

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ОДЕСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
ІМЕНІ І. І. МЕЧНИКОВА  
ФАКУЛЬТЕТ МАТЕМАТИКИ, ФІЗИКИ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ  
КАФЕДРА КОМП'ЮТЕРНИХ СИСТЕМ ТА ТЕХНОЛОГІЙ

# **КОМП'ЮТЕРНИЙ ЗІР ТА НАВІГАЦІЯ РОБОТОТЕХНІЧНИХ СИСТЕМ**

## **ЗМІСТОВНИЙ МОДУЛЬ 1 НАВІГАЦІЯ МОБІЛЬНИХ РОБОТІВ**

Методичні вказівки  
до лабораторних занять для студентів  
першого (бакалаврського) рівня вищої освіти  
галузі знань 12 Інформаційні технології

ОДЕСА  
Олді+  
2023

**УДК 004.822(072)**

**К637**

**Укладач:**

**І. В. Шаріпова**, старший викладач кафедри комп'ютерних систем та технологій Одеського національного університету імені І. І. Мечникова.

**Рецензенти:**

**Ю. А. Ніцук**, доктор фіз.-мат.наук, професор, декан факультету математики, фізики та інформаційних технологій ОНУ імені І. І. Мечникова;

**Ю. А. Максименко**, кандидат технічних наук, доцент, начальник кафедри організації розвідувальної-інформаційної роботи та технічних засобів розвідки ФПСВР та ССО Військова академія (м. Одеса).

*Рекомендовано вченою радою факультету математики,  
фізики та інформаційних технологій ОНУ імені І. І. Мечникова  
(Протокол №5 від 12 травня 2023 р.)*

**Комп'ютерний зір та навігація робототехнічних систем** : Змістовний модуль 1 "Навігація мобільних роботів" : методичні вказівки до лабораторних занять для студентів першого (бакалаврського) рівня вищої освіти галузі знань 12 Інформаційні технології / уклад.: І. В. Шаріпова. – Одеса : Олді+, 2023. – 40 с.

У методичних вказівках розкривається програма курсу, даються рекомендації для опанування лекційним матеріалом. Розглянуто зміст, завдання, наведено приклади розробки програм, порядок їх виконання. Рекомендовано вимоги до оформлення протоколів, правила захисту та оцінки. Наведено перелік завдань з темами лабораторних робіт, контрольні запитання та список рекомендованої літератури.

УДК 004.822(072)

© Шаріпова І. В., укладання, 2023

## Зміст

ВСТУП.....	4
ЛАБОРАТОРНА РОБОТА №1 .....	7
ЛАБОРАТОРНА РОБОТА №2 .....	13
ЛАБОРАТОРНА РОБОТА №3 .....	19
ЛАБОРАТОРНА РОБОТА №4 .....	24
ЛАБОРАТОРНА РОБОТА № 5 .....	32
ПРАВИЛА ОФОРМЛЕННЯ ПОЯСНЮВАЛЬНОЇ ЗАПИСКИ.....	36
СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ .....	37
Додаток А .....	38

## ВСТУП

Комп'ютерний зір – вид діяльності, в якій для отримання даних застосовують статистичні методи та використовують моделі, які побудовані за допомогою геометрії, фізики, математики і теорії навчання.

Комп'ютерний зір застосовують досить широко в управлінні мобільними роботами, військових додатках, промислових засобах спостереження, а також у сфері взаємодії людина/комп'ютер, пошуку зображення в бібліотеках, аналізі медичних зображень та реалістичному передаванні змодельованих сцен у комп'ютерній графіці.

Загальні вимоги до мобільних роботів: висока рухливість та прохідність; надійність у діях за непередбачених природних умов та в середовищах, що спеціально підготовлені для проживання людини; конструкція має забезпечувати швидке розгортання та високий рівень мобільності.

Робототехніки виділяють такі навігаційні схеми:

- глобальну (визначення абсолютних координат пристрою під час руху довгими маршрутами);
- локальну (визначення координат пристрою по відношенню до деякої (зазвичай стартової) точки);
- персональну (позиціонування роботом частин свого тіла і взаємодія з прилеглими предметами, що актуально для пристроїв, забезпечених маніпуляторами).

Глобальні методи засновані на тому, що перед початком руху роботів повністю відома карта місцевості. Знаючи своє місцезнаходження, точку фінішу, а також розташування всіх перешкод, він, використовуючи заданий алгоритм дій, знаходить найкоротший шлях від старту до фінішу і після цього долає цей шлях. Перевага глобальних методів навігації полягає в можливості заздалегідь спланувати оптимальний маршрут руху, спираючись на глобальну інформацію про зовнішнє середовище. Найбільш значимий недолік таких методів – підвищена обчислювальна складність і необхідність зберігання карти середовища (найчастіше великого розміру).

Локальні методи навігації використовуються в тих випадках, коли роботу не відома глобальна карта зовнішнього середовища або перешкоди в цьому середовищі мають динамічний характер (можуть з'являтися і зникати, змінювати своє місце розташування). Однією з основних принципів особливостей бездротових телекомунікаційних систем є фактор мобільності та можливість ідентифікації місцезнаходження відповідного комунікаційного пристрою користувача. Методи та засоби бездротового позиціонування

викликають значний інтерес, оскільки вони забезпечують формування основи для безперервного зростання кількості та якості мобільних програмно-алгоритмічних застосунків. Прикладами таких застосунків є звичайні навігаційні системи.

Значну увагу як дослідники, так і практики-програмісти звертають на розроблення мобільних програмно-алгоритмічних застосунків, що ґрунтуються на інформації про поточне місцезнаходження користувача на відкритому просторі (міста, дороги, поля, ліси, океани тощо), водночас, недостатньо уваги приділяють вирішенню проблем, що виникають у процесах позиціонування та навігації користувача в середині будівель.

При вирішенні задач навігації роботів, одночасна локалізація і картографування (SLAM) є алгоритмічною обчислювальною задачею побудови

і оновлення мапи невідомого оточення з одночасним відстежуванням місцезнаходження рухаючись по ньому. Популярні методи вирішення включають в себе фільтр часток і розширений фільтр Калмана. Алгоритми SLAM обмежуються наявними ресурсами, таким чином не можуть бути абсолютно досконалими, бо досягають оперативної доступності.

Таким чином **метою дисципліни** є сформувати систему знань студентів в області робототехнічних систем, знання функціональних складових елементів, які лежать в основі інтелектуальних робототехнічних систем, на базі яких дипломований фахівець зможе приймати участь у розробці, застосуванні за призначенням і експлуатації таких систем різного функціонального призначення в зацікавлених установах та виробництвах усіх форм власності. Вироблення у студентів практичних навичок використання теоретичного матеріалу в галузі моделювання комп'ютерних систем, вивчення застосування та використання сучасних програмних засобів моделювання при розробці моделей для комп'ютерної інженерії.

Лабораторні роботи з дисципліни виконуються з метою закріплення та поглиблення теоретичних та практичних знань та вмінь, набутих у процесі засвоєння всього навчального матеріалу дисципліни.

Час, потрібний для виконання лабораторних робіт згідно навчального плану – 12 години у комп'ютерному класі та до 22 годин самостійної роботи.

Тематика лабораторних робіт присвячена оволодінню сучасних алгоритмів визначення поточного значення координат та визначення можливості реалізації навігаційної системою робота.

## МЕТА, ЕТАПИ ПРОВЕДЕННЯ ТА ЗАХИСТ ЛАБОРАТОРНИХ РОБІТ

Лабораторні роботи з дисципліни виконуються з метою закріплення та поглиблення теоретичних та практичних знань та вмінь, набутих у процесі засвоєння всього навчального матеріалу дисципліни:

- ✓ закріплення, поглиблення та узагальнення теоретичних знань і розвиток навичок їх практичного застосування в галузі комп'ютерного зору та навігації робототехнічних систем
- ✓ самостійне розв'язання задач проектування та розробки алгоритмів оброблення зображень у системах технічного зору;
- ✓ уміння користуватися відповідною довідковою літературою, програмними засобами;

Проведення лабораторних робіт містить такі етапи:

- ✓ визначення теми, завдання і повторення теоретичного матеріалу;
- ✓ безпосереднє виконання роботи;
- ✓ оформлення пояснювальної записки;
- ✓ захист.

Після виконання лабораторної роботи і вирішення всіх поставлених у ній задач студент оформлює звіт з лабораторної роботи – протокол. Виконаний протокол студент підписує і після дозволу керівника він допускається до захисту. Якщо керівник не допускає студента до захисту, то це питання обговорюється на засіданні кафедри у його присутності.

Захист лабораторної роботи – це форма перевірки якості виконання програми та знань, отриманих під час виконання лабораторних робіт та на лекціях.

Під час захисту студент робить доповідь по суті програми та відповідає на запитання.

Якість протоколу та його захист оцінюється в балах (0- 5), за шкалою ECTS (A, B, C, D, E, FX, F) та за національною шкалою «відмінно», «добре», «задовільно», «незадовільно».

## ЛАБОРАТОРНА РОБОТА №1

### Тема: «Дослідження алгоритмів EKF-SLAM»

Реалізація навігаційної системи, якій заздалегідь відомі орієнтири або карта простору, при наявності досить точних сенсорів, не представляє великих складнощів. Побудова карти при постійно відомому положенні робота також в значній мірі вирішена проблема. Набагато складніше вирішувати ці проблеми одночасно, не знаючи заздалегідь ні карти, ні положення робота. Тому одним з найважливіших компонентів систем комп'ютерного зору, пов'язаних з орієнтацією в просторі, є SLAM-модуль. Саме він виконує глобальну оптимізацію отриманих вимірювань. Фактично, основна задача SLAM – мінімізація помилок визначення зсуву камери, за рахунок чого вирівнюється траєкторія руху робота і вивчений простір, на основі якого формується карта світу.

#### Постановка задачі

Написати та розробити програму у програмному комплексі Octave для моделювання пересування роботу у 2D просторі відносно міток використовуючи метод EKF-SLAM

#### Зміст протокола

1. Постановка задачі
2. Хід роботи (текст програми або скриншоти інтерфейсу)
3. Висновки щодо роботи

**Варіанти завдань:** смоделювати переміщення роботу у приміщенні відносно 8-ми міток (варіант 12 міток, 16 міток)

#### Хід роботи (приклад виконання)

Для розуміння методів обробки зображень важливо розв'язувати практичні завдання, а для цього треба мати комп'ютерний інструмент, здатний ілюструвати застосування тих чи інших методів. У нашому курсі ми будемо використовувати пакет Octave. Ви можете використовувати MatLab, бо Octave був написаний з урахуванням сумісності з MatLab і реалізує більшість його можливостей. Різниця в мові програмування майже немає. Навіть пакети розширення спроектовані таким чином, щоб схожі математичні перетворення виконувалися однойменними функціями з мінімальною різницею між їх аргументами. З MatLab і Octave є середовищами для виконання технічних і наукових обчислень. Їх базовою конструкцією є масив елементів (матриця), який не вимагає задання розмірності. Це дозволяє легко формулювати умови і

будувати розв'язки багатьох обчислювальних задач, що потребують матричного зображення. Обидві системи мають розширення у вигляді наборів спеціалізованих програм, які по-англійськи називаються toolbox (набір інструментів або пакет розширення). Більшість функцій Octave, призначених для роботи з зображеннями, зосереджено в пакеті 'image', який з деякими обмеженнями є двійником пакету 'Image Processing Toolbox' MatLab. Ці пакети розширюють можливості стандартного середовища для розв'язання задач цифрової обробки зображень.

Клас алгоритмів SLAM, що використовують розширений фільтр Калмана. встановлюємо програмне середовище Octave на комп'ютер. Після того, як завантаження коректно завершилося, запускаємо програму Octave (рис. 1).

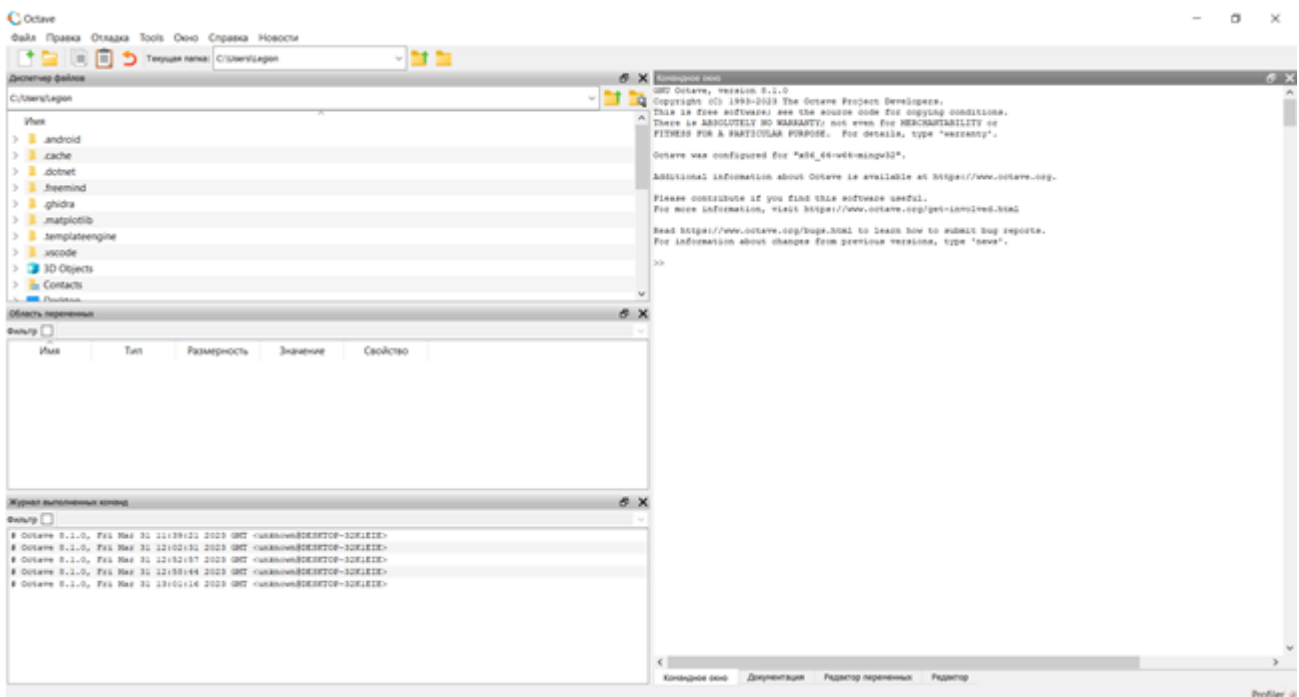


Рисунок 1 – Робоче середовище програми Octave

Наступним кроком створюємо програму на мові Matlab. Вихідний код основного модуля програми виглядатиме таким чином (лістинг 1). Код програми з коментарями нижче:

Основний код програми:

```
% This is the main extended Kalman filter SLAM loop. This script calls all the required
% functions in the correct order.
%
% You can disable the plotting or change the number of steps the filter
% runs for to ease the debugging. You should however not change the order
% or calls of any of the other lines, as it might break the framework.
%
```



```

% If you are unsure about the input and return values of functions you
% should read their documentation which tells you the expected dimensions.

% Turn off pagination:
more off;

% clear all variables and close all windows
clear all;
close all;

% Make tools available
addpath('tools');

% Read world data, i.e. landmarks. The true landmark positions are not given to the robot
landmarks = read_world('./data/world.dat');
% load landmarks;
% Read sensor readings, i.e. odometry and range-bearing sensor
data = read_data('./data/sensor_data.dat');
%load data;

INF = 9999;
% Get the number of landmarks in the map
N = size(landmarks,2);

% observedLandmarks is a vector that keeps track of which landmarks have been observed
so far.
% observedLandmarks(i) will be true if the landmark with id = i has been observed at some
point by the robot
observedLandmarks = repmat(false,1,N);

% Initialize belief:
% mu: 2N+3x1 vector representing the mean of the normal distribution
% The first 3 components of mu correspond to the pose of the robot,
% and the landmark poses (xi, yi) are stacked in ascending id order.
% sigma: (2N+3)x(2N+3) covariance matrix of the normal distribution
mu = repmat([0.0], (2*N+3), 1);
robSigma = zeros(3);
robMapSigma = zeros(3,2*N);
mapSigma = INF*eye(2*N);
sigma = [[robSigma robMapSigma];[robMapSigma' mapSigma]];

% toogle the visualization type
showGui = true; % show a window while the algorithm runs

```

```

%showGui = false; % plot to files instead

% Perform filter update for each odometry-observation pair read from the
% data file.
for t = 1:size(data.timestep, 2)
%for t = 1:80

    % Perform the prediction step of the EKF
    [mu, sigma] = prediction_step(mu, sigma, data.timestep(t).odometry, t == 1);

    % Perform the correction step of the EKF
    [mu, sigma, observedLandmarks] = correction_step(mu, sigma, data.timestep(t).sensor,
observedLandmarks);

    %Generate visualization plots of the current state of the filter
    plot_state(mu, sigma, landmarks, t, observedLandmarks, data.timestep(t).sensor,
showGui);
    disp("Current state vector:")
    disp("mu = "), disp(mu)
endfor

disp("Final system covariance matrix:"), disp(sigma)
% Display the final state estimate
disp("Final robot pose:")
disp("mu_robot = "), disp(mu(1:3)), disp("sigma_robot = "), disp(sigma(1:3,1:3))

```

Запустимо програму, відкриється вікно анімації, де ми бачимо початковий стан роботи програми (рис. 2). У режимі реального часу ми можемо спостерігати за його процесом переміщення (рис. 3). Коли робот доходить до останньої точки, його переміщення зупиняється та ми можемо бачити його кінцевий стан (рис. 4).

Продемонструємо скріншот виконаної програми:

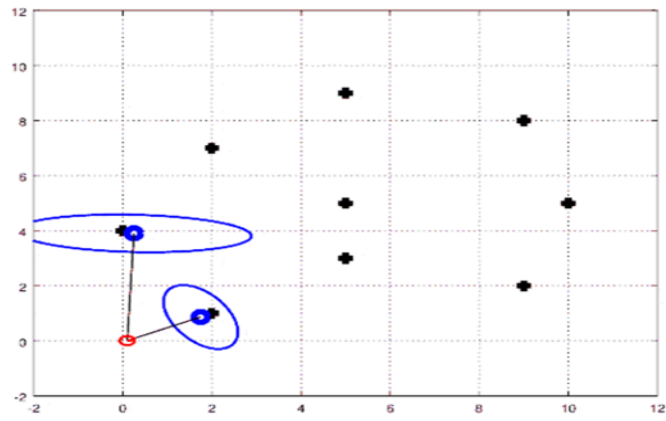


Рисунок 2 – Початкове положення робота

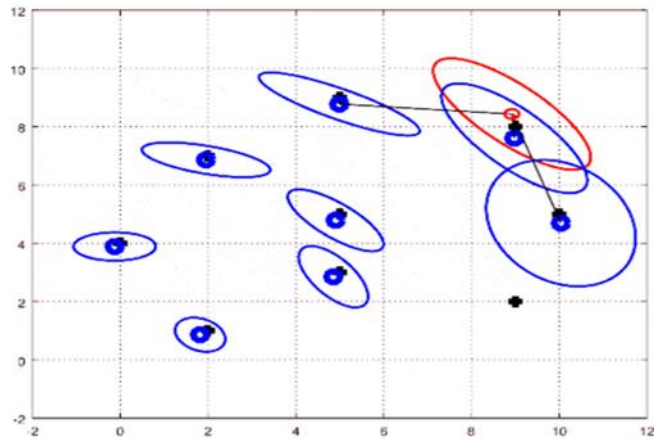


Рисунок 3 – Робот у процесі переміщення

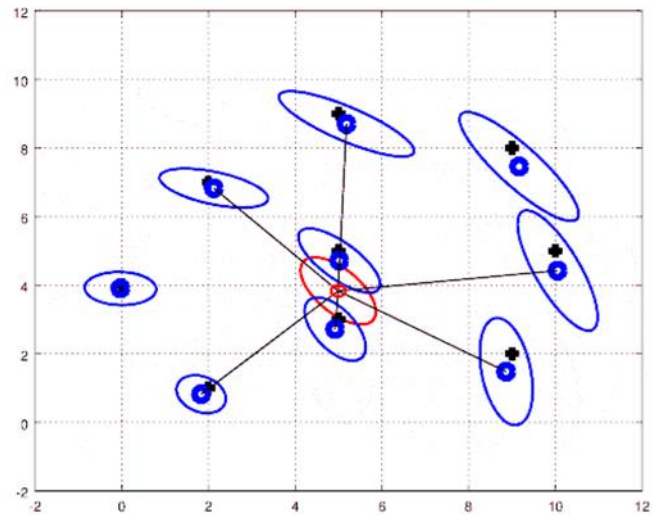


Рисунок 4 – Кінцеве положення робота

Було смодельовано переміщення роботу у приміщенні відносно 8-ми міток. При цьому робот брав у увагу те, що він має проходити через мітку один раз, та використовує не лише попередні мітки для навігації, а також наступні, які попадають у поле зору.

### **Висновки щодо роботи (приклад)**

У цій лабораторній роботі ми познайомилися з дослідженням алгоритмів EKF-SLAM.

Програма виконує поставлені завдання. Працює без помилок у штатному режимі.

### **Перелік питань на захист лабораторної роботи**

1. Сутність методів EKF-SLAM.
2. Особливості комп'ютерної реалізації EKF-SLAM
3. Оцінювання складності алгоритмів EKF-SLAM та побудови карти місцевості відповідно до опорних точок з погляду його алгоритмічно-програмної реалізації
4. Математичні співвідношення фільтрів Калмана щодо їх адаптації до особливостей алгоритмічної та програмної реалізації в бортових EOM.

## ЛАБОРАТОРНА РОБОТА №2

### Тема: «Дослідження алгоритмів DP -SLAM»

DP-SLAM – реалізація, що використовує свідчення лазерного далекоміра і фільтр частинок для зберігання гіпотез про становище робота і конфігурації навколишнього його сцени . Фільтр частинок (послідовний метод Монте-Карло) – рекурсивний алгоритм для чисельного вирішення проблем оцінювання (фільтрації, згладжування), особливо для нелінійних випадків.

Метод полягає в тому, що в фільтрі підтримується безліч гіпотез про поточний стан робота. У початковий момент генерується випадковий набір гіпотез. В ході роботи алгоритму деякі з них будуть відсіюватися через невідповідності вторинними ознаками системи, які можуть бути виміряні більш точно. Після завершення циклу сканування вибирається найбільш вірогідна з залишених в фільтрі гіпотез.

Для зберігання і структурування великого числа проміжних карт авторами була реалізована деревоподібна структура (рисунок 2.1). Вся інформація, необхідна для вирішення обох підзадач, локалізації і коригування карти, поміщається в одному фільтр частинок.

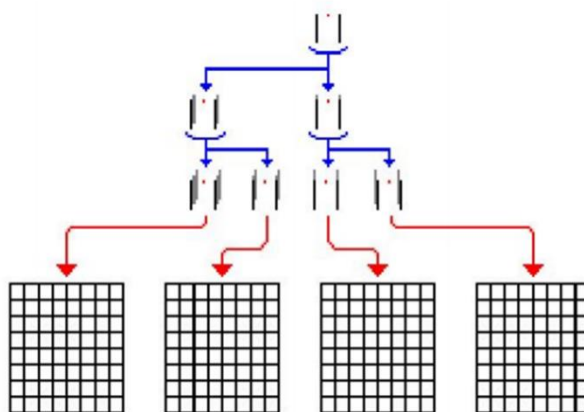


Рисунок 1 – Деревоподібна система зберігання "гіпотетичних" карт

Варто звернути увагу на те, що всі карти в цьому прикладі узгоджуються з початковим наглядом. Дві ліві карти узгоджуються зі спостереженнями, зробленими лівим ребром кореня, в той час як дві найправіші карти узгоджуються зі спостереженнями, зробленими правим ребром кореня.

Зберігання та повторне копіювання цих розділів карти кожен раз, коли частка буде повторно відбиратися - було б марною тратою пам'яті і часу. Замість

цього використовується єдина сітка зайнятості, яка зберігає дерево спостережень на кожному квадраті.

На рис 2.2 показано, як кожна частка вставляє свої спостереження в глобальну сітку. Вони зберігаються у вигляді збалансованого дерева, індексованого за унікальним ідентифікатором, який присвоюється кожній частинці.

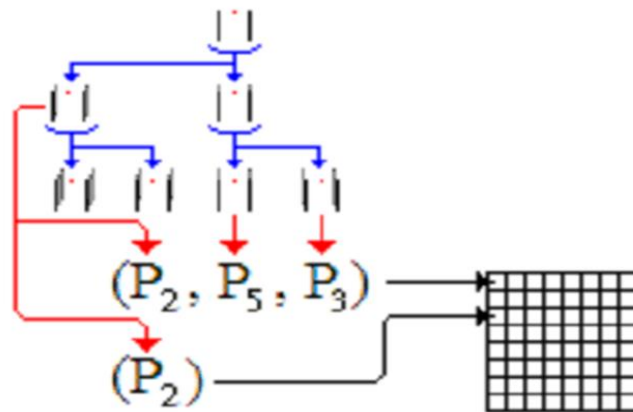


Рисунок 2 – Будування карти за допомогою часток

Розглянемо дві реалізації алгоритму DP-SLAM. Перша реалізація одночасно підтримує до 9000 можливих гіпотез, і відсіває тільки ті з них, які в силу невиконання яких-небудь умов абсолютно точно не можуть відповідати реальному стану системи. Другий підхід на кожному кроці алгоритму вибирає тільки одну найвірогіднішу гіпотезу і оновлює карту виходячи з неї. У цьому випадку має перевагу реалізація, яка одночасно підтримує велику кількість гіпотез, адже досить імовірна ситуація, при якій в силу шумів самої різної природи найбільш імовірною стане гіпотеза, досить сильно відрізняється від реального стану системи.

### Постановка задачі

Написати та розробити програму для моделювання пересування роботу у 2D просторі відносно міток, використовуючи метод DP-SLAM

### Зміст протокола

1. Постановка задачі
2. Хід роботи (текст програми або скріншоти інтерфейсу)
3. Висновки щодо роботи

Карта навколишнього середовища

Карта навколишнього середовища має бути представлена у вигляді карти модельованого середовища. Кожен рядок містить в собі:

- унікальний ідентифікатор спостережуваного орієнтиру;
- x-координату;
- y-координату.

0 2 1

1 0 4

2 2 7

3 9 2

4 10 5

5 9 8

6 5 5

7 5 3

8 5 9

9 4 7

10 8 6

Встановлюємо програмне середовище Octave на комп'ютер. Після завантаження запускаємо програму Octave.

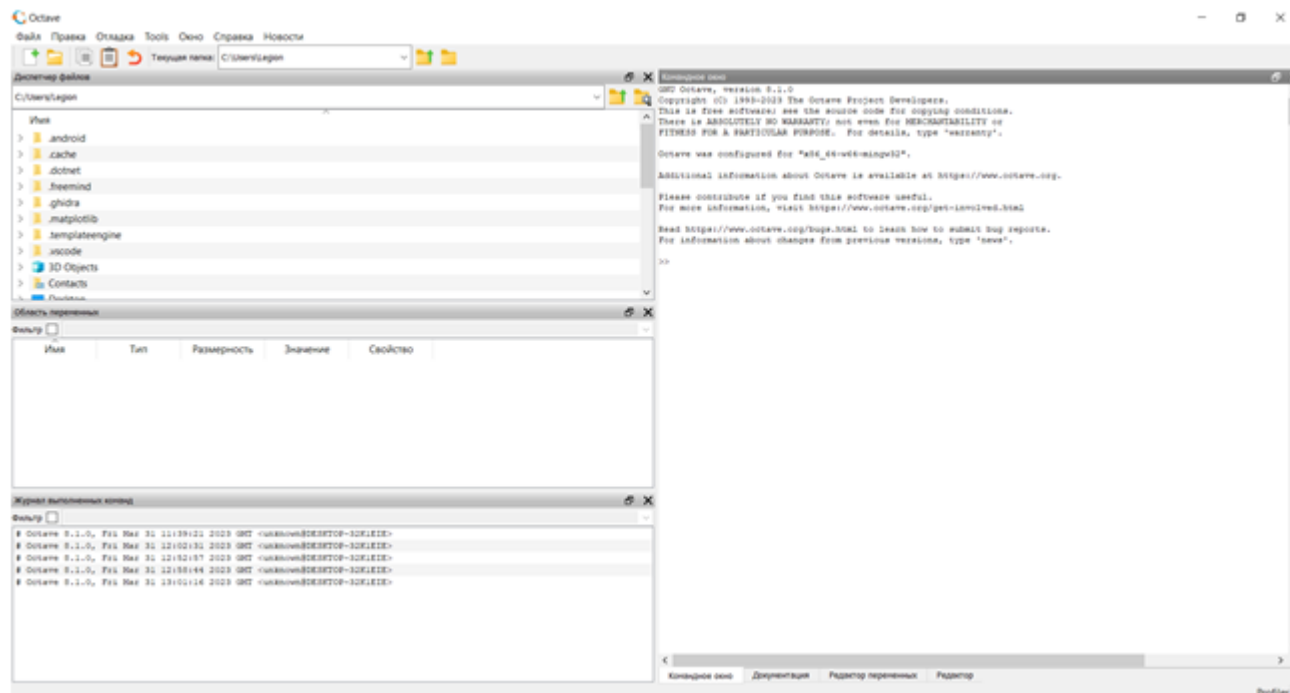


Рисунок 3 – Робоче середовище програми Octave

% Turn off pagination:

more off;

```

clear all;
close all;
% Make tools available
addpath('tools');
% Read world data, i.e. landmarks. The true landmark positions are
not given to the robot
landmarks = read_world('../data/world.dat');
% Read sensor readings, i.e. odometry and range-bearing sensor
data = read_data('../data/sensor_data.dat');
% Get the number of landmarks in the map
N = size(landmarks,2);
noise = [0.005, 0.01, 0.005]';
% how many particles
numParticles = 100;
% initialize the particles array
particles = struct;
for i = 1:numParticles
    particles(i).weight = 1. / numParticles;
    particles(i).pose = zeros(3,1);
    particles(i).history = cell();
    for l = 1:N % initialize the landmarks aka the map
        particles(i).landmarks(l).observed = false;
        particles(i).landmarks(l).mu = zeros(2,1); % 2D position of
the landmark
        particles(i).landmarks(l).sigma = zeros(2,2); % covariance of
the landmark
    end
end
% toggle the visualization type
showGui = true; % show a window while the algorithm runs
%showGui = false; % plot to files instead
% Perform filter update for each odometry-observation pair read
from the
% data file.
for t = 1:size(data.timestep, 2)
%for t = 1:50

```



```

printf('timestep = %d\n', t);
% Perform the prediction step of the particle filter
particles = prediction_step(particles,
data.timestep(t).odometry, noise);
% Perform the correction step of the particle filter
particles = correction_step(particles,
data.timestep(t).sensor);
% Generate visualization plots of the current state of the
filter
plot_state(particles, landmarks, t, data.timestep(t).sensor,
showGui);
% Resample the particle set
particles = resample(particles);
end

```

### Лістинг 1 – Основний код програми

Нижче наведемо дані з файлу “world.dat” (лістинг 2), що визначають координати положення ключових точок “landmarks”.

Для демонстрації роботи програми використовується застосунок Octave та наступні координати точок: (10;5), (9;8), (2;7), (9;2), (2;1), (5;5), (0;4), (5;3), (5;9). Робота алгоритму наведена на рисунках 1, 2 та 3.

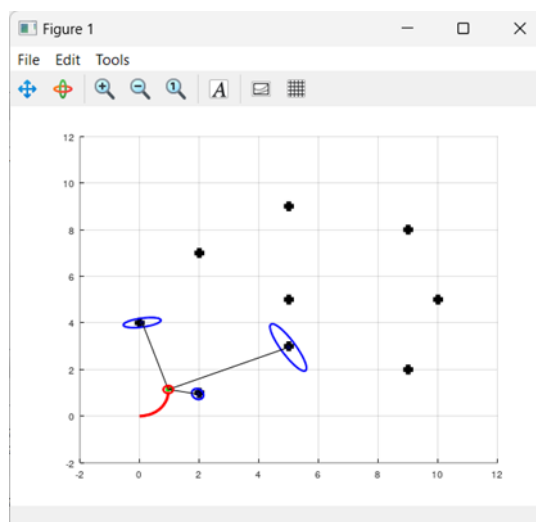


Рисунок 4 – початок роботи алгоритму

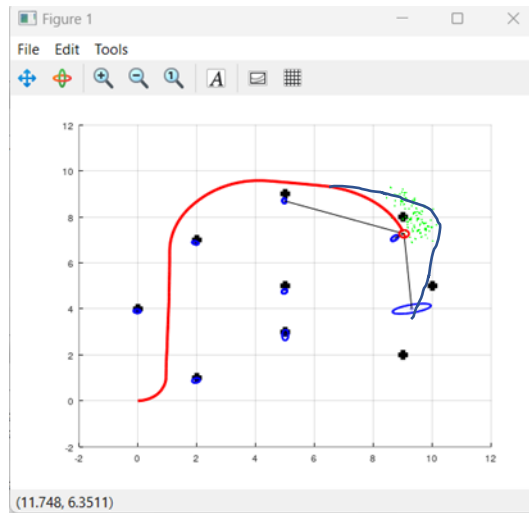


Рисунок 5 – проміжний результат роботи алгоритму

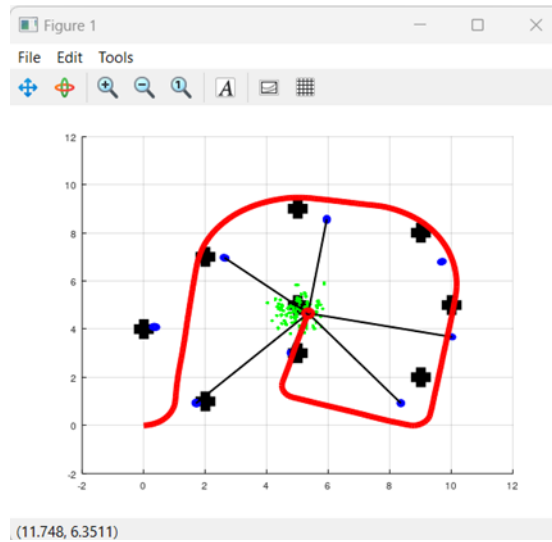


Рисунок 6 – Завершення роботи алгоритму

### Перелік питань на захист лабораторної роботи

1. Сутність методів DP –SLAM.
2. Особливості комп'ютерної реалізації DP –SLAM
3. Оцінювання складності алгоритмів DP –SLAM та побудови карти місцевості відповідно до опорних точок з погляду його алгоритмічно-програмної реалізації

## ЛАБОРАТОРНА РОБОТА №3

Тема: «Дослідження алгоритмів GMapping»

### Постановка задачі

Алгоритм GMapping. В основі даного алгоритму лежить метод рекурентної фільтрації частинок Рао-Блеквелла на ґратчастій карті. Кожна частинка є потенційною траєкторією робота і містить в собі інформацію про стан карти робочого середовища в поточний момент часу. Було запропоновано два підходи, які дозволяють збільшити продуктивність фільтра для можливості вирішення завдання SLAM в реальному часі, це допоміжна функція розподілу і техніка адаптивного ресемплінгу. Алгоритм призначений для фільтрації оцінки положення робота, яка обчислюється деяким методом реєстрації сканів, і служить для підвищення точності локалізації. Перевагою даного алгоритму є висока точність побудови карти. Однак недолік полягає в тому, що розглянутий алгоритм має досить високу обчислювальну складність.

Фільтр Рао-Блеквелла для картографування поступово обробляє спостереження і свідчення одометра в міру їх надходження. Це робиться шляхом поновлення набору зразків, які представляють апостеріор на карті і траєкторію руху транспортного засобу. Це робиться шляхом виконання наступних чотирьох кроків:

1) Вибірка: наступне покоління часток  $x_t(i)$  отримано з поточного покоління шляхом вибірки з розподілу пропозицій  $\pi(x_t | z_{1:t}, u_{0:t})$ ;

2) Зважування часток: кожній частці присвоюється її власну вагу відповідно до  $w(i) = p(x_t(i) | z_{1:t}, u_{0:t}) \pi(x_t(i) | z_{1:t}, u_{0:t})$ . (2.13) Ваги  $w$  враховують той факт, що розподіл пропозицій  $\pi$  в цілому не дорівнює істинному розподілу станів-спадкоємців.

3) Повторна вибірка: частинки з низькою вагою зазвичай замінюються зразками з високою вагою. Цей крок необхідний, оскільки для апроксимації неперервного розподілу використовується тільки кінцеве число частинок. Крім того, повторна вибірка дозволяє застосовувати фільтр частинок в ситуаціях, коли дійсний розподіл відрізняється від передбачуваного.

4) Оцінка карти: для кожної вибірки поз, відповідна оцінка карти обчислюється на основі траєкторії і історії спостережень відповідно до  $p(m_t(i) | x_t(i), z_{1:t})$ .

GMapping використовує теорію Рао-Блеквелла, при цьому:

- зменшується невизначеність по положенню робота на етапі прогнозування циклу передбачення і корекції;
- проблема виснаження частинок вирішується вибіркової передискретизацією. Модель осередку сітки, яка використовується GMapping, зберігає середні значення зайнятості і положення всіх перешкод, виявлених лазерним далекоміром в цьому осередку.

### **Хід роботи**

GMapping – це інструмент для створення карток, що використовує програмну бібліотеку OpenSlam. Програма дозволяє створювати карти для використання в інших програмах.

Програма GMapping використовує можливості LIDAR еталонного робота Carter та дозволяє використовувати одометрію для позиціонування робота залежно від його можливостей. Для покращення результатів необхідно використовувати інерційний вимірювальний блок (IMU). Хоча ви можете запустити програму GMapping на робочому столі, для фактичного картографування її необхідно розгорнути та запустити на роботі Carter.

Розгортання та запуск програми GMapping.

Розгорніть `//apps/carter/gmapping:gmapping-pkg` на роботі, як описано в розділі *«Параметри консолі програми»* .

Запустіть програму GMapping, вибравши профіль цільового робота:

```
bob@carter1:~/gmapping-pkg$ ./apps/carter/gmapping/gmapping--more  
"apps/carter/robots/carter_1.json"
```

### **Висновок**

Приклад запису карти в `/tmp/map.img_<N>.png` папку під час виконання. Вкажіть вихідний шлях з наступними параметрами конфігурації для кодлета Gmapping:

```
"config": {  
  "gmapping.gmapping": {  
    "GMapping": {  
      "file_path": "/tmp"  
    }  
  }  
}
```

Приклад

Послідовність зображень нижче показує процес картування з впливом часу від першого збору даних LIDAR до карти (мапи) будівлі

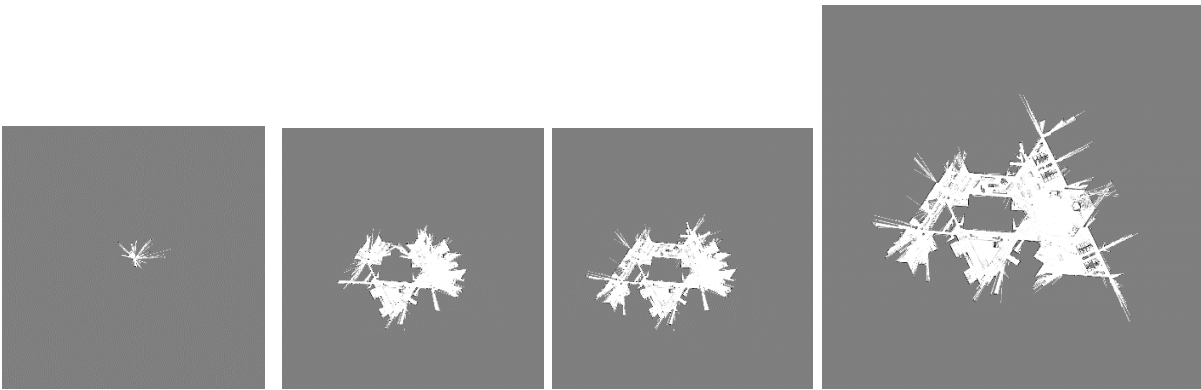


Рисунок 1 – Початок роботи алгоритму



Рисунок 3 – Проміжна стадія роботи алгоритму

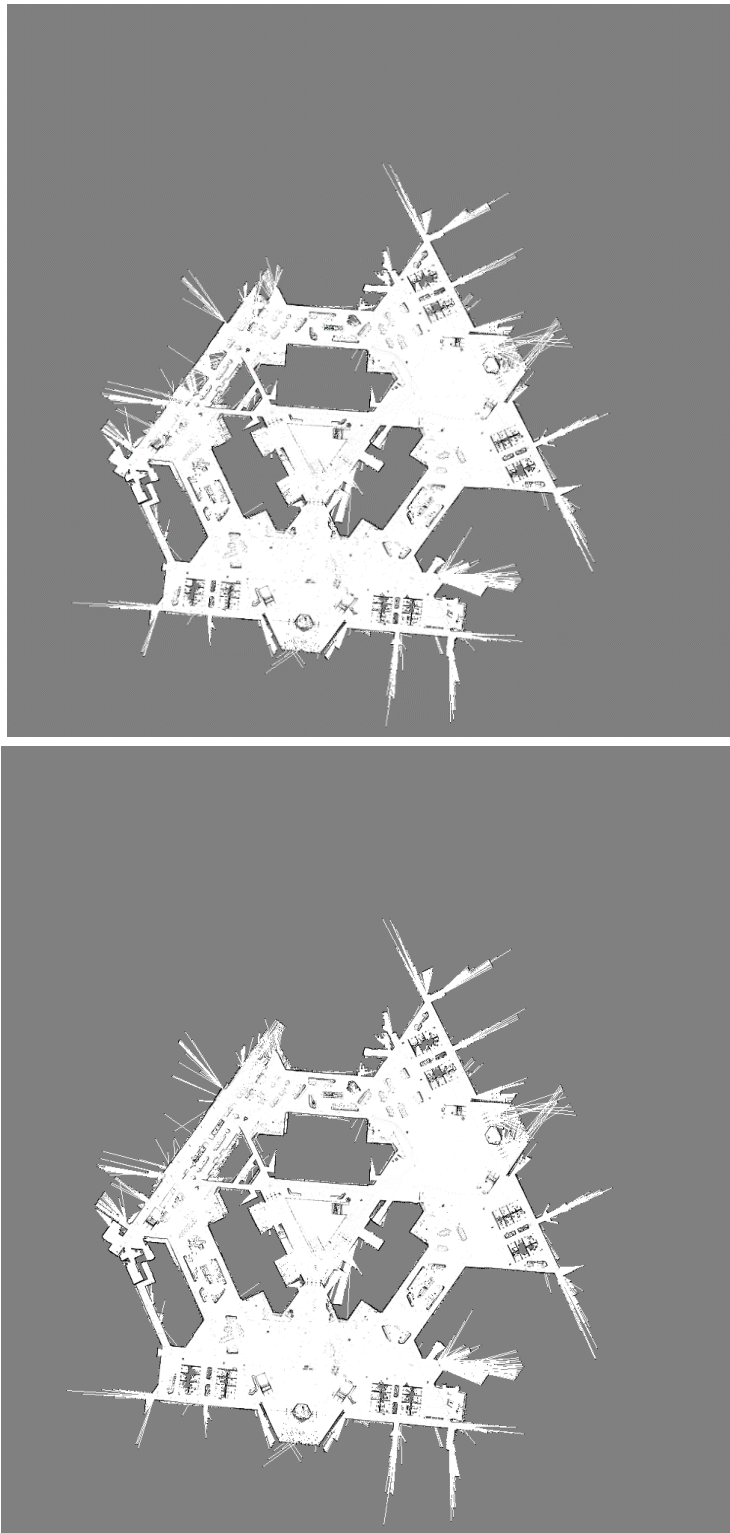


Рисунок 4 – Закінчення роботи алгоритму

#### Рекомендації щодо картування

Швидкість, з якою рухається робот під час картування, впливає на результати. Чим нижча швидкість, тим більша кількість вибірок LIDAR, що призводить до підвищення точності. Необхідно уникати різких поворотів та

мати на увазі налаштування робота на обмеження максимальної лінійної та кутової

швидкості. Регулярно зіставляйте та замикайте контури шляху, щоб виправити дрейф та помилки в одометрії та інерційних вимірах під час картування. Глибина відповідності є кінцевою. Там, де це можливо, переміщайтеся по колу навколо блоків будівлі, таких як: контури кабін і великі архітектурні елементи. Немає потреби проїжджати через вже нанесену на карту область; це збільшує похибку.

Підтримуйте достатню кількість опорних точок від кадру до кадру. Це особливо важливо при виході чи вході в нові зони або під час поворотів в коридор. Уникайте їзди надто близько до стін. Виберіть таку кількість (збігаючих) частинок, яка дозволяє отримати топологію вашої будівлі, не втрачаючи збігу сканування або повертаючись тільки до одометрії, що призводить до поганих результатів картування.

Використання досить довгого діапазону для збереження опорних точок, але використовуючи невеликий діапазон оновлення, дає можливість намалювати чітке зображення карти. В ідеалі діапазон оновлення повинен бути не більше половини довжини найбільшої області для відображення.

Запишіть свої канали сканування та одометрії, щоб відтворити та створити нові карти з іншими параметрами. Налаштуйте параметри конфігурації, щоб поекспериментувати з варіантом використання зіставлення.

Під час роботи з картами, створеними GMapping або logamppings, обріжте сірі краї зображення картки. Це зменшує розмір файлу та підвищує продуктивність алгоритмів, що використовують зображення карти. Після модифікації збережіть ці карти у стислому вигляді у відтинках сірого формату PNG для значного зменшення розміру файлу.

### **Перелік питань на захист лабораторної роботи**

1. Сутність методу G - Mapping.
2. Розглянути особливості комп'ютерної реалізації G - Mapping з точки зору комп'ютерної інженерії.
3. Розглянути властивості G – Mapping.
4. Зробити аналіз G - Mapping (таблиця, 3-4 властивості, висновок – переваги та недоліки).

## ЛАБОРАТОРНА РОБОТА №4

**Тема: «Алгоритм magnetic matching approach - MM з підтримкою технології Wi-Fi»**

Позиціонування є важливою інформацією для особистої навігації, і для отримання інформації щодо місцезроположення використовувалися різні методи. Зокрема, знаходження положення людини в приміщенні важливо, оскільки його можна використовувати для працівників зі спеціальними обов'язками, для підвищення ефективності роботи. Крім того, персональна навігаційна система може бути застосована до людей похилого віку та пацієнтів для підтримки їх безпеки. Аналіз позицій пішоходів також може бути використаний в різних внутрішніх додатках, таких як послуги на основі місцезнаходження для маркетингу.

У випадку перебування на відкритому повітрі інформацію про місцезнаходження можна отримати за допомогою глобальної навігаційної супутникової системи (GNSS), а також інших технологій точної навігації, які поєднують дані GNSS та датчиків з картографічною інформацією. Однак існує обмеження у використанні точної інформації про місцезнаходження для внутрішніх корпусів, оскільки сигнал GNSS загалом недоступний. Тому було проведено багато досліджень для отримання точної інформації про положення в приміщенні, особливо для робототехніки та особистої навігації.

Були проведені дослідження на основі різних датчиків, щоб знайти положення в приміщенні, де сигнали GNSS не можуть бути прийняті. Датчик виявлення та дальності світла (LiDAR) є широко використовуваним датчиком для визначення положення в приміщенні, і його ефективність визначення положення була доведена при його злитті з іншими датчиками за допомогою різних фільтрів. Зокрема, картографічне визначення положення за допомогою одночасної локалізації та картографування (SLAM) має перевагу знаходження абсолютного положення. Однак датчик LiDAR в основному використовується для безпілотних наземних транспортних засобів (UGV), робототехніки або безпілотних систем, а не для пішоходів, оскільки важко ефективно прикріпити датчики до людського тіла.

Інший підхід до виявлення місць розташування в приміщенні заснований на різних бездротових сигналах, таких як Wi-Fi, Bluetooth і надширококутний (UWB) в бездротовій локальній мережі (WLAN). Цей метод має недолік у зв'язку з високою вартістю, оскільки вимагає наявності



інфраструктури для всієї будівлі, для якої запитується інформація про місцезнаходження. Хоча він може виміряти абсолютне положення, точність розташування в деяких випадках може бути недостатня.

Методи визначення положення в приміщенні за допомогою магнітних датчиків мають переваги, що абсолютне положення вимірюється без будь-якої додаткової інформації і що ніяка інфраструктура не потрібна в порівнянні з методами позиціонування на основі WLAN. Оскільки деформоване магнітне поле в основному залежить від структури будівлі, його можна використовувати для визначення положення. Однак дані деформованих датчиків мають деякі неясності, оскільки структура будівлі, в цілому, формальна і однорідна. Так, в даний час вивчаються персональні навігаційні алгоритми, що інтегрують ПДР і геомагнітну інформацію. Зокрема, останні дослідження провели оцінку розташування в приміщенні за допомогою PDR і магнітного датчика, за допомогою смартфона.

Спотворення магнітного поля в будівлі - це унікальні характеристики, які залежать від розмірів і розташування металоконструкцій в будівлі, а сталева конструкція всередині будівлі не змінюється після зведення будівлі [33]. Таким чином, магнітна карта-узгодження, яка використовує магнітні спотворення, може бути застосована для визначення місця розташування. Крім того, магнітний курс також можна отримати більш точно, використовуючи кілька тривісних магнітних датчиків, прикріплених до різних частин тіла.

Техніка пом'якшення викидів на основі декількох магнітних датчиків, зважування жорсткості та нормалізації

Метод зіставлення магнітних карт з використанням декількох датчиків має ту перевагу, що можна виявити абсолютне положення та відносно точний курс. При використанні декількох датчиків також очікується, що надмірність відображення і неоднозначність положення можуть бути стримані, оскільки використовується додаткова інформація. Для узгодження магнітних карт в загальному випадку норма вимірюваного магнітного поля порівнюється з нормою збереженої норми за положенням. Оскільки характеристики магнітних спотворень різні залежно від конфігурації датчика, додаткова інформація має істотне значення. Замість використання одновимірних даних для порівняння, для порівняння можна використовувати двовимірну інформацію, що розширює можливість правильного зіставлення карт.

Коли пішоход йде по кімнаті, де побудована карта, магнітні дані датчика порівнюються з усім набором даних на магнітній карті на кожному етапі стійки. Перевіривши ймовірність, можна визначити позицію. Для порівняння

карти магнітного поля і вимірювань функція витрат визначається методом середньоквадратичного відхилення .

Коли для зіставлення карти магнітного поля використовується один датчик, виконується лише порівняння величини. Таким чином, кілька позицій користувача можуть мати подібну величину магнітного поля, а це означає, що може існувати багато винятків. Викиди погано впливають на точність положення, оскільки вони збільшують надмірність позицій та неоднозначність.

Однак, коли використовується кілька магнітних датчиків, для порівняння використовується більше двох норм магнітного поля, з яких складається вектор. При цьому функція витрат повинна бути модифікована.

### **Постановка задачі:**

Представлена структурна схема запропонованого алгоритму MAPDR. Алгоритм MAPDR умовно розділений на три частини: інерціальний навігаційний алгоритм, зіставлення карт магнітного поля з пом'якшенням відхилень та вибіркою важливості, а також розширений фільтр Кальмана. В основному, ставлення, швидкість і положення отримані за допомогою інерціального навігаційного процесу, заснованого на кватерніоні. Заголовок вимірювання з магнітного датчика, вимірювання положення на основі зіставлення магнітної карти та ZUPT використовуються для ЕКФ для виправлення помилок навігації та зміщення датчика.

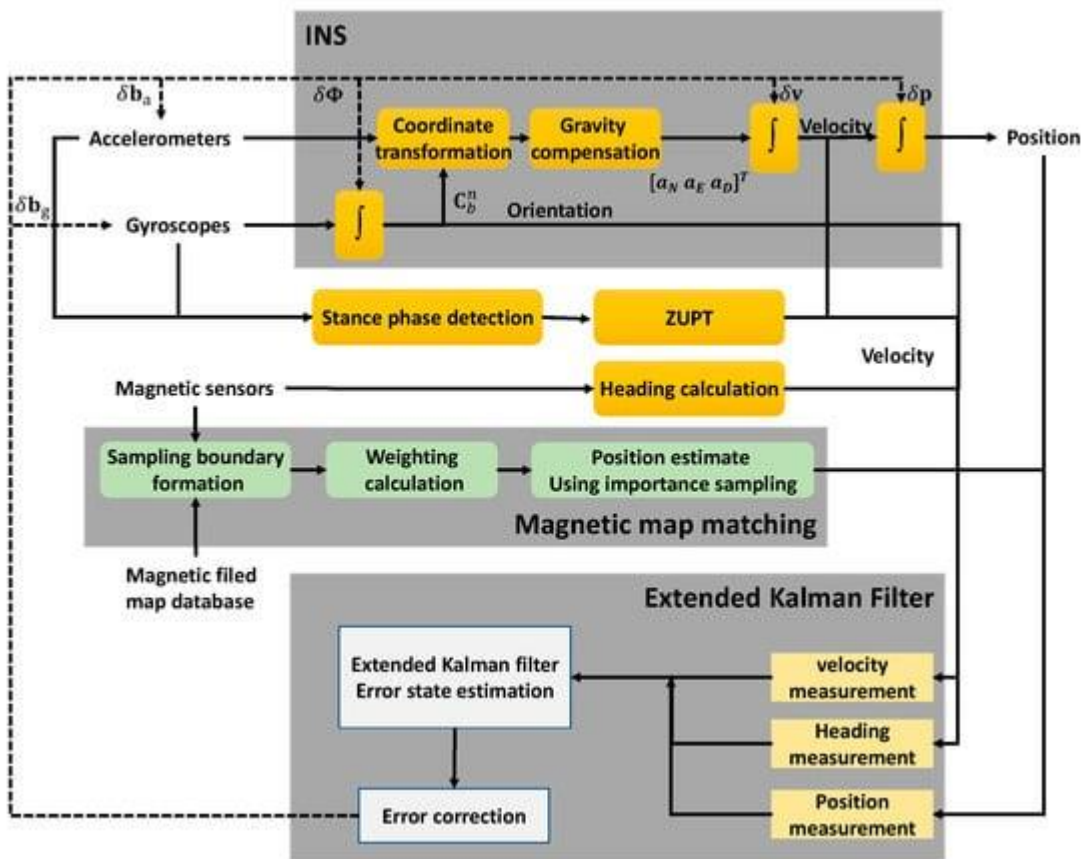


Рисунок 1 - Структурна схема алгоритму підрахунку пішоходів за допомогою зіставлення карт (MAPDR).

У цьому дослідженні було проведено кілька експериментів для перевірки запропонованого алгоритму. В експериментах використовувалися IMU на основі MEMS компанії Xsens Inc. Їх технічні характеристики наведені в таблиці специфікація датчика ([www.xsens.com](http://www.xsens.com)).

Два IMU були встановлені на п'яті шматка взуття і пряжки пояса відповідно. Один датчик кріпився на відстані 5 см над землею, а інший - на 1 м над землею. Всі дані датчиків збиралися на частоті 100 Гц, а виміряні дані датчиків були отримані комп'ютером за допомогою бездротової системи прийому даних компанією Xsens.

Місця експерименту були відібрані ретельно, щоб відобразити багато різних випадків. Експерименти проводилися в двох різних будівлях університету Седжон і в різних місцях, таких як величезні зали, коридори, офіси, ліфтові зали та ескалаторні зали, які мають різне магнітне середовище. Конфігурація магнітних датчиків для побудови карти була такою ж, як і для пішохідних експериментів. Експерименти проводили сім чоловіків і жінок-експериментаторів різного віку. Для складання карти магнітного поля був

встановлений розмір сітки 30 см × 30 см кожна. Конфігурація експериментальної системи представлена на рисунку:

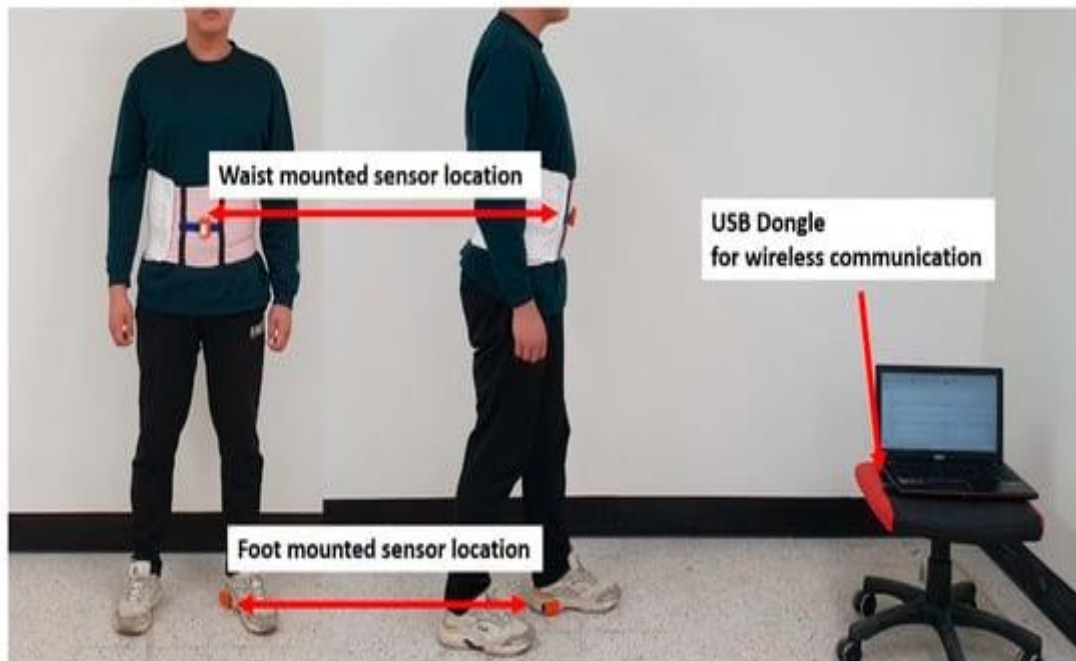
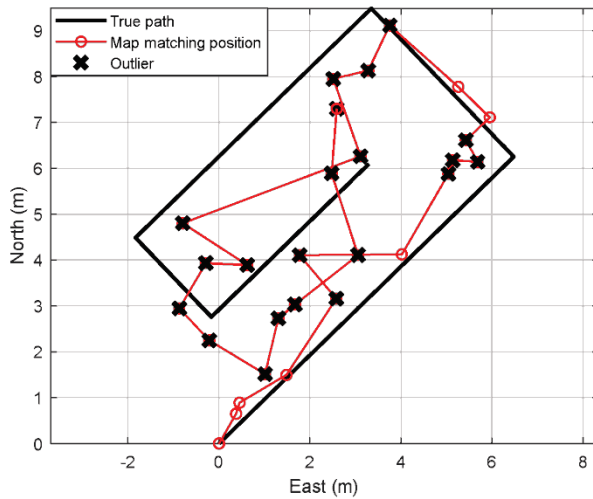
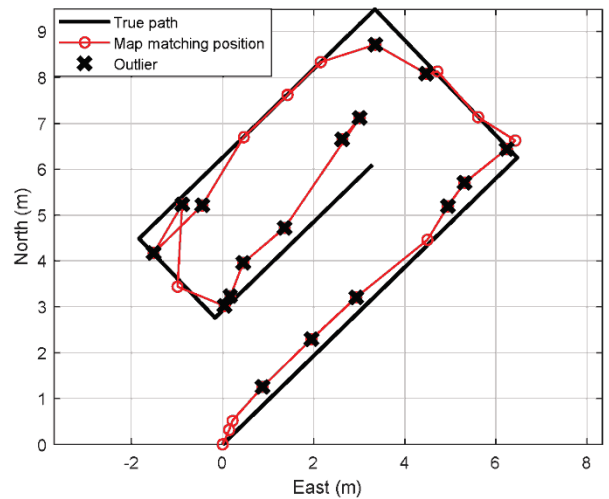


Рисунок 2 - Конфігурація експериментальної системи

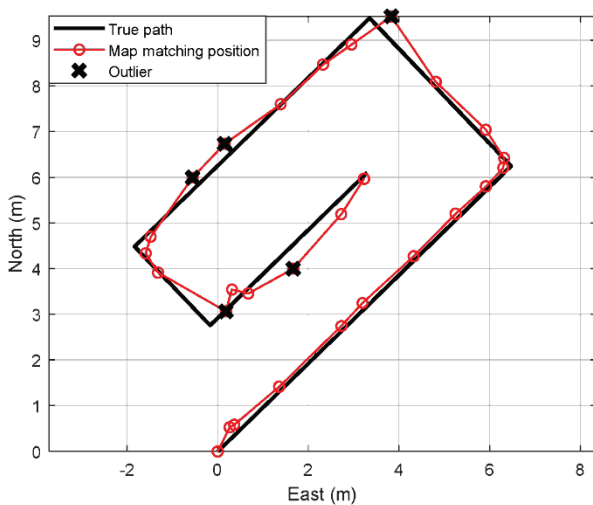
Запропонований алгоритм був застосований, і були проведені експерименти для підтвердження пом'якшення відхилень. Порівняльні результати наведені на рисунку 3, де відображені викиди. Як показано на рисунку 3 а,б, помилки положення у випадках, пов'язаних з одним датчиком, багато в чому обумовлені неоднозначністю і надмірністю. Як показано на рисунку 3 с, при використанні декількох датчиків, хоча похибка значно зменшується і загальний шлях може бути ідентифікований, викиди все ще існують. При застосуванні пом'якшення викидів з нормалізацією, зважуванням жорсткості та відбором проб важливості викиди можуть бути зменшені значно більше.



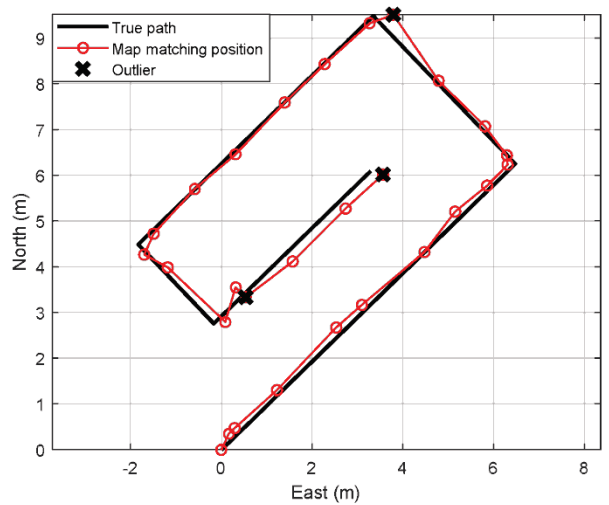
(a)



(b)



(c)



(d)

Рисунок 3 - Порівняння результатів зіставлення карт на основі істинного шляху. (a) Футляр з одним датчиком, встановлений на нозі, (b) одиночний сенсорний корпус, встановлений на поясі, (c) кілька датчиків без зменшення викидів та (d) кілька датчиків із зменшенням відхилень

Кілька експериментальних результатів, проведених на різних ділянках. Випадки з 1 по 4 проводилися у величезних залах з різними джерелами магнітних перешкод, таких як ескалатори, ліфти, залізні двері та залізні стовпи. Експерименти 5 і 6 проводилися в коридорах і кабінетах. Результати показують, що запропонований алгоритм різко пом'якшує відхилення для всіх випадків. Наприклад, 48,7% позицій, що відповідають карті, є винятками у випадку датчиків, встановлених на одній нозі. Коефіцієнт виключення знижується до 8,2% при застосуванні запропонованого алгоритму пом'якшення наслідків.

Концептуальна схема вибірки важливості, межі кандидата та блок-схеми алгоритму. Алгоритм зіставлення магнітних карт з використанням вибірки важливості узагальнюється наступним чином:

## Importance sampling magnetic map matching

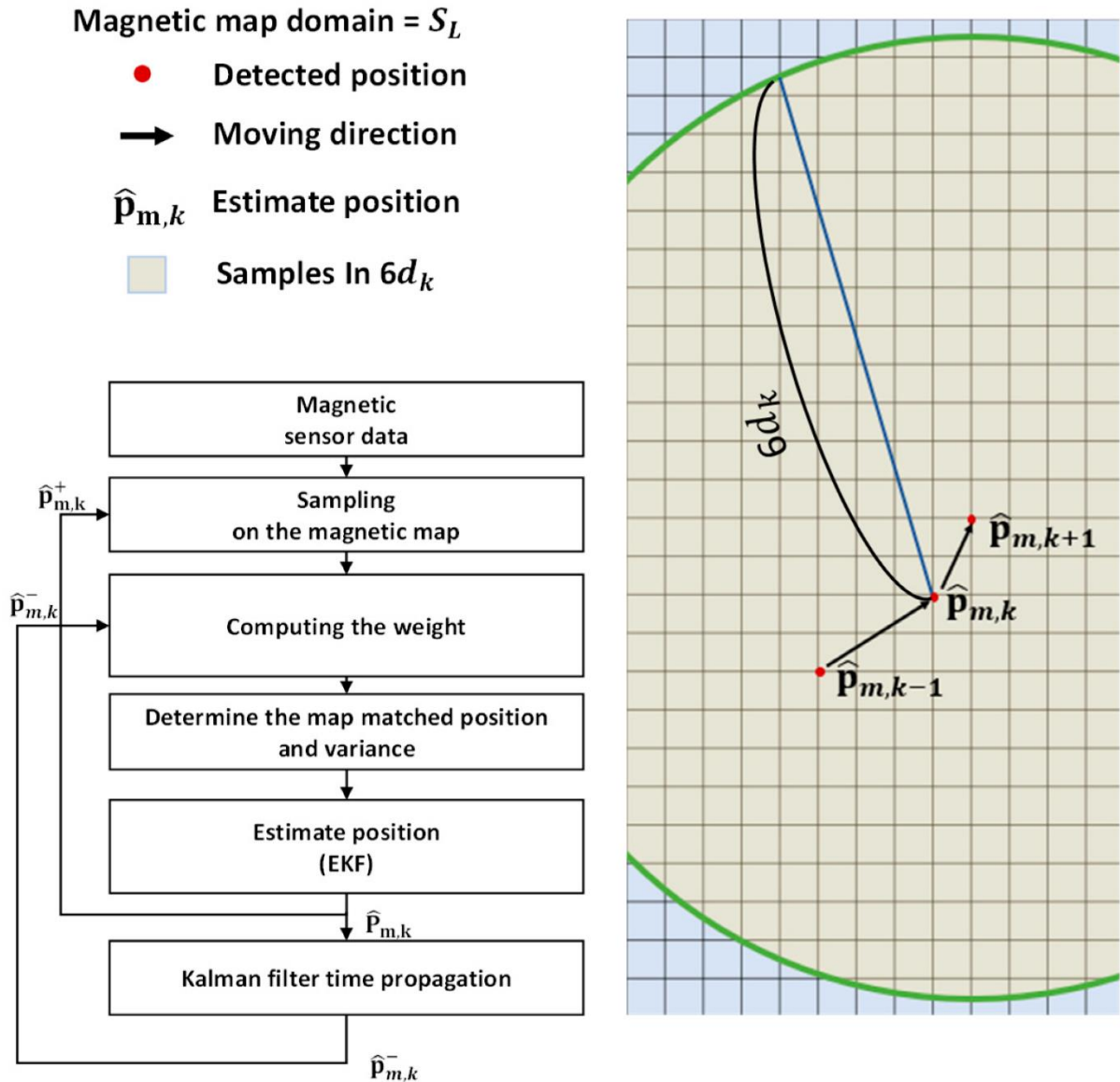


Рисунок 4- Алгоритм зіставлення магнітних карт з використанням вибірки важливості

6. Поняття алгоритму зіставлення магнітних карт з використанням вибірки важливості.

1. Розрахуйте крок від попередніх позицій і встановіть межу кандидата.
2. Виконати відбір проб в межах кандидата.
3. Обчисліть вагу важливості (або функції витрат), що відповідають кожній вибірковій позиції.

4. Використовуючи зважування жорсткості та нормалізацію, знайдіть нормалізований PDF-файл для кожного положення вибірки.
5. Визначте відповідне карті положення та його коваріацію.
6. Використовуйте положення та коваріацію як нове вимірювання для оновлення вимірювань фільтра Кальмана.

Розглянути алгоритм PDR у приміщенні за допомогою магнітної карти, який використовує метод пом'якшення викидів.

### **Перелік питань на захист лабораторної роботи**

1. На підставі чого базується запропонований алгоритм.
2. Що використовується нормалізоване магнітне поле та концепцію зважування жорсткості для зменшення кількості викидів.
3. Чи необхідна вибірка. Для чого використовуються дані.
4. Чи з'являється поліпшення показників визначення позиції.

### **Джерела інформації**

1. <https://www.mdpi.com/1424-8220/19/21/4782>
2. WiFi-Aided Magnetic Matching for Indoor Navigation with Consumer Portable Devices by You Li 1,2,\*, Yuan Zhuang 1, Haiyu Lan 1, Peng Zhang 2, Xiaoji Niu 2 and Naser El-Sheimy <https://doi.org/10.3390/mi6060747>

## ЛАБОРАТОРНА РОБОТА № 5

### Тема «Дослідження алгоритмів навігації мобільних роботів. Вибір та селекція значущих точок по маршруту руху»

Використання мобільних роботів є виправданим у багатьох застосуваннях в неструктурованих середовищах, де вимагається висока ступінь автономії. Така необхідна автономна чи інтелектуальна поведінка обумовлена інтенсивними дослідженнями сьогодення. Управління рухом, наприклад, колісних мобільних роботів є основним предметом досліджень науковців за останні декілька років. При цьому, за умовами відсутності необхідної ступені свободи мобільного робота, достатньо важливою і складною задачею є забезпечення керованості роботом.

Мобільні роботи застосовуються переважно в екстремальних умовах, де люди не можуть знаходитися усередині транспортного засобу, або якщо їх перебування там зв'язане з ризиком для життя. Крім того, в умовах підвищеної небезпеки людина починає здійснювати помилки, знижується їх працездатність і ефективність дій. Виправдано використання мобільних роботів і в звичайних умовах для виконання важких або тривалих монотонних робіт, а також в якості внутрішньоцехового транспорту, на автоматизованих складах, при проведенні земляних робіт

**Постановка задачі:** Мета лабораторної роботи - дослідження алгоритмів навігації мобільних роботів.

**Завдання - Здійснити:**

- класифікацію мобільних роботів;
- провести аналіз та класифікацію способів управління мобільних роботів;
- провести класифікацію підзадач стабілізації мобільного робота;
- провести аналіз моделей управління колісних роботів;
- вибрати значущі точки по маршруту руху;
- провести селекцію значущих точок по маршруту руху



## Хід роботи

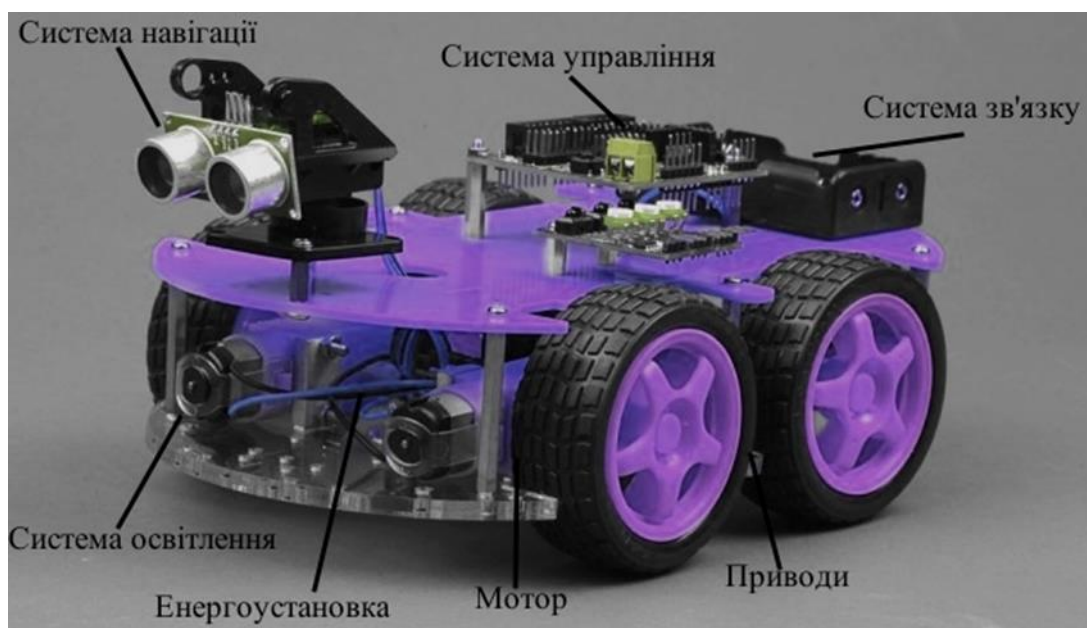


Рисунок 1 - Узагальнена структура мобільного робота

Таблиця 1 - Класифікація мобільних роботів

Група	Базові вимоги	Базовий варіант використання
Спеціального призначення	Компактність, безшумність	Автономне, один канал передачі відеоінформації
Для військових і воєнізованих застосувань	Надійність, простота в управлінні, стандартизоване корисне навантаження	У складі розвідувальних, ударних і охоронних комплексів, багатоканальні системи передачі різноманітних даних
Для екстремальних ситуацій, наукових досліджень	Стійкість до несприятливих зовнішніх впливів, універсальність по відношенню до корисної бортовий навантаженні	Автономне, багатоваріантність реалізації каналів передачі даних
Для спортивних, промислових і побутових застосувань	Простота в управлінні, економічність, надійність	Автономне, один канал передачі відеоданих

## Локалізація та навігація

Локалізація - це основна здатність навігації. Існує два види локалізації: відстеження позиції щодо початкового положення робота, і глобальна локалізація, яка знаходить його глобальні координати.

Оптична локація - галузь науки і техніки, що забезпечує отримання відомостей про об'єкти шляхом прийому і аналізу поширюваних від них електромагнітних хвиль оптичного діапазону. В сьогоdnішній час поняття "локація" далеко виходить за історичні термінологічні рамки. Так як, латинське слово "локус" означає місце, а "локація" - визначення місця розташування.

Два найпоширеніші підходи до глобальної локалізації - орієнтири та показники датчиків. Орієнтир визначається як відмітна особливість довкілля. Робот ідентифікує та зберігає орієнтири, які зустрічаються на карті. Робот локалізується, коли новий орієнтир відповідає збереженому орієнтиру. Підходи на основі орієнтирів поширені у роботів, які використовують датчики на основі зору. Робот може використовувати масштабні інваріантні особливості навколишнього середовища як орієнтири. Функції масштабування не змінюються під час зчитування інформації з зображення або обертання. Сукупність відомостей про об'єкти, одержуваних шляхом прийому і аналізу електромагнітних хвиль оптичного діапазону, називають оптичною інформацією.

Оптичну інформацію можна розділити на координатну і некоординатну. Під координатної інформацією розуміють відомості про місцезнаходження об'єкта в будь-який момент часу спостереження і про характер зміни місця розташування (дальність, кутові координати, похідні координати, траєкторні параметри і т.д.). Отримання координатної інформації пов'язано з поданням об'єкта локації у вигляді фізичної точки.

Під некоординатною інформацією розуміють сукупність ознак, притаманних об'єкту незалежно від положення його центру мас в просторі і характеризують індивідуальні особливості об'єкта, які виходять за рамки його подання у вигляді фізичної точки.

До некоординатної інформації можна віднести відображальні характеристики об'єкта (залежні від матеріалу його поверхні, розмірів, форми

та інших параметрів), зображення, параметри обертального руху об'єкта навколо центру мас і ін.

#### Класифікація навігаційних систем

До навігаційних систем відносяться технічні засоби, призначені для автоматичного дистанційного вимірювання координат об'єктів маніпулювання. Специфічні вимоги до моделей навігаційних систем впливають з їх принципів дії і особливостей функціонування.

Навігаційні системи мобільних роботів поділяються на:

1) Системи вимірювання кутових координат (телевізійна кутомірна система, оптична кутомірна система зі скануванням променя, індукційна кутомірна система, радіаційна кутомірна система);

2) Системи вимірювання дальності і швидкості (акустична далекомірна система, лазерна далекомірна система, радіаційна далекомірна система);

3) Системи пошуку і виявлення (оптична система пошуку по кутових координатах, акустична система пошуку по дальності).

**Вирішити** питання огляду сучасних технічних вітчизняних рішень в області оптичних систем навігації . Завдання навігації із застосуванням систем технічного зору на сьогоднішній день. Розглянути технічні переваги і недоліки алгоритмів навігації мобільних роботів.

#### **Перелік питань на захист лабораторної роботи**

1. Дослідження алгоритмів навігації мобільних роботів
2. Основні моменти вибору та селекції значущих точок по маршруту руху.

## ПРАВИЛА ОФОРМЛЕННЯ ПОЯСНЮВАЛЬНОЇ ЗАПИСКИ

1. Текстовий та графічний матеріали записки друкують комп'ютерним способом на одному боці односторонніх білих аркушів формату А4 (розмір 210 x 297 мм) через 1,5 міжрядковий інтервал, текст вирівнюють по ширині аркуша. Текстовий редактор – Word з пакета Microsoft Office, Open Office Writer, Star Office Writer та ін. Шрифт – Times New Roman Cyr, 14.

### 2. Оформлення ілюстрацій

Усі ілюстрації в пояснювальній записці (креслення, схеми, фотографії, діаграми, графіки) називають рисунками.

Кількість ілюстрацій має бути достатньою для пояснення тексту, який викладається. Ілюстрації потрібно розмішувати як по тексту записки (якомога ближче до відповідних частин тексту), так і в кінці його або наводити в додатках. Ілюстрації належить виконувати у відповідності до вимог стандартів ЄСКД і ЄСПД за допомогою різних графічних редакторів та систем автоматизованого проектування.

Усі ілюстрації послідовно нумерують у межах розділу арабськими цифрами. Номер ілюстрації складається з номера розділу і порядкового номера ілюстрації, наприклад, «Рис 2.5 Граф алгоритму». Посилання на ілюстрації подають так: «... на рис. 2.5 ...». Повторне посилання на ілюстрацію наводять із скороченням слова «дивись», наприклад, «... див. рис. 2.5 ...». Допускається нумерація ілюстрацій у межах лабораторної роботи.

Ілюстрації повинні мати назву, яку розміщують під ілюстрацією в одному рядку з її номером, наприклад, «Рис. 3.2. Схема». За потреби під назвою ілюстрації записують пояснювальні дані

Розмір шрифту всіх без винятку надписів у рисунках має бути таким самим, як і в тексті пояснювальної записки.

Ілюстрації розміщують так, щоб їх можна було розглядати, не повертаючи аркуш або повертаючи його за ходом стрілки годинника.

## СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ

### Основна література

1. Вовк С.М., Гнатушенко В.В., Бондаренко М.В. Методи обробки зображень та компютерний зір // Навчальний посібник. Д.: ЛПРА. 2016. – 148 с
2. Методи та системи штучного інтелекту: навч. посіб. // укл. Д.В. Лубко, С.В. Шаров. – Мелітополь: ФОП Однорог Т.В., 2019. – 264 с.
3. Ю.А. Ніцук, О.М. Семчак, Шаріпова І.В. Визначення шляхів зменшення похибок розрахунків координат бортовими ЕОМ автономного рухомого об'єкту для реалізації алгоритмів SLAM навігації // Збірник наукових праць Житомирський військовий інститут імені С. П. Корольова. – Житомир : ЖВІ, 2020.- № 27 (4). – С. 38 – 49.
4. Н.В. Морзе, Л.О. Варченко-Троценко, М.А. Гладун. Основи робототехніки: навчальний посібник // Н.В. Морзе, Л.О. Варченко- Троценко, М.А. Гладун. – Кам'янець-Подільський : ПП Буйницький О.А., 2016. – 184 с.
5. Ковальов Ю.А. Проектування промислових роботів та маніпуляторів // Ю.А. Ковальов, С.О. Кошель, О.П. Манойленко. – Київ: Видавництво "Центр учбової літератури", 2019. – 256 с.

### Додаткова література

1. Глухов О. В. Вивчення властивостей мікроконтролерів і електронних систем на базі платформи Ардуіно : навч. посіб. / О. В. Глухов, О. О. Кравчук, Є. В. Левченко ; М-во освіти і науки України, Харків. нац. ун-т радіоелектроніки. – Харків : ХНУРЕ, 2019. – 192 с.
2. <https://docs.nvidia.com/isaac/apps/carter/gmapping/doc/index.html>

## Додаток А

Одеський національний університет імені І. І. Мечникова

---

(повне найменування вищого навчального закладу)

Факультет математики, фізики та інформаційних технологій

---

(повне найменування інституту/факультету)

Кафедра комп'ютерних систем та технологій

---

(повна назва кафедри)

Звіт з лабораторної роботи № \_\_\_\_

з дисципліни «Комп'ютерний зір та навігація робототехнічних систем»

Тема: «\_\_\_\_\_»

Виконав (-ла) : студент (ка) денної форми навчання  
спеціальності 123 «Комп'ютерна інженерія»  
курс \_\_\_\_ гр. \_\_\_\_

Прізвище Ім'я \_\_\_\_\_

Керівник (доц.) кафедри КСТ  
Прізвище Ім'я \_\_\_\_\_

Одеса – 2023

ДЛЯ ПОДАТОК

*Навчальне видання*

# **КОМП'ЮТЕРНИЙ ЗІР ТА НАВІГАЦІЯ РОБОТОТЕХНІЧНИХ СИСТЕМ**

## **ЗМІСТОВНИЙ МОДУЛЬ 1 НАВІГАЦІЯ МОБІЛЬНИХ РОБОТІВ**

Методичні вказівки  
до лабораторних занять для студентів  
першого (бакалаврського) рівня вищої освіти  
галузі знань 12 Інформаційні технології

*Укладач*

**Шаріпова Ільнара Вільївна**

*В авторській редакції*

Затвердж. авт. 2,32 умов. друк. арк. Шрифт Times New Roman.  
Системні вимоги: операційна система сумісна з програмним забезпеченням для читання  
файлів формату PDF. Зам. № 1123-2701.

Видавництво та друк: ОЛДІ+  
65101, Україна, м. Одеса, вул. Інглезі, 6/1  
Свідоцтво ДК № 7642 від 29.07.2022 р.

---

Тел.: +38 (098) 559-45-45,  
+38 (095) 559-45-45, +38 (093) 559-45-45  
Для листування: 65101, Україна, м. Одеса, вул. Інглезі, 6/1  
E-mail: office@oldiplus.ua

