RoboND Project 2: Where Am I?

Jaeyoung Lee

Abstract—Localization is the challenge of determining the robot's pose in a mapped environment. This is done by implementing a probabilistic algorithm to filter noisy sensor measurements and track the robot's position and orientation. The amount of information present and the nature of the environment that a robot is operating in determine the difficulty of the localization task. In this document the famous two localization algorithm Extended Kalman Filter(EKF) and Monte Carlo Localization(MCL) is explained in depth using formulas. And then one of the most well-used implementation of MCL, called ACML(Adaptive MCL) demonstration based on ROS platform is represented. Additionally parameter-tuning for the simulation and configurations are fully described.

Index Terms—Robot, IEEEtran, Udacity, LaTeX, Localization.

1 Introduction

Localization is the challenge of determining the robot's pose in a mapped environment. This is done by implementing a probabilistic algorithm to filter noisy sensor measurements and track the robot's position and orientation. The amount of information present and the nature of the environment that a robot is operating in determine the difficulty of the localization task. The easiest localization problem is Position Tracking, also known as Local Localization. In this problem the robot knows its initial pose and the localization challenge is estimating the robot's pose as it moves out the environment. since there is noise in robot motion, it is not trivial. The uncertainty is limited to regions surrounding the robot. More complicated one is called Global Localization, where the robot's initial pose is unknown and the robot must determine its pose relative to the ground truth map. The amount of uncertainty in Global Localization is much greater than in Local Localization, which makes it more difficult. The most challenging localization problem is the Kidnapped Robot problem. This problem is just like Global Localization except that robot may be kidnapped at any time and moved to the new localization on the map.
There are four popular localization algorithms: Extended Kalman Filter(EKF), Markov Localization, Grid Localization, Monte Carlo Localization(MCL). These four algorithms are described briefly in the rest part of this section. In Section 2, EKF and MCL will be explained in depth using formulas. And then one of the most well-used implementation of MCL, called ACML(Adaptive MCL) demonstration based on ROS platform will be represented in Section 3 to 4 [1]. Additionally simple navigation stack for Localization will also be described. Finally discussions on issues in depth and future work will be mentioned in Section 5 and 6.

1.1 Localization Algorithms

1.1.1 *Extended Kalman Filter*

Most common Gaussian filter, helps in estimating the state of non-linear models

1.1.2 *Markov Localization*

A Bayes filter localization algorithm. It maintains a probability distribution over the set of all possible position and orientation the robot might be located at.

1.1.3 *Grid Localization*

A histogram filter which is capable of estimating the robot's pose using grids.

1.1.4 *Monte Carlo Localization*

A particle filter which estimate the robot's pose using particles.

2 Background

At this stage, you should begin diving into the technical details of your approach by explaining to the reader what are the characteristics of the filters, what localization method was chosen, and the reason that it was selected (i.e. particle filters). This should be factual and authoritative, meaning you should not use language such as "I think this will work" or "Maybe Monte Carlo Localization with these parameters is better...". Instead, focus on items similar to, "Adaptive Monte Carlo Localization was chosen because..." Provide a sufficient background into the scope of the problem technologically while also identifying some of the current challenges in robot localization and why the problem domain is an important piece of robotics. [2]

2.1 Kalman Filters

Briefly describe Kalman filters. Explain how they work and why they are used for localization. Additionally, discuss the drawbacks of linear Kalman filters and how Extended Kalman Filters (EKFs) help resolve some of these issues. Overall Kalman filter process as follows:

- Measurement Update
- State Prediction

Assumptions of Kalman filter is as follows:

- Motion and Measurement models are linear

- State space can be represented by a unimodal Gaussian distribution

Because of these limitation, we need to extend Kalman filter to Extended Kalman Filter(EKF) for nonlinear transformation. It is simple: applying approximation using 1st order Taylor series. $f(x) = f(\mu) + f'(\mu)(x - \mu)$. In multiple dimensions case transformation approximation is as follows:

$$T(x) = f(a) + (x - a)^T Df(a)$$

$$Df(a) = \begin{bmatrix} \frac{\delta f_1}{\delta x_1} & \frac{\delta f_1}{\delta x_2} & \cdots & \frac{\delta f_1}{\delta x_n} \\ \frac{\delta f_2}{\delta x_1} & \frac{\delta f_2}{\delta x_2} & \cdots & \frac{\delta f_2}{\delta x_n} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\delta f_m}{\delta x_1} & \frac{\delta f_m}{\delta x_2} & \cdots & \frac{\delta f_m}{\delta x_n} \end{bmatrix}$$

Even though EKF resolves nonlinearity, Unimodality limitation still remains.

2.2 Particle Filters

Overall MCL process is as follows:

- Previous Belief
- Motion Update
- Measurement Update
- Resampling
- New Belief

The following pseudo code show MCL in detail [3].

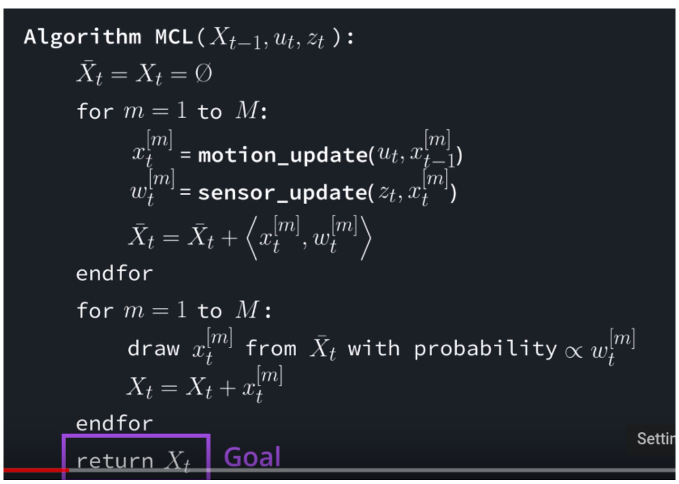

Fig. 1. MCL Pseudo Code

2.3 Comparison / Contrast

Table 1 explains the benefits and disadvantages of using a Kalman Filter / Particle Filter. The following demo uses MCL to represent Global Localization. Only MCL can be used for Global Localization based on pre-set global map data.

3 SIMULATIONS

In this section 2 robot design (the benchmark model and the customized model), used ros packages and their parameters. The customized model is focused on in more depth and detail.

TABLE 1
EKF vs MCL

	EKF	MCL
Measurements	Landmarks	Raw Measurement
Measurement Noise	Gaussian	Any
Posterior	Gaussian	Particles
Efficiency(Memory)	OO	O
Efficiency(Time)	OO	O
Ease of Implementation	O	OO
Resolution	OO	O
Robustness	X	OO
Memory & Resolution Control	No	Yes
Global Localization	No	Yes
State Space	Unimodal Continuous	Multimodal Discrete

3.1 Achievements

All parameters are tuned after a lot of trials on the benchmark model. After parameters are set up, they are applied to the customized model. move_base parameters are also very important in the experiments. They are carefully chosen after more than 10 trials. All other parameters not described here are set as default.

3.2 Benchmark Model

3.2.1 Model design

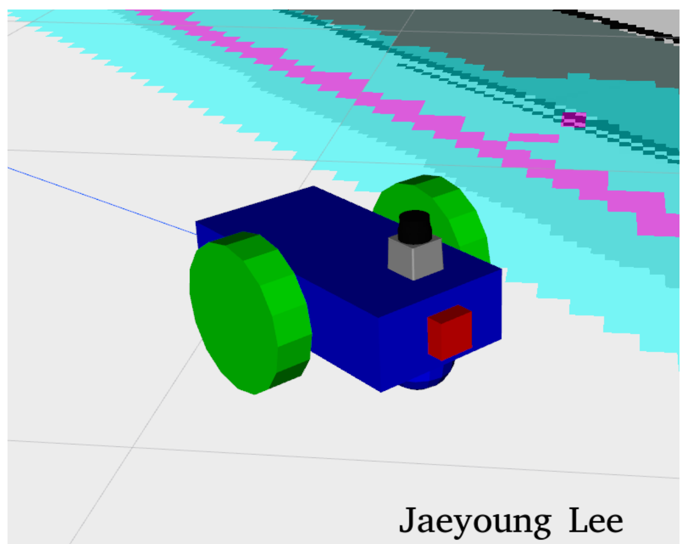

Fig. 2. Robot

3.2.2 Packages Used

- gazebo_ros
- move_base
- map_server
- amcl

3.2.3 AMCL Parameters

- min_particles: 50
- max_particles: 200
- laser_max_beams: 100

- laser_max_range: 20
- update_min_d: 0.05
- update_min_a: 0.01

3.2.4 Costmap Common Parameters

- obstacle_range : 20.0
- raytrace_range : 20.0
- transform_tolerance : 1.0
- inflation_radius : 0.3

3.2.5 Global Costmap Parameters

- global_frame: map
- robot_base_frame: robot_footprint
- update_frequency: 1.0
- publish_frequency: 1.0
- width: 40.0
- height: 40.0
- resolution: 0.05
- static_map: true
- rolling_window: false

3.2.6 Local Costmap Parameters

- global_frame: odom
- robot_base_frame: robot_footprint
- update_frequency: 1.0
- publish_frequency: 1.0
- width: 20.0
- height: 20.0
- resolution: 0.05
- static_map: false
- rolling_window: true

3.2.7 Local Planner Parameters

- controller_frequency: 5.0
- holonomic_robot: false
- meter_scoring: true
- pdist_scale: 1.0
- gdist_scale: 1.0
- sim_time: 3.0
- vtheta_samples: 40

3.3 Customized Personal Model

3.3.1 Model design

After the experiments for the benchmark model, It was founded that localization at the start position and the end position are a bit difficult, even after increasing the number of particles to 1000 or more. It means that more accurate sensor data are needed. Therefore It was decided to increase the maximum angle range of the laser scanner. Increasing 180 degree to 360 degree without changing resolution makes the sensor data easier to be distinguish from each others. More features means more accurate localization. For 360 degree scan range the height of the laser scanner is also increased. Two wheels interfere the laser scan under the benchmark's design. Fig 3 shows taller design than the previous one. The customized one is painted in red for visual difference.

3.3.2 Packages Used

All used packages for the customized model are same as the benchmark's.

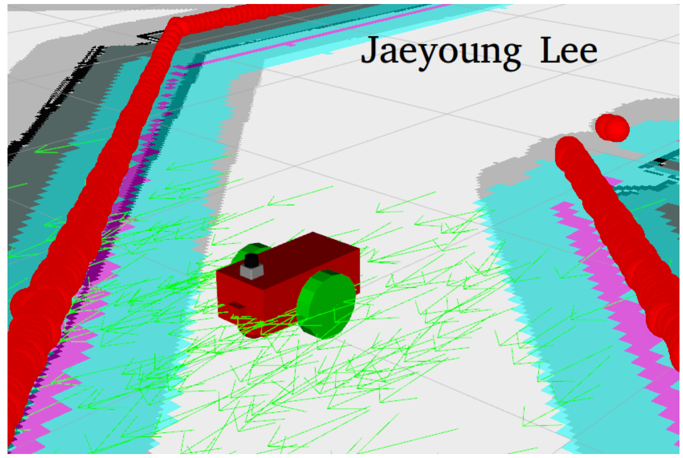

Fig. 3. My Robot

3.3.3 Parameters

All used parameters for the customized model are same as the benchmark's.

4 Results

The performance of the customized model with 360 degree laser scanner is higher than the model with 180 degree one. This results were expected.
The benchmark model rotated several times at the starting point to localize its pose even after particles were converged to the same position. It means that the robot had a trouble in detecting its orientation. This situation also happened at the end point.
On the other hand, the customized one succeeded to detect its orientation just after rotating once or sometimes even without rotation. It did not rotate at the end point. This proves that more accurate sensor data makes localization faster.

4.1 Localization Results

4.1.1 Benchmark

In the initialization stage particles are scattered with gaussian distribution (Fig 4). Just after a few step particles gathered into the ground-truth robot pose (Fig 5). Whole video clip is uploaded here: https://youtu.be/VznrFkwLLHY Fig 6 shows that the robot succeeded to arrive at the goal.

4.1.2 Student Model

In the initialization stage particles are scattered with gaussian distribution (Fig 7). Just after a few step particles gathered into the ground-truth robot pose (Fig 8). Whole video clip is uploaded here: https://youtu.be/meEVGA-2GS8 Fig 9 shows that the customized robot succeeded to arrive at the goal.

4.2 Technical Comparison

Fig 3 shows its 360 degree range of the laser scanner. 180 degree scan range means that the robot does not know about the back side. It seems like that a human driver do not look at the back mirror when he drives a car. For the more perfect localization the robot with limited scan range should rotate to get 360 degree.

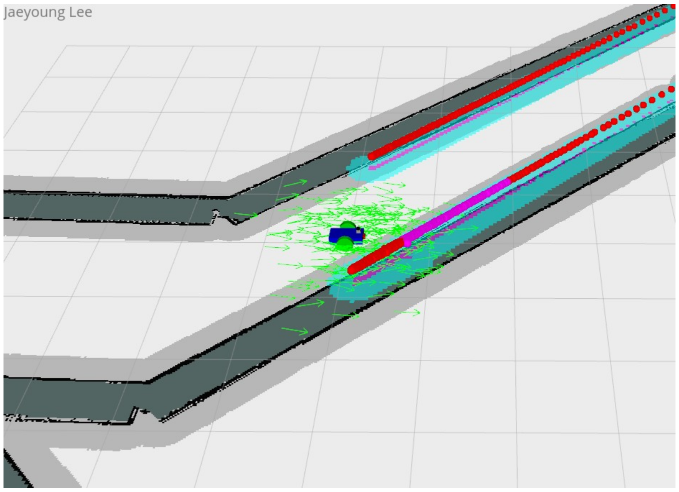

Fig. 4. Initialization

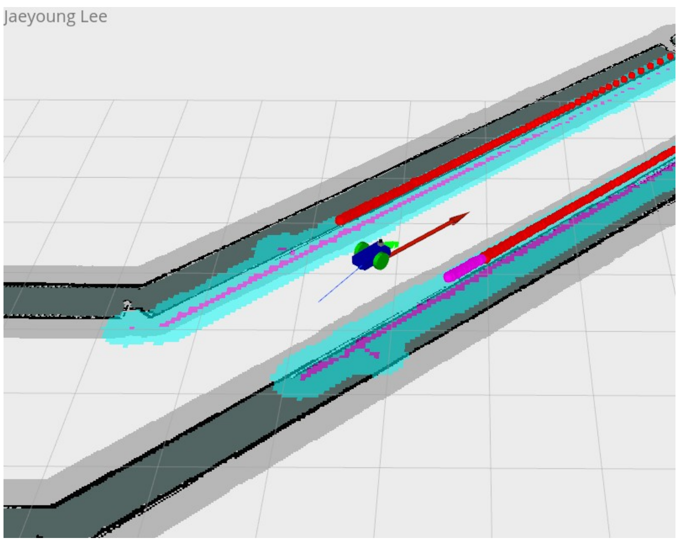

Fig. 5. Status after few step

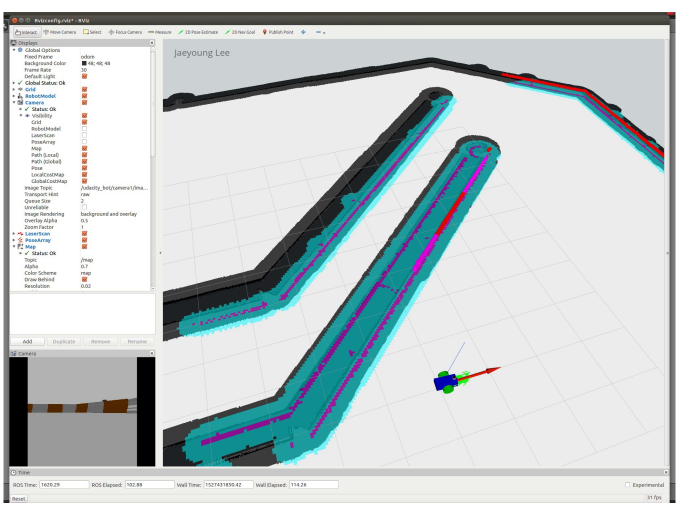

Fig. 6. Final Status

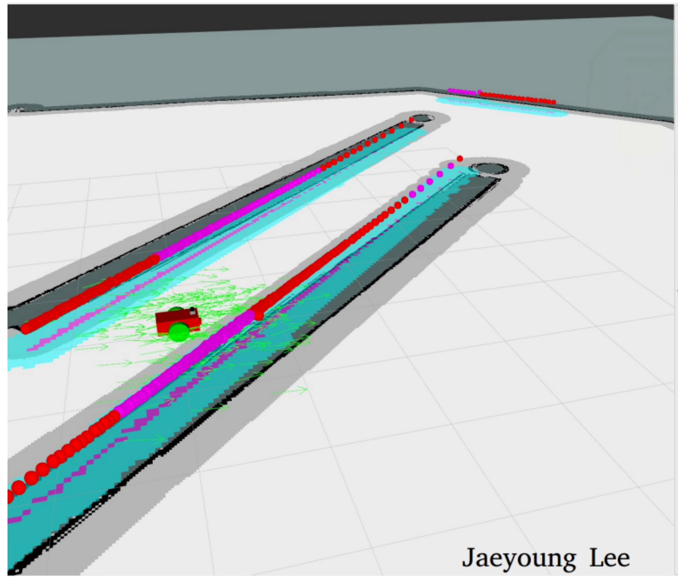

Fig. 7. Initialization

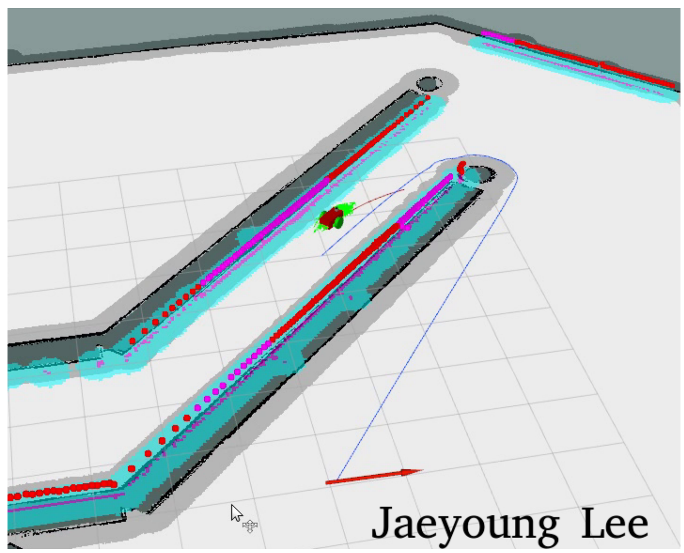

Fig. 8. Status after few step

5 Discussion

5.1 The student presents an unbiased view of their results and justifies their stance with facts.

Under the simulation constraint localization by AMCL are quite good even small size of particles. Since the sensor setup in the simulation are very accurate, particles converges the right robot pose through resampling process in just few step. Noise level and maximum sensor range can affect to not only the quality of MCL localization but also all kinds of localization methods. In the simulation these two conditions are too ideal. This issue are considered in Section 6 again in preparation for real hardware environment.

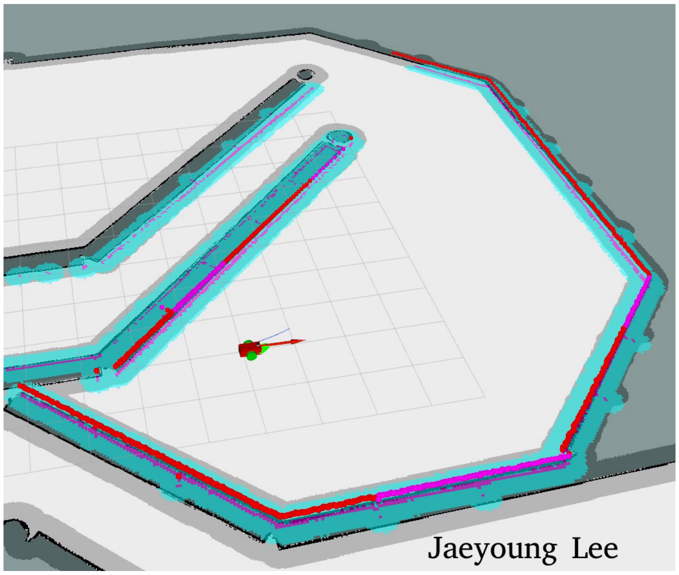

Fig. 9. Final Status

5.2 The student discusses about whether AMCL would work well for the kidnapped robot problem and what kind of scenarios would need to be accounted for it.

The ROS implementation of AMCL is not working well for the kidnapped robot problem. But if kidnapped time is detected by some algorithm, AMCL can be used for searching the new position by generating and distributing particles again.
One another option is that in particle management process including resampling small portion of particles are yield for randomized pose. These random posed or uniform pose particles can be a seed for the new position after kidnapping. For these approach more heuristic resampling algorithms or threshold tuning might be needed.

5.3 The student provides examples with very brief discussions on where they would use MCL/AMCL in an industry domain.

Unamaned Ground Vehicles (UGVs) are used in the many automated factories including AMAZON warehouse. They replaced with human workers to convey packages. Since the factory or warehouse indoor map can be extracted easily from the CAD data or 2D/3D scanning, they can move easily with map and localize its pose.
Another examples are UGVs in military. They can transfer military packages in mountain or other dangerous regions instead of soldiers. Or they can save wounded soldiers and transfer them faster than humans. Attacking by drones are one option but it is one of the most controversial ethical issues in the field.

6 Conclusion / Future work

6.1 The student can accurately and effectively explain the trade-offs in accuracy and processing time. The student identifies other areas of the robot for improvement including the addition of more sensors, different base size, etc.

Accuracy and processing time are always representative one of dilemmas over all engineering area including dynamics, mechatronics and electronics. If the number of particles is increased, particles can represent more wide area. But it takes time more to calculate weights and resampling in linear. Processing time should be under the time limit for appropriate time slice. All sub process including localization should be done determined schedule for the next subprocess. One example is control process after localization.

References

[1] D. Fox, "Kld-sampling: Adaptive particle filters and mobile robot localization," in *In Advances in Neural Information Processing Systems (NIPS*, 2001.
[2] L. Lamport, *LATEX: a document preparation system: user's guide and reference manual*. Addison-wesley, 1994.
[3] S. Thrun, "Particle filters in robotics," in *in Proceedings of the 17th Annual Conference on Uncertainty in AI (UAI*, 2002.