POLYTECH PARIS-SACLAY

université PARIS-SACLAY

U.PORTO

FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

Perception and Mapping

Extended Kalman Filter

Simultaneous Localization and Mapping

Author: Moraless Philius

Abstract – This document provides an overview of my project focused on robot navigation. In the initial phase, we utilized the Extended Kalman Filter (EKF) algorithm to estimate the robot's position. Subsequently, we employed the Simultaneous Localization and Mapping (SLAM) algorithm to not only determine the robot's position but also map the locations of surrounding beacons. The project aims to combine these two approaches to enable the robot to autonomously navigate while constructing a map of its environment.

I. Introduction

The autonomous navigation of robots in unknown environments poses a major challenge in the field of robotics. To enable a robot to move autonomously, it must be able to determine its position and map its surroundings. In this project, we will explore two essential methods to achieve this: the Extended Kalman Filter (EKF) and Simultaneous Localization and Mapping (SLAM).

The EKF is an algorithm for estimating the state of a dynamic system, which is crucial for tracking the position of a robot. SLAM is an advanced technique that allows a robot to map its environment while simultaneously localizing itself in real-time. Our project aims to combine these two approaches to enable a robot to move autonomously while constructing a map of its environment.

We will divide this project into three distinct phases: firstly, we will conduct a state-of-the-art review, highlighting the motion equations of the robot. Then, in the second phase, we will implement an algorithm to determine the robot's position using multiple assumptions about sensor data. Finally, in the last phase, we will focus on the implementation of an algorithm to localize the robot and map its environment.

II. State of art

A. Statement

The position of the robot on the plane is defined by the coordinates (x, y) and its orientation is defined by θ. The robot can measure its linear and angular velocities, respectively v and ω and consider the motion can be modelled by

$\dot{x} = v\cos\theta$

$\dot{y} = v\sin\theta$

$\dot{\theta} = \omega$

v and ω measurements are affected by noises which are characterized by independent normal distributions with zero mean and variances $\sigma_v^2 = 0.5$ and $\sigma_\omega^2 = 0.05$. The robot has a laser system that can measure its distance (r) and bearing (ψ, in the robot own reference frame) to landmarks located in the operation area. These measurements are affected by noises characterized by independent normal distributions with zero mean and variances $\sigma_r^2 = 0.5$ and $\sigma_\psi^2 = 0.5$

B. Equations of motion

Our goal is to determine the position. To do this, we will use the equations from the previous section to obtain the functions x_{k+1}, y_{k+1}, and θ_{k+1}. In the first step, we will assume that θ depends on time: $\theta_{k+1} = \theta_k + \omega * dt$

$$f\begin{cases} x_{k+1} = x_k + \dfrac{v_k}{\omega_k}[\sin[\theta_k + \omega_k * dt] - \sin(\theta_k)] \\ y_{k+1} = y_k + \dfrac{v_k}{\omega_k}[-\cos[\theta_k + \omega_k * dt] + \cos(\theta_k)] \\ \theta_{k+1} = \theta_k + \omega_k * dt \end{cases}$$

Now we assume that θ does not depend on time.

$$f\begin{cases} x_{k+1} = x_k + v_k\cos(\theta_k) * dt \\ y_{k+1} = y_k + v_k\sin(\theta_k) * dt \\ \theta_{k+1} = \theta_k + \omega_k * dt \end{cases}$$

We observe two sets of equations to express the robot's position. This choice is crucial because, for example, when the robot does not detect any obstacles, its position will be estimated solely from the chosen set of equations, regardless of its environment. In our case, we have opted for model 2 for the sake of simplicity, which is suitable for our application.

III. Implementation Extended Kalman Filter

Algorithm 1 Extended Kalman Filter $(x_{k-1}, P_{k-1}, u_k, z_k)$

1: $F_x = \begin{bmatrix} 1 & 0 & -v_k \sin(\hat{\theta}_{k-1})\, dt \\ 0 & 1 & v_k \cos(\hat{\theta}_{k-1})\, dt \\ 0 & 0 & 1 \end{bmatrix}$

2: $F_u = \begin{bmatrix} \cos(\hat{\theta}_{k-1})\, dt & 0 \\ \sin(\hat{\theta}_{k-1})\, dt & 0 \\ 0 & dt \end{bmatrix}$

3: $Q = \begin{bmatrix} \sigma_v^2 & 0 \\ 0 & \sigma_\omega^2 \end{bmatrix}$

4: $\hat{x}_k = x_{k-1} + F_u u_k$

5: $\hat{P}_k = F_x P_{k-1} F_x^T + F_u Q F_u^T$

6: $y = (z_k - \hat{z}_k)$

7: $R = \begin{bmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_\psi^2 \end{bmatrix}$

8: $S = H \hat{P}_k H^T + R$

9: $K = \hat{P}_k H^T S^{-1}$

10: $\hat{x}_k = \hat{x}_k + K y_k$

12: $\hat{P}_k = \hat{P}_k - K H \hat{P}_k$

13: $x_k = \hat{x}_k$

14: $P_k = \hat{P}_k$

15: **Return** x_k, P_k

Algorithm EKF

IV. Implementation EKF-SLAM

Algorithm 2 EKF-SLAM $(x_{k-1}, P_{k-1}, u_k, z_k)$

1: $F_x = \begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 1 & 0 & \cdots & 0 \end{bmatrix}$

2: $\hat{x}_k = x_{k-1} + F_x \begin{bmatrix} v_k \cos(\hat{\theta}_{k-1}) * dt \\ v_k \sin(\hat{\theta}_{k-1}) * dt \\ \omega_k * dt \end{bmatrix}$

3: $G = I + F_x^T \begin{bmatrix} 0 & 0 & v_k \cos(\hat{\theta}_{k-1}) * dt \\ 0 & 0 & v_k \sin(\hat{\theta}_{k-1}) * dt \\ 0 & 0 & 0 \end{bmatrix} F_x$

4: $Q = \begin{bmatrix} \sigma_v^2 & 0 \\ 0 & \sigma_\omega^2 \end{bmatrix}$

5: $\hat{P}_k = GP_{k-1}G^T + F_x Q F_x^T$

6: $R = \begin{bmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_\psi^2 \end{bmatrix}$

7: for all observed features do

8: $j = c_i$

9: if landmark j never seen before

10: $\begin{pmatrix} \hat{x}_j \\ \hat{y}_j \end{pmatrix} = \hat{x}_k[0{:}2] + \begin{pmatrix} r_i \cos(\psi_i + \hat{x}_k[2]) \\ r_i \sin(\psi_i + \hat{x}_k[2]) \end{pmatrix}$

11: end if

12: $\delta = \begin{pmatrix} \hat{x}_j - \hat{x}_k[0] \\ \hat{y}_j - \hat{x}_k[1] \end{pmatrix}$

13: $\hat{z}_i = \begin{pmatrix} \sqrt{\delta^T \delta} \\ atan2(\delta_y, \delta_x) - \hat{x}_k[2] \end{pmatrix}$

14: $H_i = \frac{1}{\sqrt{\delta^T \delta}} \begin{pmatrix} -\sqrt{\delta^T \delta}\delta_x & -\sqrt{\delta^T \delta}\delta_y & 0 & \sqrt{\delta^T \delta}\delta_x & \sqrt{\delta^T \delta}\delta_y \\ \delta_y & -\delta_x & -\sqrt{\delta^T \delta} & -\delta_y & \delta_y \end{pmatrix} F_x^i$

15: $S_i = H_i \hat{P}_k H_i^T + R$

16: $K_i = \hat{P}_k H_i^T S_i^{-1}$

17: $\hat{x}_k = \hat{x}_k + K_i(z_i - \hat{z}_i)$

18: $\hat{P}_k = (I - K_i H_i)\hat{P}_k$

19: end for

20: $x_k = \hat{x}_k$

21: $P_k = \hat{P}_k$

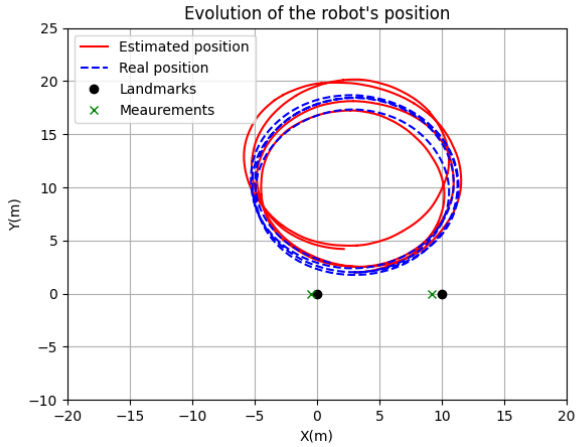
22: **Return** x_k, P_k

Algorithm EKF-SLAM

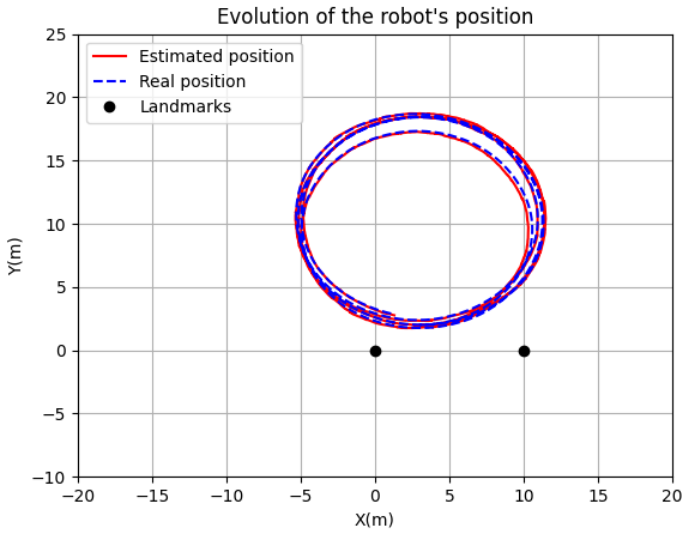
V. Results and discussions

A. Task 1

Now, we will examine our algorithms by doing multiple tests and discussing the results. In the first test, we assume that there are two landmarks located at (0, 0) and (10, 0), and the initial pose of the robot is known. We use the data from the “data1.txt” file.



For this initial test, we perform an estimation without correction to observe the effect of the Kalman filter we will use. We notice that the robot's position estimation is not in line with the true value.

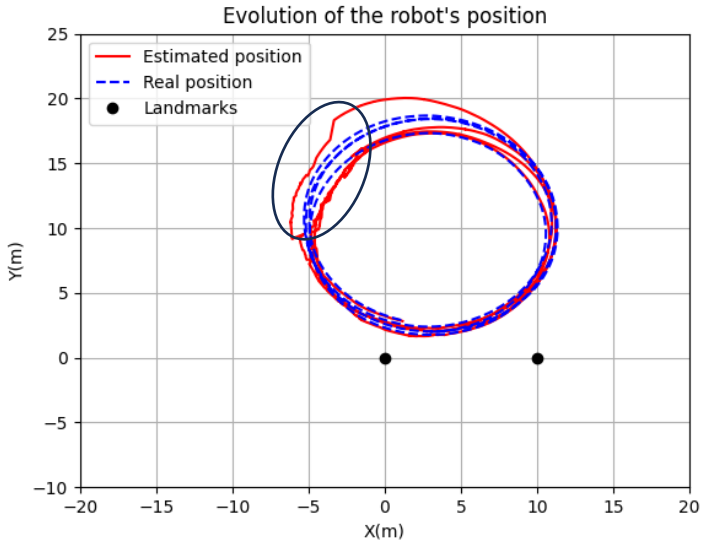


In the first test, we observe that our robot closely follows the actual trajectory. However, when it makes a left turn, we notice a slight deviation from the actual path. This initial deviation is quickly corrected, and the robot subsequently follows the actual trajectory more accurately. This behaviour can be explained by the uncertainties and measurement errors inherent in our model and sensors. The robot adjusts its trajectory as it receives new information and as uncertainties are reduced, allowing it to converge toward the expected path.

Furthermore, the first test results in a low (**0.023**) Mean Squared Error, indicating that there are very few errors in our estimation. This demonstrates the effectiveness of our algorithm in accurately tracking the robot's trajectory while minimizing errors.

B. Task 2

In the second test, we will restrict the vision of our laser sensor. Now, when no obstacles are detected within the sensor's field of view (± π/4), it returns a distance value of zero. We modify our algorithm so that when no obstacles are detected, the correction step will not be executed. In other words, the robot will follow the motion equation we have imposed. We use the data from the “data2.txt” file.



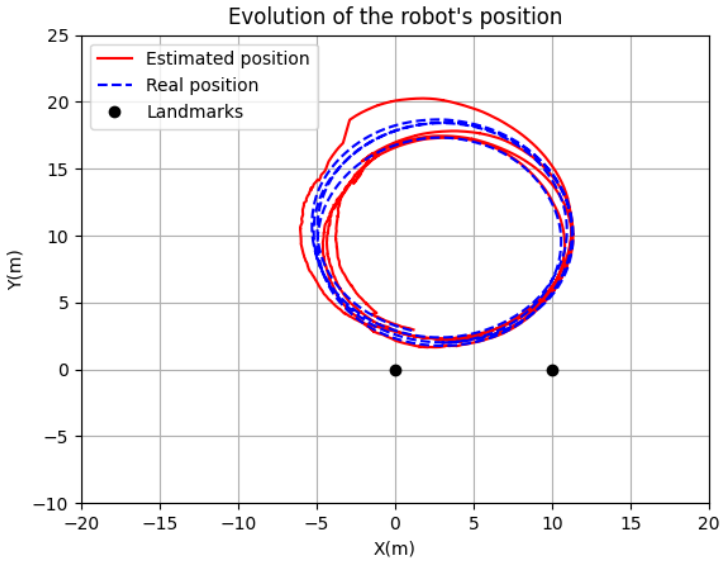
In the second test, we observe that reducing the field of vision of our sensor has a significant impact on the robot's behaviour. The robot now tends to follow the motion

equation without making major corrections. Reducing the field of vision is a substantial modification since the robot can no longer detect obstacles outside of this restricted field. Consequently, it does not receive correction information, affecting the accuracy of its trajectory. The observed deviations are a consequence of this perceptual limitation.

It is also worth noting that in this test, the Mean Squared Error of the error has increased tenfold compared to the previous test (**0.21**). This significant increase in MSE indicates that errors have propagated more due to the limited field of vision. The robot is less capable of correcting its trajectories, resulting in a larger error in its estimations.

C. Task 3

Now assume that besides the field of view constraint the laser system only returns the range and distance to the closest landmark, not proving information to which landmark the measurement refers. We use the data from the “data3.txt” file.

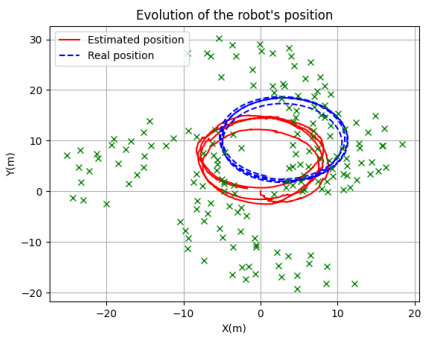
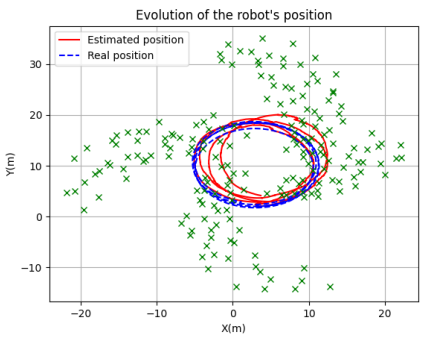


In the third test, we introduced an additional constraint: the laser sensor provides measurements of distance and orientation to the nearest landmark without specifying the identity of that landmark. To achieve this, we calculated the Mahalanobis distance for both landmarks and selected the one with the minimum Mahalanobis distance. This approach proved to be simpler but introduces uncertainties, as there can be ambiguities in certain situations. Nevertheless, we observed that the overall trajectory closely

resembles that of test 2, although occasional errors may occur when identifying the landmarks is uncertain. In terms of performance, the Mean Squared Error has slightly increased (**0.24**) compared to the previous test.

D. Task 4

For our final test, we use the EKF-SLAM algorithm to enable the robot to move while building its map. We still assume that the robot has a limited field of view. We use the data from the "data4.txt" file.

Pose initial (0,0,0)	Pose initial (3,2,0)
	

Tab1. Comparaison

During the last task, our goal was to test our trajectory estimation algorithm under more complex conditions, particularly by disregarding the initial positions of landmarks and reference points. However, we observed that our approach lacks reliability, especially when initiated from a starting position of (0, 0, 0). The inaccuracies observed in the trajectory can be attributed to inherent measurement errors in our sensors, which may be exacerbated in the absence of prior information about the environment.

We also observed that the execution time was slower than the previous tests because our matrices and vectors were increasingly larger. Regarding the detected landmarks, we noticed that there are four groups of landmarks, leading us to hypothesize that there are 4 true landmarks.

VI. Conclusion

In summary, this project tackled the fundamental challenge of autonomous robot navigation using the Extended Kalman Filter (EKF) and Simultaneous Localization and Mapping (SLAM) approaches. We developed motion equations to model the robot's behaviour and implemented algorithms to estimate its position under various constraints. Our tests and experiments explored scenarios with limited sensor visibility and noise.

Through this project, we gained valuable insights into the intricacies of robotic localization and mapping, which have broad applications in fields such as autonomous robotics, indoor navigation, and environmental monitoring. By combining EKF and SLAM, we enabled the robot to move autonomously while mapping its environment, exemplifying the potential for advanced robotics and automation in the future.

The implementation of the EKF-SLAM algorithm, particularly in scenarios with unknown landmarks, brought forth intriguing results. The observation regarding the existence of four distinct landmark groups raises questions about the environment's underlying structure. This insight can be further explored and refined in future research or applications.