Sistemas Autónomos / Autonomous Systems

# [S5] – FastSLAM using UWB

João Canas
joaocarloscanas@tecnico.ulisboa.pt
MEAero

Manuel Rosa
manuel.d.rosa@tecnico.ulisboa.pt
MEAero

Pedro Trindade
pedro.m.dias.trindade@tecnico.ulisboa.pt
MEAero

Pedro Carmo
pedro.carmo@tecnico.ulisboa.pt
MEEC

Instituto Superior Técnico, Universidade de Lisboa

## Introduction

The principle of SLAM consists in using a relative localization method to estimate the position and an observation model to map the environment in relation to the agent's location. One of the most recent methods is FastSLAM, which represents a stochastic approach, where a particle filter is implemented to estimate the most likely state. The observations given to the SLAM algorithm may come from a variety of sensors commonly used in mapping problems. In this work, an alternative sensor for observations was studied, the ultra wideband (UWB) range-finder. The implemented FastSLAM solution implements a stochastic approach using the algorithm of a particle filter.

## Algorithm

For each particle the joint distribution of a pose and a landmark given the evidences (observations and movements) is factorized, based on Rao-Blackwellization, and re-written as the product of the pose probability given the evidences (path posterior) and the map probability given the position and the observations (map posterior):

$$p(x_{0:t}, l_{1:M} | z_{1:t}, u_{1:t}) = p(x_{0:t} | z_{1:t}, u_{1:t}) p(l_{1:M} | x_{0:t}, z_{1:t})$$

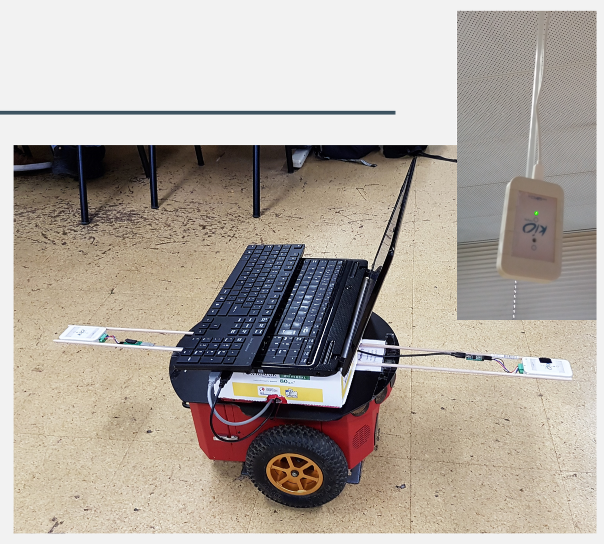
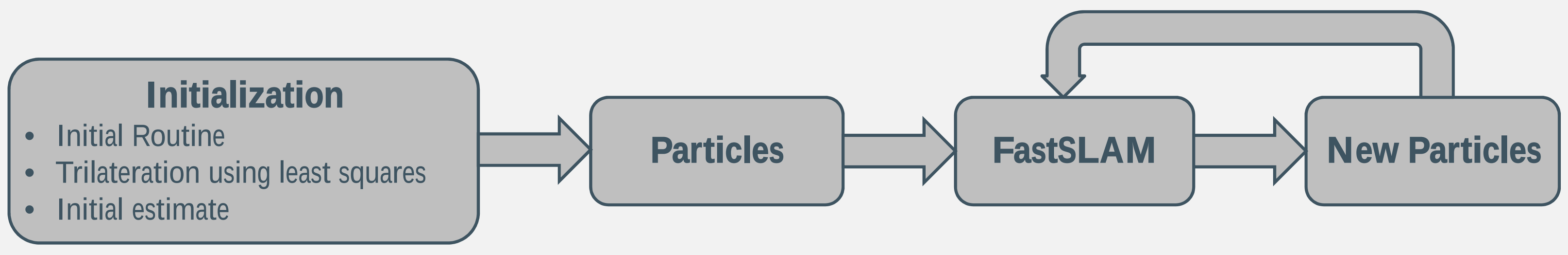


Fig. 1 - Pioneer robot setup with UWB

Each Particle has:

- A representation of a possible robot pose;
- An Extended Kalman Filter (EKF) is used for each landmark representation;

Every time a new landmark is found, it is used a trilateration algorithm to obtain initialization values for FastSLAM algorithm.




## Implementation

- Uses python and ROS;
- Begins by generating a set of N particles, in the origin of the odometry frame;
- Receives an input from odometry and computes its moved distance, at each update step;
- Adds noise to each value so that each particle receives a different input;
- An EKF is ran for each particle to estimate landmark's position;
- UWB only gives distances, requiring an alternative approach to bypass observability and EKF non-convergence problems:

$$o_{L_j} = \begin{bmatrix} x_{p_i} + d\cos(\theta_{p_i}) \\ y_{p_i} + d\sin(\theta_{p_i}) \\ z_{p_i} \end{bmatrix} + \begin{bmatrix} d_j \sin(\phi_j)\cos(\alpha_j) \\ d_j \sin(\phi_j)\sin(\alpha_j) \\ d_j \cos(\phi_j) \end{bmatrix}$$

*with its variables represented in Figure 3*

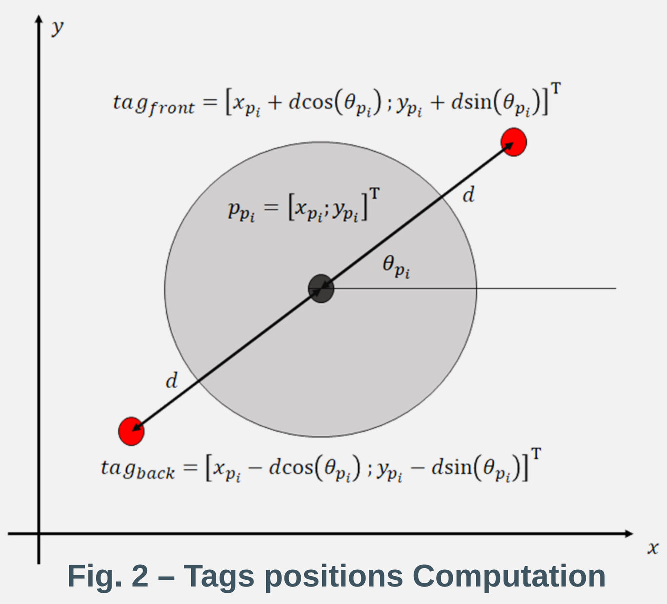


Fig. 2 – Tags positions Computation

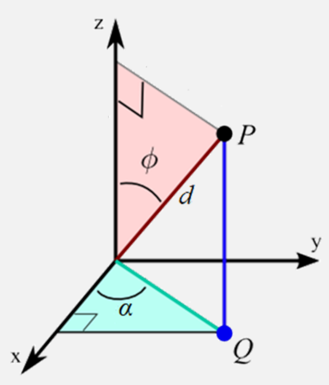
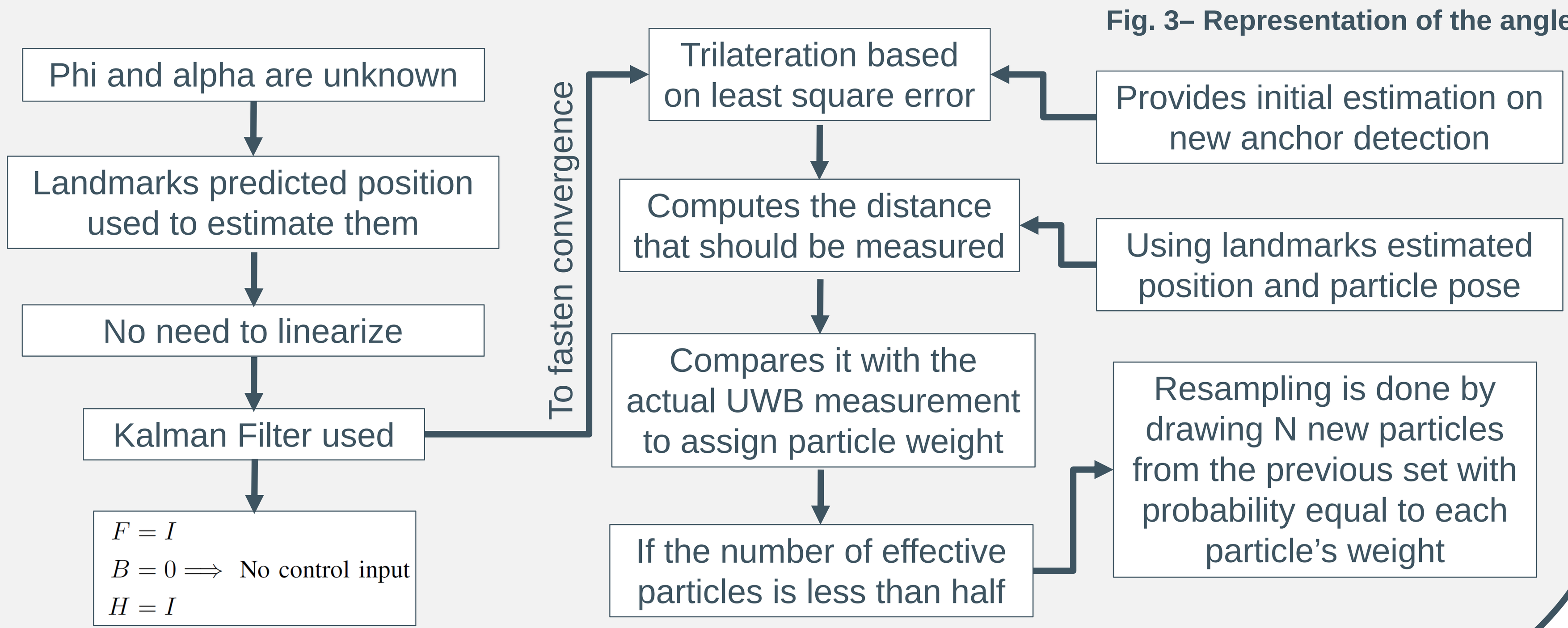


Fig. 3– Representation of the angles




## Results

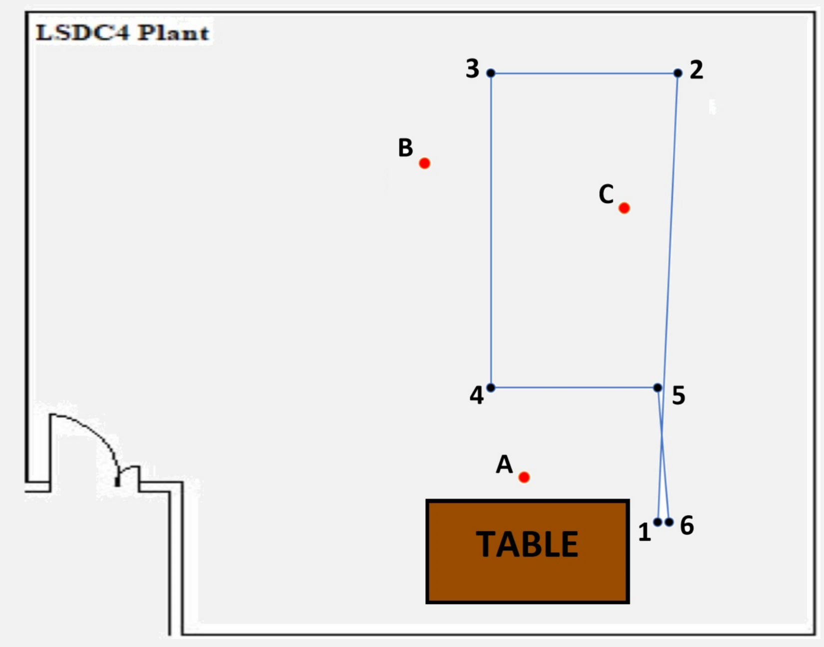
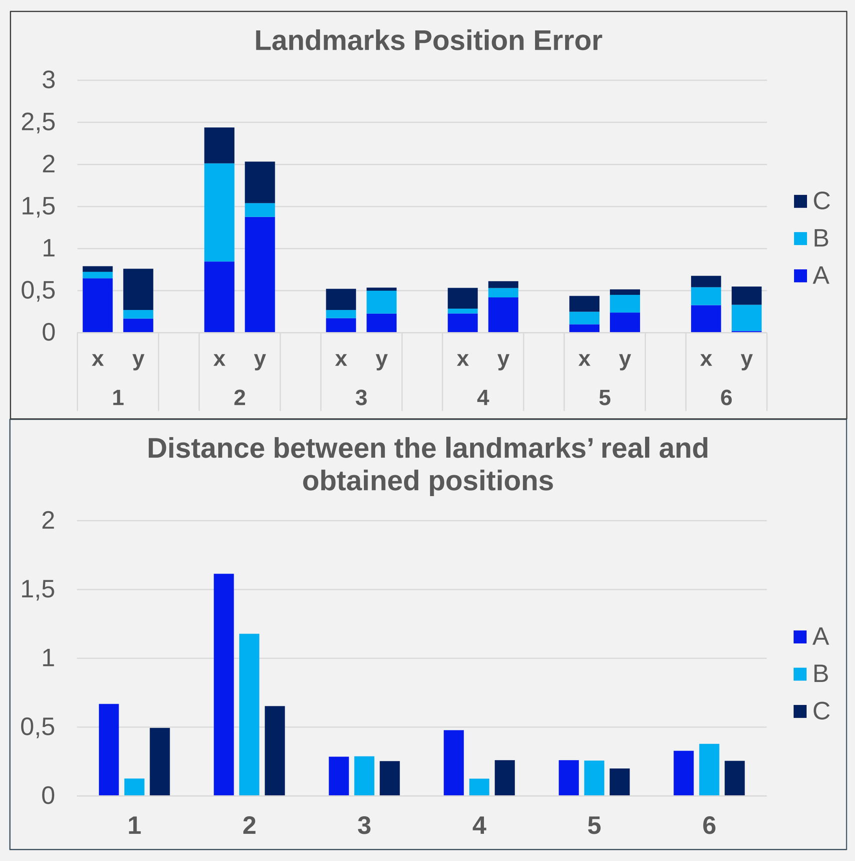


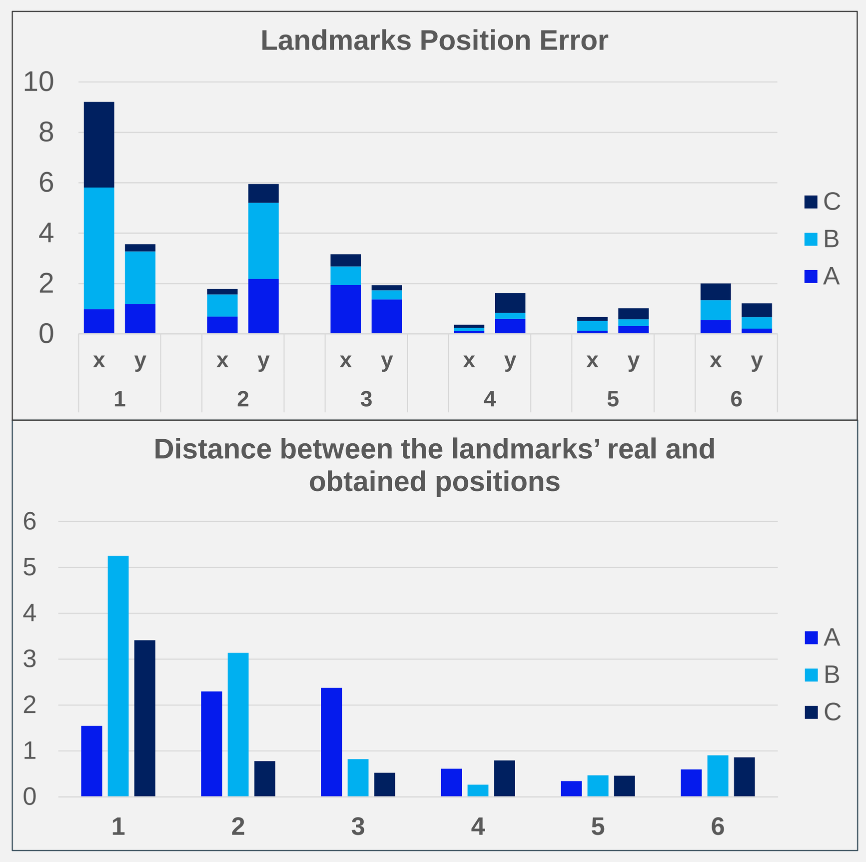

Fig. 4 – Illustration of the robot path and measurement points

To obtain some data about the performance of our algorithm, several experiments were made, in order to validate it. Here are shown the results obtained for one of those experiments, in which the robot followed a path, and some measurements of its pose were made, for posterior comparison. The path and those points are shown in the Fig. 4. Using this conditions, 2D and 3D algorithms were tested.

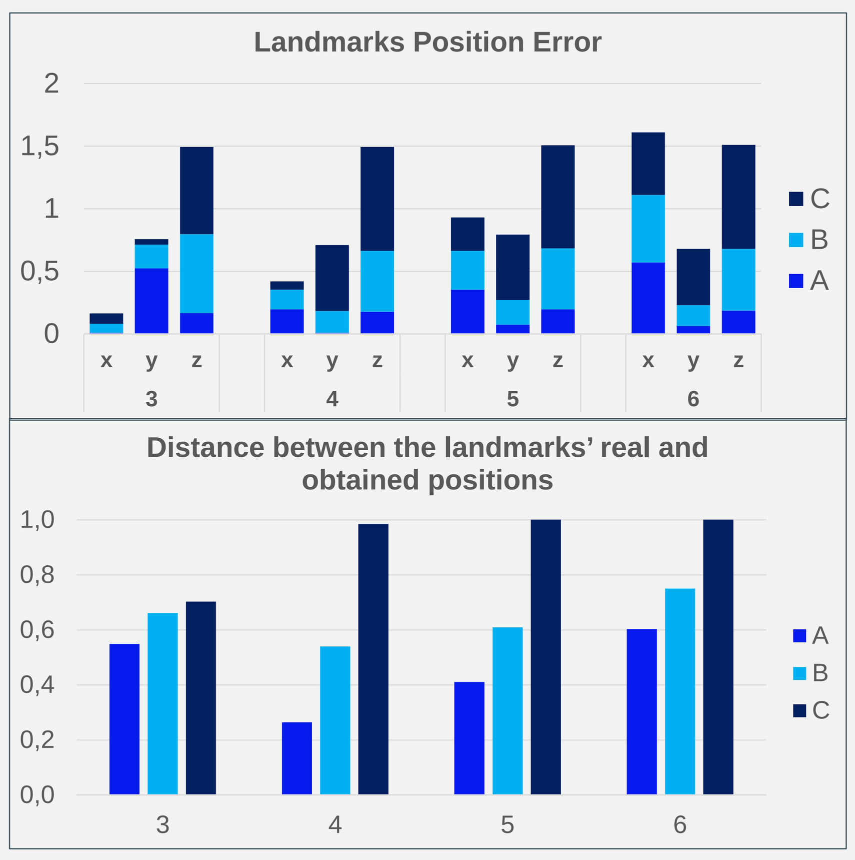
### 2D with Trilateration



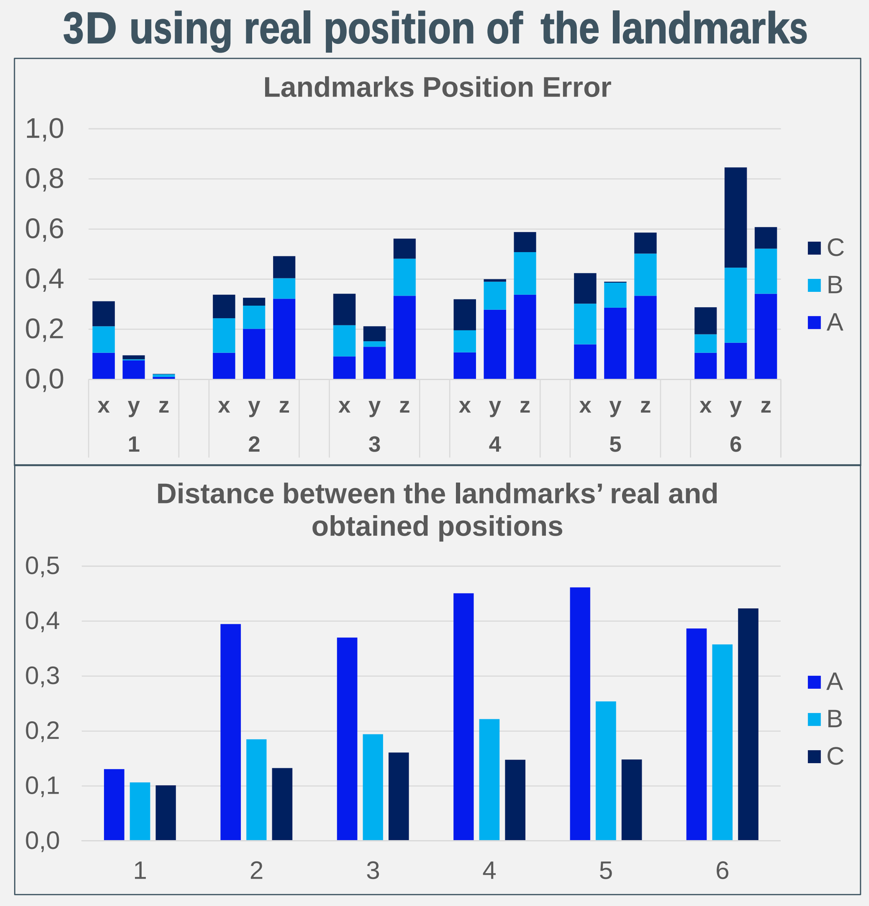

### 2D without initialization




### 3D with Trilateration




### 3D using real position of the landmarks




The criteria to choose the best algorithm was based on the convergence for both 2D and 3D examples. The 3D, and most general case, is only presented when a good initial estimate for each landmark's position is given or calculated by an auxiliary function. When the initial position is very far from the real one, the algorithm was not able to converge. Results also show that, even with a precise initial position, the algorithm induces some errors due to the difficulty to measure a coordinate in z from a fix plane which then induces errors in the other coordinates. For a simulation in 2D, it is possible to obtain accurate results even without a precise initialization.

## Conclusion

The main conclusion that can be drawn regards the observability of this type of systems. Using only the measured distance to the anchors proved to be a challenge to the development of the algorithm. It was possible to conclude that to estimate a position in space, which consists in three independent states, the bearing of the said position when observed by the robot is also important. Although the redesign of the observation model provided better results, it still does not work very nicely, mostly due to the difficulty in estimating the z component of the landmarks position. It was also concluded that communication systems such as the Ultra-Wide Band may give very noisy measurements mostly due to multipath effects.

https://github.com/pcarmo/AutSys_FastSLAM_UWB