Implementation of Robot Behaviour Learning Simulator

Kushagra Singh Bisen

Ecole des Mines de Saint Etienne

k.bisen@yahoo.com

May 18, 2021

Overview

1. SLAM (Simultaneous Localisation and Mapping.)
   - Introduction
   - Visualising SLAM
   - Defining a SLAM Problem
   - Probabalistic Approach for Motion
   - Graphical Model of SLAM
   - More on SLAM
   - MultiRobot SLAM
   - Models in SLAM
2. Simulation Videos
3. Conclusion
   - Comments and Future Work.

What is SLAM?

SLAM [2] stands for Simultaneous Localisation and Mapping, which as the name states, is a way to initiate path planning and navigation in mobile autonomous robots.
It computes a trajectory and models the whole environment (called a 'world' in Gazebo).

What is SLAM?

Localisation

Localisation is employed to estimate the position of a robot, given the landmarks and obstacles. In other words, it 'shows' where the robot is in perspective of the other landmarks along with angular orientation which is important in our work.

Mapping

Mapping is the process through which the robot will know about the world it is in. Mapping is done with a LIDAR (Light Detection and Ranging) Sensor [1] from the robot.

SLAM

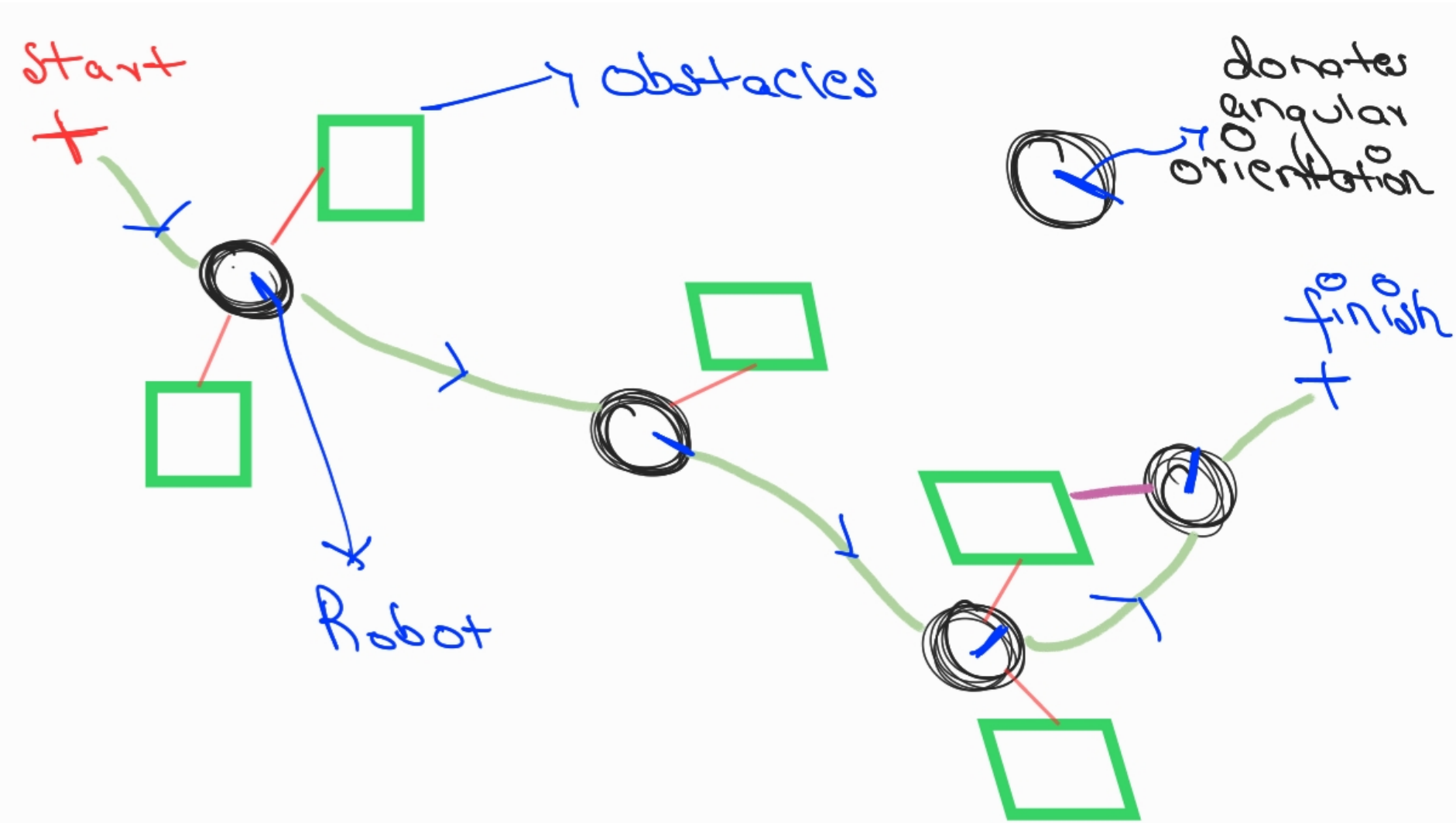
SLAM consists of both Localisation and Mapping and estimates robot's pose and the landmark simultaneously, solving the problem of where to go?

Chicken/Egg Problem

SLAM is a Chicken Egg Problem in the context of the classic argument of who came first.

- A Map is needed for Localization.
- Pose Estimate (Robot's Location) is needed for Mapping.

SLAM Diagram



SLAM Problem

Given

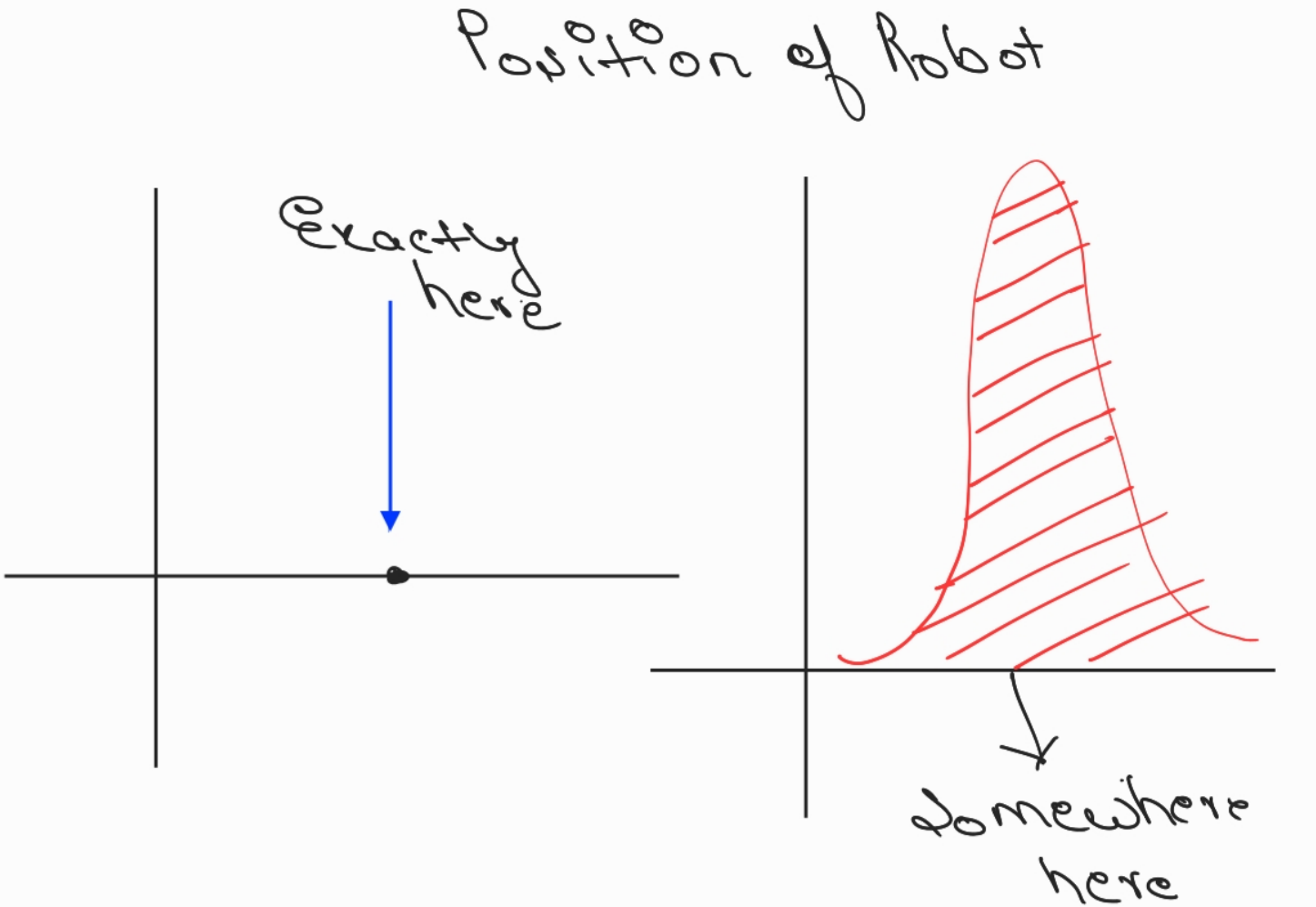
- Robot's Controls, $u_1 : T = u_1, u_2, ..., u_r$
- Observations, $z_1 : T = z_1, z_2, ..., z_r$

Wanted

- Map of the whole environment, m
- Path of the Robot, $x_0 : T = x_0, x_1, x_2, ...x_r$

Probabalistic Approach

A probabalistic approach is applied for SLAM problems as defining exactly where the robot is for Localisation sometimes increases it's uncertainity.

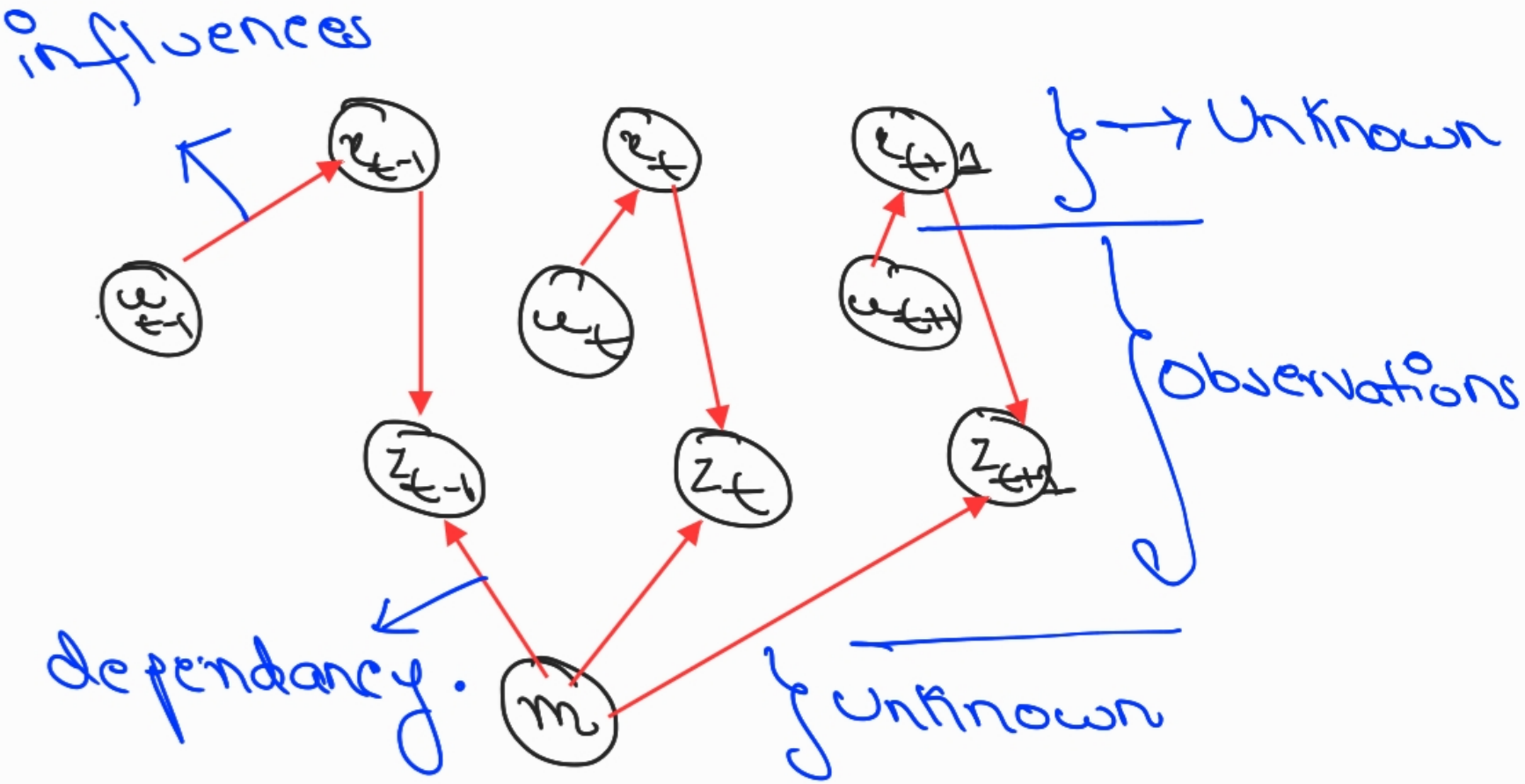


Probabalistic Approach

Path estimation and mapping in probabalistic world will be, $p(x_{0:T}, m | z_{1:T}, u_{1:T})$ where,

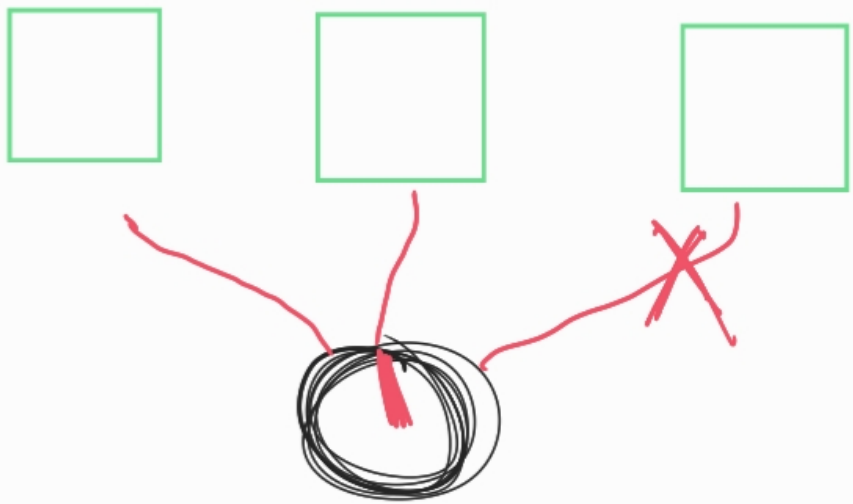
- p $\longrightarrow$ distribution
- x $\longrightarrow$ path
- m $\longrightarrow$ map
- $|$ $\longrightarrow$ given that
- z $\longrightarrow$ Observations
- u $\longrightarrow$ Controls

Graphical Model



Why is SLAM a difficult problem?

- Path of the robot as well as the map are both unknown, and they are correlated.
- Mapping between Observations and the map is unknown. Thus, picking a different data association can cause divergence. This divergence is somewhat similar to the Weka's prediction of the log file, in which at some
instances, it was predicted a different turn from the turn it already took.



Multiple Robots

In a MultiRobot SLAM Simulation, there is an data association among the platforms. It is usually very tricky, but can be done using graph based appraches.

Motion Model

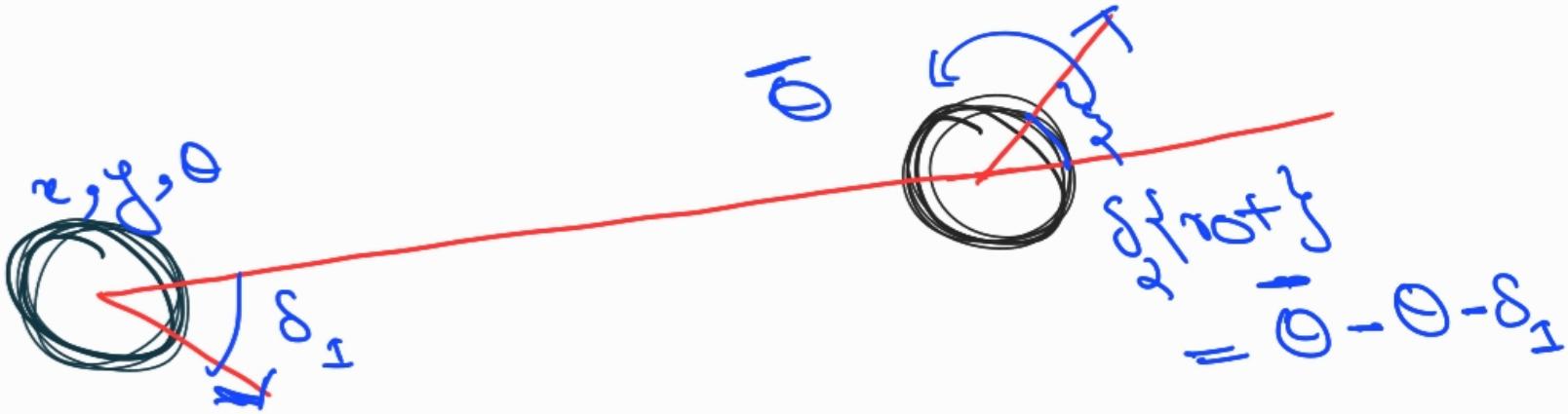
Motion Model is being applied in the SLAM we are using[4]. The probabalistic equation for it will be, $p(x_t|x_{t-1}, u_t)$ where,

- $x_t \longrightarrow$ new pose of the robot.
- $x_{t-1} \longrightarrow$ old pose of the robot.
- $u_t \longrightarrow$ control

In the gaussian model, the robot goes from Point A to Point B, without any pose change. SLAM we are using is a Non Gaussian Model thus helping us to know the pose, and direction of the robot.

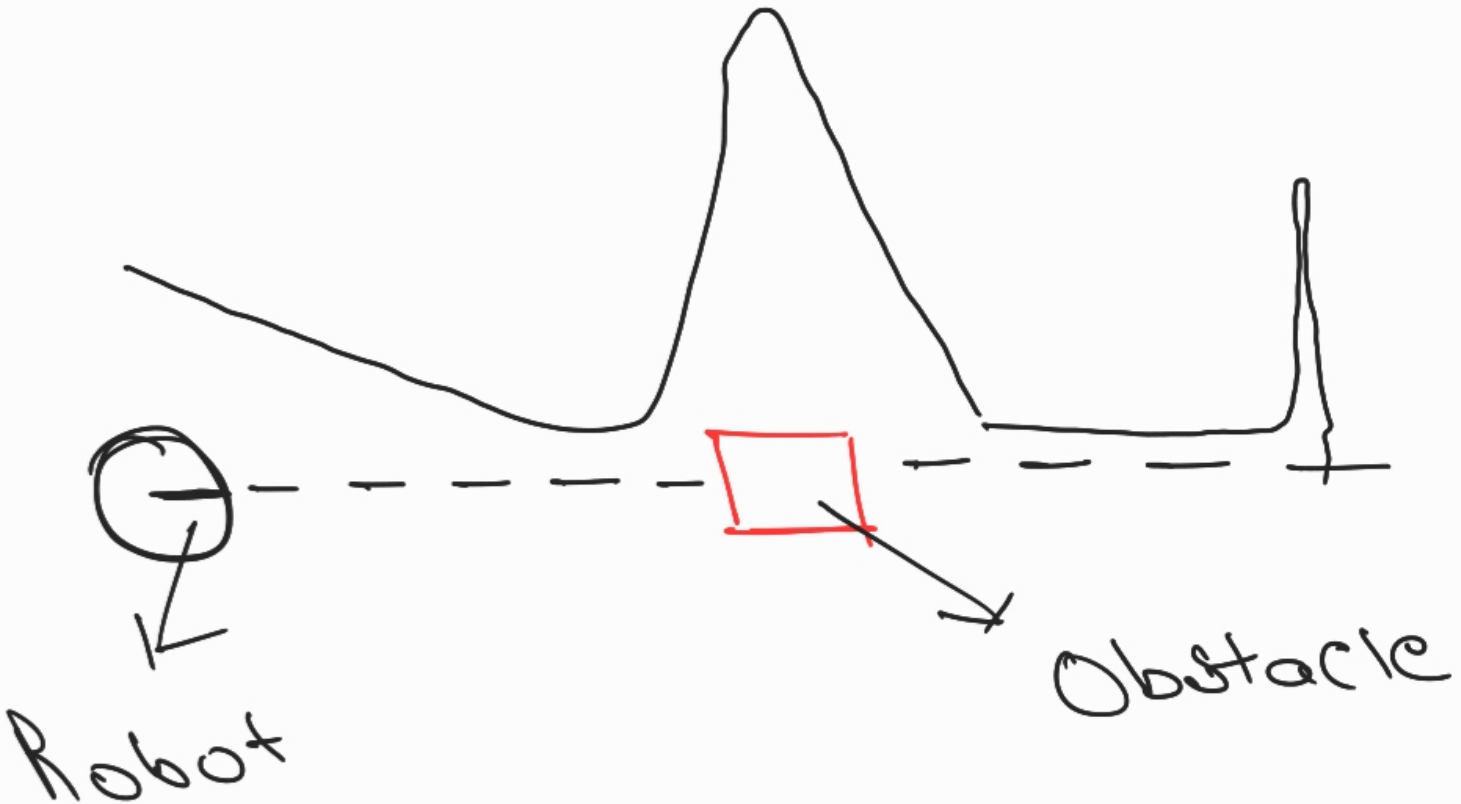
Non Gaussian Model

The transitional rotation is calculated with the value of δ_2.[3]



Observational Model

This model relates measurement with it's pose. $p(z_t|x_t)$ The gaussian model goes straight to the obstacle, whereas the non gaussian model is shown below,



Simulation of SLAM

I will now demonstrate the simulation videos for Simultaneous Localisation and Mapping (SLAM).

- Autonomous navigation with SLAM [2]
- A simple .csv log file demonstration.

Future Work

- As I was trying the SLAM, I implemented it in a world which was already there. I will implement the SLAM on our world now.
- For the log file, the present log file formats already provided by ROS aren't usable for us. I searched and concluded that, the best bet is to subscribe to the /odometry topic in ROS and write file in python by making a function.
- I implemented and got a sort of .csv output. I wish to work on that in future.

References


Wikipedia contributors. "Lidar." Wikipedia, The Free Encyclopedia. Wikipedia, The Free Encyclopedia, 15 May. 2021. Web. 17 May. 2021.


Wikipedia contributors. "Simultaneous localization and mapping." Wikipedia, The Free Encyclopedia. Wikipedia, The Free Encyclopedia, 6 May. 2021. Web. 17 May. 2021.


Introduction to Mobile Robotics - Book.


Probabalistic Robots - Book.

Thank you for listening!