із інформацією про закінчення процедури виконання алгоритму. На основі переривань диспетчер встановлює послідовність виконання завдань для функціонування мобільного робота, що формується за допомогою нових переривань.

Отже на основі аналізу структурних рішень, було встановлено основні вимоги до системи управління АМР в комплексних (динамічних) середовищах, що дало змогу сформулювати деталізовану модель системи управління, та визначити основні потоки інформації та взаємозв'язок функціональних модулів на програмному рівні.

2.3 Програмне забезпечення клієнт-серверної взаємодії

Згідно запропонованої в підрозділі 2.2 моделі управління АМР всі функціональні блоки можна розділити на апаратно-залежні та програмно-залежні (рис. 2.11). Головними завданнями апаратно залежних блоків є

приймання сигналів, перетворення в керуючі сигнали та отримання даних від давачів. Алгоритми функціонування цих блоків є чітко визначеними на етапі виготовлення і не піддаються зміні. На програмному рівні встановлюються задачі відтворення структурисередовища, локалізація та навігації автономного мобільного робота, що піддаються вдосконаленню та модифікації.

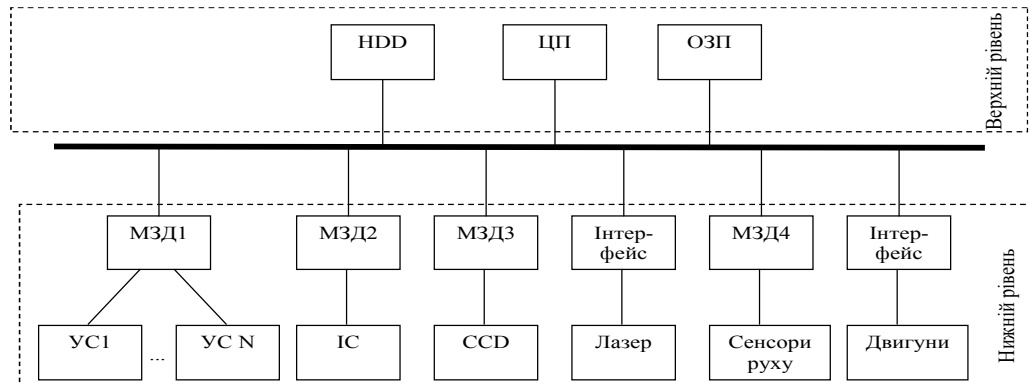


Рисунок 2.11 - Структура АРМ

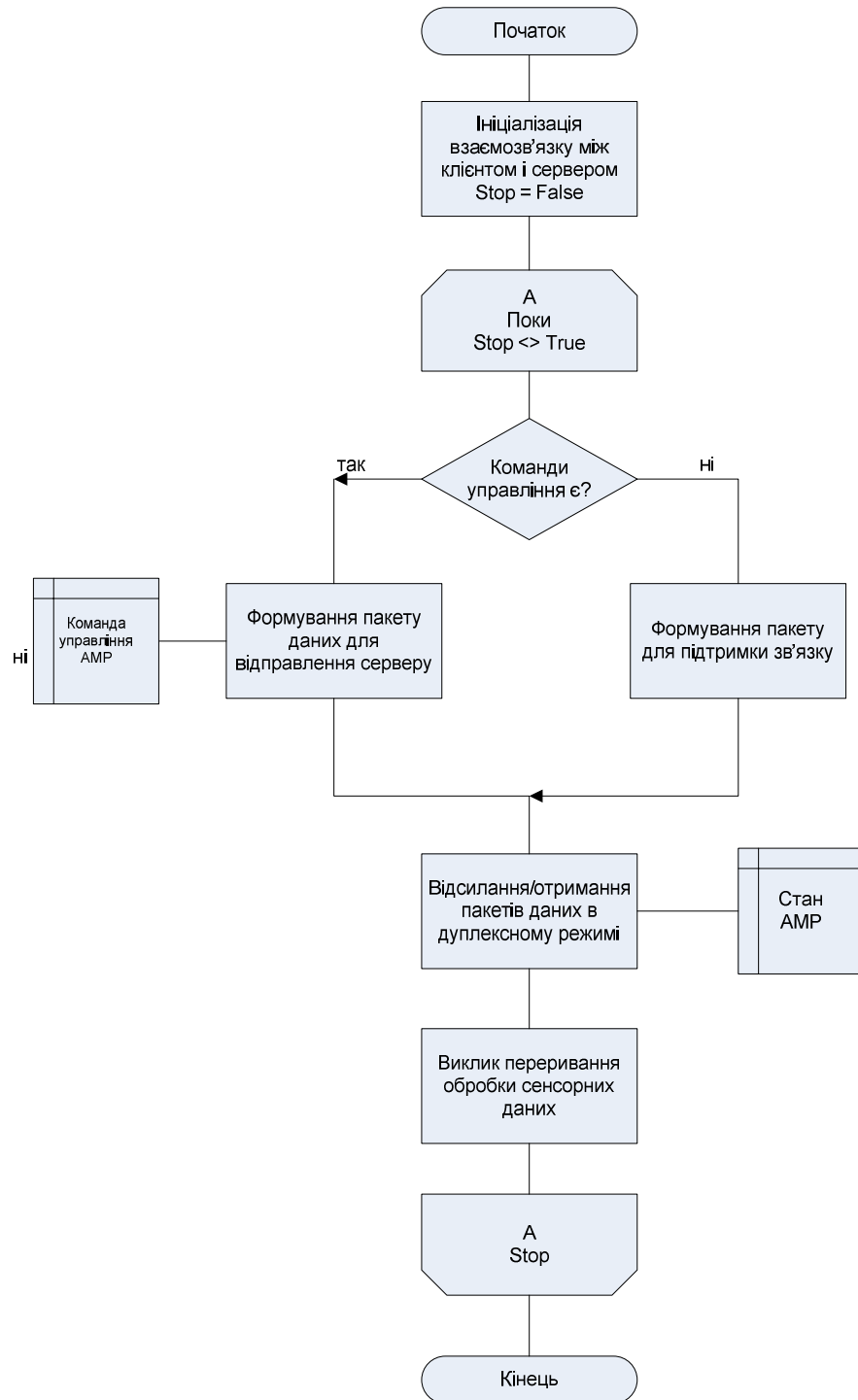
Взаємодію між програмно-залежними та апаратно-залежними функціональними блоками мобільного робота імплементовано за архітектурою клієнт-сервер [49,143]. Вона визначає функціональні блоки

- клієнта (клієнт запитує у сервера інформацію) та
- сервера (сервер відповідає на запити клієнтів).

У загальному випадку кожен клієнт може обмінюватись інформацією з багатьма серверами, а сервер – з багатьма клієнтами паралельно.

Для реалізації цього підходу використана технологія СОМ [152]. Технологія СОМ (Component Object Model) – це об’єктно-орієнтована технологія, яка реалізує принцип компонентного програмного забезпечення (ПЗ) (рис 3.7). Сервер - це частина коду (модуль, підпрограма), який складається з компонентів, що формують об’єкти, і забезпечує клієнтів доступом до цих компонент. Клієнт – це фрагмент коду, якому надається доступ до функцій і даних об’єктів. Клієнт зв’язується з об’єктом як його інтерфейс. Як і кожна об’єктно-орієнтований підхід, СОМ працює з поліморфізмом, інкапсуляцією і наслідування. Одна з основних переваг

та задачами верхнього рівня, що реалізуються на ПК. Розроблено алгоритм зв'язку (рис.2.12), що дозволяє сформувати мінімізацію часових затрат на оброблення пакетів між сервером та клієнтом при умові відсутності підтвердження отриманих сигналів [94]. Це в цілому дозволяє збільшити продуктивність елементів мобільного робота на 10-15 відсотків.



Змн.	Арк.	№ докум.	Підпис	Дат

Рисунок 2.12 - Алгоритм зв'язку між серверним та клієнтським блоками мобільного робота

Всі інші компоненти формують програмно-залежну частину і розробляються на мові програмування. Нижній рівень дворівневої структури мобільного робота (рис. 2.10) формують активатори та сенсори (ІС, УС, двигунів, камер, сенсорів руху, лазера), модулі збору даних (МЗД1-МЗД4) і контроллерів активаторів, а також з ОЗП, накопичувача на жорсткому диску (HDD) та центрального процесора на верхньому рівні. Задачами модулів збору даних (МЗД) є формування зміни сигналу давачів у цифровий код. При цьому алгоритм зміни є незмінним і жорстко записується в ПЗП. МЗД в цілому управляє процесом збору інформації та зв'язком з шиною даних на верхньому рівні. МЗД є апаратно-орієнтованим на отримання інформації від конкретних типів давачів і є найменш загальним серед усіх рівнів мобільного робота. Головною вимогою до обчислювального пристрою МЗД є його робота в реальному часі збору інформації із частотою поступлення показів давачів. Такими ж функціями відповідають контроллери засобів виконання, які формують інтерфейс з шиною даних і апаратно орієнтовані на конкретний тип засобу виконання.

На верхньому рівні формується блок центрального процесора (ЦП), який встановлює функціонування всіх компонентів мобільного робота і виконує найбільш вагомі процедури відтворення ЛКС, локалізації, навігації, та управління іншими компонентами мобільного робота. Спеціалізація завдань процесора формується спеціалізованим ПЗ. При цьому коди програмних модулів алгоритмів імплементації методів керування АМР та результати оброблення збору сенсорної інформації і функціонування програм записуються в оперативній пам'яті (ОЗП). Інформація, що не потребує постійного оброблення, зберігається на жорсткому диску (HDD), наприклад, положення АМР і ГКС може зчитуватись із жорсткого диску на початковому етапі переміщення мобільного робота.

мобільного робота. Також, враховуючи те, що операційні системи Windows, UNIX, LINUX є універсальними, які застосовують багато часу центрального процесора для власних потреб, доцільно верхні рівні архітектури програмного забезпечення реалізовувати на мовах високого рівня на цих системах, тоді як підсистеми комунікації із зовнішніми засобами та давачами – на контролерах.

					ДП.КСМ. 07254/16.00.00.000 ПЗ	55
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дат</i>		

функціональні компоненти формують засоби для вбудовування операцій у тип.

Об'єктно-орієнтоване проектування програм - це збільшення застосування абстракції даних. Абстрактні типи даних, чи об'єктні типи, не тільки не показують структуру інформації, але інкапсулюють функціонування як операції над об'єктами. Суть об'єктно-орієнтованого проектування полягає в пошуку найбільш прийнятних об'єктних типів.

Також виявляється інший факт об'єктно-орієнтованого проектування - застосування успадкування для побудови спеціалізованих варіацій базових об'єктних типів. Окрім розроблення зовсім іншого об'єктного типу для кожної програми можна сформувати нові типи з загальних базових типів

Нове в ООП - це метод розгляду програм як реалізацій об'єктних типів, а не як реалізацій алгоритмів. Це концептуальне вдосконалення іноді важко оцінити програмісту. Абстрактні концепції і сутності, що не є реальними об'єктами, може бути особливо складно віднайти як кандидатів в об'єктні типи.

Багато програмних модулів ніколи не забезпечують прямо з інтуїтивно зрозумілими формуванням фізичних об'єктів. Необхідні в цих ситуаціях об'єктні типи є частиною абстрактної проблемної сфери. В операційній системі процедура може бути об'єктом, навіть якщо "process" вважається як діяльністю. Загальне ядро операційної системи може формувати процес дистрибутивними структурами інформації і функціями, що маніпулюють різними структурами. В об'єктно-орієнтованому проектуванні ці компоненти інформації і функції вкладаються в клас для формування об'єктного типу "процес". Інтерфейс об'єктного типу - це багато функцій, за допомогою яких ядро управляє процесами.

Наступний аспект об'єктно-орієнтованого проектування визначається у встановленні родового типу. Так, у програмному забезпеченні оброблення списків можуть бути присутні "списки рядків", "списки цілих" і "списки

записів про працюючий". Усі ці списки з іншими типами компонентів можуть бути сформовані як сутності родового типу список. Ключ до розуміння родового типу полягає в тому, що це сховище інших типів. Ймовірно не варто формувати родовий тип, якщо тільки розробник не застосує більш одного приклада типу, чи тип не буде ще раз застосовуватися в інших програмних кодах.

Для розробника, що застосовує для декомпозиції проектування в термінах процедур, потрібно мати деяку практика при застосуванні до проектування в об'єктному стилі. Якщо розробник вступив в сферу об'єктно-орієнтованого проектування, він повинний бути прискіпливий, щоб не перестаратися. Багато свформованих об'єктних типів зменшують цінність розробки ПЗ. Хороше об'єктно-орієнтоване програмування чітко відповідає заданню, що повинен вирішувати програмний модуль, зменшуючи розуміння, впровадження і підтримку програми.

В ООП компоненти задачі відповідають елементам розробки, що є об'єктними типами, що забезпечують формування програмного модуля. Одне із досягнень цього методу розробки програм полягає в тому, що є концептуальну єдність усіх фаз формування програми, і компоненті модулі об'єктного типу, що сформувалися, легко ще раз застосовувати[30].

Концептуальна структура програмного забезпечення формується не тільки від опису задань і до імплементації, але і при удосконалюванні реалізації. Після визначення об'єктних типів розробки формується додаванням деталей цих типів. Це відмінно від покрокового уточнення функції зверху вниз, при якому задача більш високого рівня трансформується в іншу структуру. Маючи цілу структуру проекту до того, як відомі всі компоненти, ми отримуємо можливість швидкого прототипування. Прототипні формування об'єктних типів можуть бути сформовані зміщенням малого числа операцій і приховуванням незначних нюансів, що дозволяє перевірити доцільність проекту на ранніх етапах. Можливість швидкого

формування прототипів зменшує витрати на розробку методом проб і спроб.

Оскільки об'єктно-орієнтоване проектування не є проектуванням зверху вниз, сформовані програмні компоненти більш незалежні і, тому, їх краще ще раз застосовувати. Модуль не імплементується в ієрархію проекту конкретної програми, і тому, не залежить від конкретної структури програмного забезпечення. Модуль об'єктного типу є імплементацією елемента конкретної проблемної сфери, замість того, щоб служити підфункцією одного вирішення завдання. У цьому зв'язку модуль може застосовуватися в будь-якій частині програмі в тієї ж чи схожій проблемної сфери.

Наступний аспект повторного застосування модуля забезпечується в легкості збільшення об'єктних типів спадкуванням. Вихідний модуль стає недоторканим, тоді як нове функціонування добавляється в похідний тип. Модулі, що є загальними об'єктними типами, можуть бути легко повторно застосовані для різних типів завдання.

ООП має зв'язок із стилем розробника, що формується в перетворення об'єктів під час реалізації програми і динамічному зв'язуванні операцій над об'єктами. Цей стиль розробки переходить до інтерпретуючого об'єктно-орієнтованим мовам і системам типу Flavors і Smalltalk. Ці системи були імплементовані для гнучкості під час реалізації і показали себе як діючі інструменти прототипування, безпосередньої реалізації вирішення завдань і імітаційного моделювання.

Можна розробити об'єктно-орієнтовані програмні модулі на C++, що формуються цілком із розробки зразків об'єктів під час реалізації і динамічне зв'язування операцій над об'єктами з віртуальними функціями і процедурами. Формування об'єктів і виклик об'єктних операцій впроваджується в тому місці тексту програмного модуля, що транслюється. Хоча виклики віртуальних функцій і процедур динамічно пов'язані, вони повинні пройти статичну перевірку типів. Те, чого не вистачає C++ у змісті інтерактивної

гнучкості, викликано або великою перевіркою типів на етапі трансляції, або сформовано для ефективного виконання про програмних модулів.

Динамічне ООП багато застосовується для моделювання. Для моделювання різних сутностей у задачі формуються об'єктні типи. Еталони об'єктів у цьому випадку застосовуються в ролі моделей. Об'єкти комунікують, при цьому викликаються операції один над іншим і створюючи нові об'єкти, що формуються моделюванням.

З проведеного аналізу можна сказати, що для реалізації підсистеми управління автономним мобільним роботом варто використати технологію об'єктно-орієнтованого програмування, що забезпечить гнучкий підхід до імплементації запропонованої архітектури системи управління АМР. Серед системи розробки програмного середовища виділимо засоби мови програмування C++, Matlab що забезпечить сформувати ОО підхід, а також збільшить швидкодію процесу взаємозв'язку компонентів системи управлінн роботом за рахунок вбудованих функцій прямого звертання як до пам'яті так і до накопичувача на жорсткому диску.

3.2 Імітаційне моделювання програмного забезпечення управління

З метою дослідження розробленого методу навігації, що представлений в §2.1 використано програмний симулятор мобільних роботів компанії ActivMedia MobileSim, що дозволяє проводити дослідження методів навігації з різними базами мобільних роботів Pioneer 1, Pioneer AT, Pioneer 2™ -DX, -DXe, -DXf, -CE, -AT, Pioneer 2™-DX8/Dx8 Plus і -AT8/AT8 Plus, а також новими розробками Pioneer 3-DX і –AT **Ошибка! Источник ссылки не найден.Ошибка! Источник ссылки не найден..**

Для побудови карти середовища використано також продукт компанії ActivMedia Mapper3, що дозволяє побудувати статичне середовище, з

визначенням початкового положення робота та координатів цілі. Зображення побудованого середовища, для дослідження та моделювання як відомих так і запропонованого методу навігації представлено на рисунку 3.1.

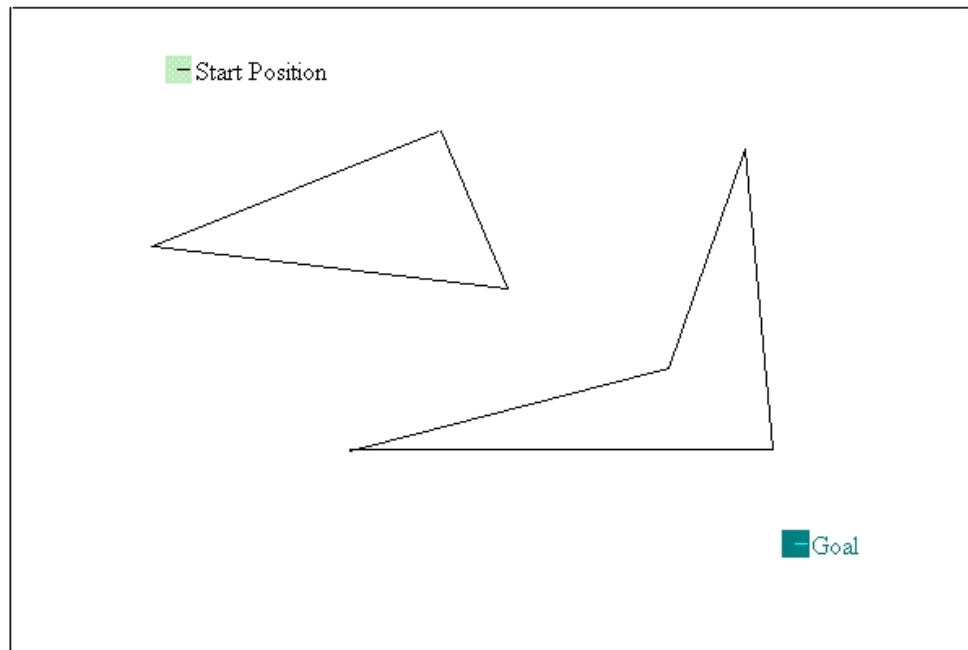
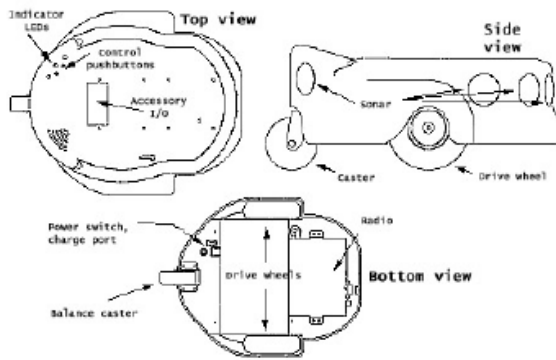


Рисунок 3.1 - Карта середовища мобільного робота сформована за допомогою Mapper3.

Результати досліджень та моделювання запропонованого методу навігації представлені на рисунку 3.2.



a)



б)

Рисунок 3.3 - Мобільний робот AmigoBot:

а) схематичне представлення робота; б) навігація мобільного робота в середовищі

Для комунікації з ПК застосовується стандартний інтерфейс вводу-виводу на основі RS232.

Під час експериментальних досліджень було використано протокол комунікації між базою АМР та персональним комп'ютером. Зв'язок формувався на основі клієнт-серверної архітектури. Такий підхід до взаємодії зумовлений потребою розподілу обчислювальних потужностей між завданнями низького рівня, наприклад, керування сенсорами, двигунами, формування даних від одометричних пристроїв та сенсорів, які можуть розроблятися на спеціалізованих засобах, і завданнями верхнього рівня, що розробляються на персональному комп'ютері на рівні програмного забезпечення. Комунікація між клієнтською і серверною частиною формується спеціалізованим протоколом обміну інформацією, що має встановлену структуру, що представлено в таблиці 3.1.

також можливістю зміни Matlab модулів для вирішення нових технічних завдань за рахунок впровадження нових toolboxes, відкритою архітектурою, яка надає користувачеві формувати повний доступ до лістингів ПЗ пакету за рахунок потужної мови програмування, можливістю трансляцією коду процедур та функцій, розроблених в середовищі Matlab на мови більш низького рівня, такі як C, C++ з метою збільшення продуктивності обробки інформації.

					ДП.КСМ. 07254/16.00.00.000 ПЗ	
Змн.	Арк.	№ докум.	Підпис	Дат		67

ВИСНОВКИ

З метою дослідження розробленого методу навігації використано програмний симулятор мобільних роботів компанії ActivMedia MobileSim, що дозволяє проводити дослідження методів навігації з різними базами мобільних роботів Pioneer 1, Pioneer AT, Pioneer 2™ -DX, -DXe, -DXf, -CE, -AT, Pioneer 2™-DX8/Dx8 Plus і -AT8/AT8 Plus, а також новими розробками Pioneer 3-DX і –AT **Ошибка! Источник ссылки не найден.Ошибка! Источник ссылки не найден..**

Для побудови карти середовища використано також продукт компанії ActivMedia Mapper3, що дозволяє побудувати статичне середовище, з визначенням початкового положення робота та координатів цілі

Для здійснення управління мобільним роботом реалізовано програмне забезпечення отримання інформації від одометричних двигунів АМР та проведення співставлення реального місця знаходження робота та інформації від системи управління автономним мобільним роботом АМР. Для програмної реалізації і експериментальних досліджень методу управління в якості програмного засобу використано пакет Matlab. Обрання системи Matlab зумовлено її перевагами над іншими інженерними прикладними пакетами програм, а також можливістю зміни Matlab модулів для вирішення нових технічних завдань за рахунок впровадження нових toolboxes, відкритою архітектурою, яка надає користувачеві формувати повний доступ до лістингів ПЗ пакету за рахунок потужної мови програмування, можливістю трансляцією коду процедур та функцій, розроблених в середовищі Matlab на мови більш низького рівня, такі як C, C++ з метою збільшення продуктивності обробки інформації.

СПИСОК ВИКОРСИТАНИХ ДЖЕРЕЛ

1. UVS International = Міжнародної асоціації безпілотних систем [Електронний ресурс]. – Режим доступу: http://uvs-international.org/index.php?option=com_content&view=article&id=204&Itemid=284
2. Інерціально-супутникові навігаційні системи / М.К. Філяшкін, В.О. Рогожин, А.В. Скрипець, Т.І. Лукінова. – К : НАУ, 2009. – 296 с.
3. Долінце Б.І. Методи зменшення похибок інтегрованої інерціально-супутникової навігаційної системи / Б.І. Долінце // Проблеми навігації і управління рухом: тези доповідей Всеукраїнської науково-практичної конференції молодих учених і студентів, 28-29 листопада, 2012. – К. : НАУ, 2012. – С. 45.
4. Low Cost INS/GPS Integration : Concepts and Testing / O.S. Salychev, V.V. Voronov, M.E. Cannon, R. Nayak, G. Lachapelle // Proceedings of the National Technical Meeting, Anaheim, CA, January 2000. – Anaheim : Institute of Navigation, 2000. – С. 98–105.
5. Распопов В.Я. Микросистемная авионика: учебное пособие / В.Я. Распопов. – Тула: «Гриф и К», 2010. – 248 с. – ISBN 978-5-8125-1467-9.
6. Матвеев В.В. Основы построения бесплатформенных инерциальных навигационных систем / В.В. Матвеев, В.Я. Распопов; [ред. В.Я. Распопов]. – СПб.: ГНЦ ЦНИИ "Электроприбор", 2009. – 300 с. – ISBN 978-5-900780-73-3.
7. Wolf R. A Kalman Filter for the Integration of a Low Cost INS and an attitude GPS [Електронний ресурс] / R. Wolf, B. Eissfeller, G.W. Hein // Proceedings of the International Symposium on Kinematic Systems (KIS), Banff, AB, Canada. – Banff : 1997. –Режим доступу : http://www.ifen.com/fileadmin/publications/KIS1997_GPS_INS.pdf
8. Miao C.X. Design on a Mult-Sensor Integrated Attitude Determination System for Small UAV [Електронний ресурс] / C.X. Miao, J.J. Cao, J.C. Fang // Journal of Harbin Institute of Technology, 2014. – Режим доступу : cnki.com.cn

9. Hao Y. Research on Data Fusion for SINS/GPSI Magnetometer Integrated Navigation based on Modified CDKF / Y. Hao, Z. Zhang, Q. Xia // IEEE International Conference on Progress in Informatics and Computing (PIC), Shanghai, 10-12 Dec. 2010. – № 2. – С. 1215 - 1219. – ISBN: 978-1-4244-6788-4.

10. Sokolovic V. Integration of INS, GPS, Magnetometer and Barometer for Improving Accuracy Navigation of the Vehicle / V. Sokolovic, G. Dikic, R. Stancic // Defence Science Journal. – 2013. – № 63(5). – С. 451-455.

11. Huang Z. J. Integration of MEMS Inertial Sensor-Based GNC of a UAV / Z. J. Huang and J. C. Fang // International Journal of Information Technology. – 2005.– № 11(10). – С. 123-132.

12. Brown A. Video-aided GPS/INS Positioning and Attitude Determination [Электронный ресурс] / A. Brown, R. Silva // NAVSYS Corporation. – Режим доступу : <http://www.navsys.com/papers/index.htm>

13. Ким Н.В. Определение углов крена и тангажа беспилотного летательного аппарата на основе обработки и анализа последовательности изображений подстилающей поверхности / Н.В. Ким, Н.В. Степанова // Авиакосмическое приборостроение. – 2006. – № 8. – С. 18-23.

14. Cornall T.D. Aircraft attitude estimation from horizon video / T.D. Cornall, G.K. Egan, A. Price // ELECTRONICS LETTERS 22nd, June 2006 – IET, 2006. – № 42(13). – С. 744 - 745.

15. Thurrowgood S. A vision based system for attitude estimation of UAVs / S. Thurrowgood, D. Soccol, R. J. D. Moore, D. Bland, M. V. Srinivasan // IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 11-15 October, St. Louis, Missouri, USA, 2009. – IEEE, 2009. – С. 5725 - 5730. – ISBN 978-1-4244- 3803-7.

16. Todorovic S. A Vision System for Intelligent Mission Profiles of Micro Air Vehicles / S. Todorovic, M. C. Nechyba // IEEE Transactions on Vehicular Technology.– 2004. – № 53(6). – С. 1713 - 1725.

17. Bao G.-Q. Vision-based horizon extraction for micro air vehicle flight control / G.-Q. Bao, S.-S. Xiong, Z.-Y. Zhou // IEEE Transactions on Instrumentation and Measurement. – 2005. – № 54(3). – С. 1067-1072.

18. Dusha D. Attitude estimation for a fixed-wing aircraft using horizon detection and optical flow [Електронний ресурс] / D. Dusha, W. Boles, R. Walker // 9th Biennial Conference of the Australian Pattern Recognition Society on Digital Image Computing Techniques and Applications, Glenelg, Australia, 3-5 December, 2007. – С. 485-492. – Режим доступу : <http://core.ac.uk/download/pdf/10882228.pdf>

19. Cornall T.D. Measuring horizon angle from video on a small unmanned air vehicle [Електронний ресурс] / T.D. Cornall, G.K. Egan // 2nd International Conference on Autonomous Robots and Agents, Palmerston North, New Zealand, December 13-15, 2004. – С. 339-344. – Режим доступу : http://www.ist.massey.ac.nz/conferences/icara2004/files/Papers/Paper59_ICARA2004_339_344.pdf

20. Chen Y. Vision-Based Horizon Detection and Target Tracking for UAVs / Y. Chen, A. Abushakra, J. Lee // 7th International Symposium, ISVC 2011, Las Vegas, NV, USA, September 26-28, 2011. Proceedings, Part II. Springer-Verlag Berlin Heidelberg, 2011. – С. 310–319.

21. Ettinger S.M. Vision-Guided Flight Stability and Control for Micro Air Vehicles / S.M. Ettinger, M.C. Nechyba, P.G. Ifju, M.R. Waszak // IEEE International Conference on Intelligent Robots and Systems, October 2002. – №3. – С. 2134 -2140.

22. Bao G. Towards micro air vehicle flight autonomy research on the method of horizon extraction / G. Bao, Z. Zhou, S. Xiong, X. Lin, X. Ye // In IEEE Instrumentation and Measurement Technology Conference, Ottawa, Canada, May 2003. – № 2. – С. 1387–1390.

23. Dusha D. Error analysis and attitude observability of a monocular GPS/visual odometry integrated navigation filter / D. Dusha, L. Mejias // The International Journal of Robotics Research. – 2012. – № 31(6). – С. 714–737.

24. Oreifej O. Horizon constraint for unambiguous UAV navigation in planar scenes / O. Oreifej, N. Lobo, M. Shah // IEEE International Conference on Robotics and Automation (ICRA), 2011. – С. 1159-1165. – ISBN 9781612843803.

25. Todorovic S. Sky/Ground Modeling for Autonomous MAV Flight / S. Todorovic, M. C. Nechyba, P. G. Ifju // IEEE International Conference on Robotics and Automation, September 14-19, 2003. – № 1. – С. 1422 - 1427

26. Bry A. State Estimation for Aggressive Flight in GPS-Denied Environments Using Onboard Sensing [Електронний ресурс] / A. Bry, A. Bachrach, N. Roy // Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), 2012. – Режим

27. Bachrach A. RANGE – Robust Autonomous Navigation in GPS-denied Environments / A. Bachrach та ін. // IEEE International Conference On Robotics and Automation (ICRA), 2010. – С. 1096-1097. – Режим доступу : <http://hdl.handle.net/1721.1/65608>.

28. Bachrach A. Autonomous flight in unknown indoor environments / A. Bachrach, R. He, N. Roy // International Journal of Micro Air Vehicles. – 2009. – № 1(4). – С. 217–228.

29. Muratet L. A Contribution to Vision-Based Autonomous Helicopter Flight in Urban Environments [Електронний ресурс] / L.Muratet, S. Doncieux, Y. Briere, J.-A. Meyer // Preprint submitted to Elsevier Science. – Elsevier, 2004. – Режим доступу : <http://www.isir.upmc.fr/files/2005ACLI465.pdf>

30. Методичні рекомендації до виконання дипломного проекту з освітньо-кваліфікаційного рівня «Бакалавр» напряму підготовки 6.050102 «Комп'ютерна інженерія» фахового спрямування «Комп'ютерні системи та мережі» / О.М. Березький, Л.О. Дубчак, Р.Р. Трембач, Г.М. Мельник, Ю. М. Батько, С.В. Івасьєв / Під ред. О.М. Березького. – Тернопіль: ТНЕУ, 2016. – 60 с.

31. Методичні вказівки до написання техніко-економічного розділу для дипломних проектів на здобуття освітньо-кваліфікаційного рівня «Бакалавр» напряму підготовки 6.050102 «Комп'ютерна інженерія» / І.Р. Паздрій. – Тернопіль: ТНЕУ, 2015. – 36 с.