Where Am I

WenTao Wu

Abstract—The purpose of this report is to develop a mobile robot model for Gazebo, and integrate the AMCL and Navigation ROS packages for localizing the robot in the provided map.Using the Parameter Tuning sectionas the basis, add and tune parameters for ROS packages to improve localization results in the map provided.

Index Terms—Robot, IEEEtran, Gazebo, Localization,AMCL,Parameter Tuning.

1 Introduction

THE robot has three positioning problems: local localizing, global localizing, and Kidnapped Robot problem. This project mainly studies the localization problem. Localization algorithm is an indispensable challenge in robot. At present, there are four popular localization algorithms: Extended Kalman Filter algorithm, Markov Localization algorithm, Grid Location algorithm and Monte Carlo Location algorithm. Monte Carlo algorithm is also known as particle swarm algorithm. Each particle represents a possible direction, and it locates the object position through continuous measurement, computation and update. Monte Carlo location algorithm can only solve local localization and global localization problems.

2 Background

Three positioning problems: local localizing, global localizing, and Kidnapped Robot problem.In local localizing problem,robot know its initial position,algorithm need estimating robot's posture.In global localizing problem,The initial posture of the robot is unknown, and the robot must determine its posture relative to the ground map.In Kidnapped Robot problem,algorithm need estimating robot's posture and robot may move to new position on the map at any time.

2.1 Kalman Filters

The Kalamn filters is an estimation algorith that is very prominent in control.It's used to estimate the value of a variable in real time as the data is being collected.The variable can present the position or velocity of a robot.The reason that Kalman filter is so noteworthy is because it can take data with a lot of uncertainty or noise in the measuerments.

The principle of the Kalamn filtersFirst, the residual y between the measured value and the calculated value x is calculated using the value of the measured value Z and the predictive value x of the current state.Then the covariance S of the measurement is calculated. after that, The Calman gain K is calculated using covariance P and measurement covariance S and measurement conversion matrix H.Using the Calman gain K to get a new state X and the new covariance matrix P.The finally, prediction of the next state.

For Kalman filter,it assumption that:

- Motion and measurement models are linear
- State space can be represented by a unimodal Gaussian distribution

It turn out that these assumptions are very limiting,amd would only suffice for very primitive robots.Most mobile robots will execute non-linear emotions. Extended Kalman Filter using multi-dimensional Taylor series to linearize functions with multiple dimensions.It can deal the problem which Kalman Filter can not deal.

2.2 Particle Filters

The Monte Carlo localization algorithm,or MCL is the most popular location algorithms in robots.User choose MCL and deploy it to robot to keep track of its pose.Robot be navigating inside its known map and collecting sensory information using range finder sensors.MCL use these measurements to keep track of robot pose.MCL uses particles to localize robot.In robotics,a particles as a virtual element that resembles the robot.Each particle has a position and orientation and represent a guess of where robot might be located.These particle are re-sampled each time robot moves and sense its environment.

2.3 Comparison / Contrast

	MCL	EKF
Measurements	Raw Measurements	Landmarks
Measurement Noise	Any	Gaussian
Posterior	Particles	Gaussian
Efficiency(memory)	✓	✓✓
Efficiency(time)	✓	✓✓
Ease of Implementation	✓✓	✓
Resolution	✓	✓✓
Robustness	✓✓	x
Memory & Resolution Control	Yes	No
Global Localization	Yes	No
State Space	Multimodel Discrete	Unimodal Continuous

Fig. 1. Comparison.

With the extended Kalman Filter localization algorithm, we can estimate the pose of almost any robot with accurate onboard sensors.But MCL presents many advantages over EKF.first,MCL is easy to program compared to

EKF.Sencond,MCL represents non-Gaussian distrubutions and can approximate any other practical important distribution.This allows MCL to model a much greater variety of environment.Third,in MCL,we can control the computational memory and resolution of our solutiuon by changing the number of pariticles distributed uniformly and randomly throughout the map.

3 Simulations

3.1 Achievements

Through the establishment of ROS packet, the robot URDF file is written to build a robot model. Add rviz files and maps. Add the AMCL package and eventually modify the AMCL Parameters, and move base Parameters enables the robot to reach the target location.

3.2 Benchmark Model

3.2.1 Model design

none	length	width	height	radius
chassis	0.4	0.2	0.1	***
wheel	***	***	***	0.1
camers	0.05	0.05	0.05	***
hokuyo	0.1	0.1	0.1	***

3.2.2 Packages Used

1) RobotModel
2) Camera
3) LaserScan
4) PoseArray
5) Map
6) Pose using topic-current goal
7) Path using topic-global plan
8) Path using topic-local plan

3.2.3 Parameters

1) acml.lanuch
2) base local planner params.yaml
3) local costmap params.yaml
4) costmap common params.yaml
5) global costmap params.yaml

3.3 Personal Model

3.3.1 Model design

Model design:

none	length	width	height	radius
chassis	0.4	0.2	0.1	***
another block	0.3	0.2	0.1	***
wheel	***	***	***	0.1
camers	0.05	0.05	0.05	***
hokuyo	0.1	0.1	0.1	***

3.3.2 Packages Used

1) RobotModel
2) Camera
3) LaserScan
4) PoseArray
5) Map
6) Pose using topic-current goal
7) Path using topic-global plan
8) Path using topic-local plan

3.3.3 Parameters

In personal model,using the same parameters as Robot Models.It can see in git

4 Results

4.1 Localization Results

Navigating through the MCL algorithm, the robot finally reaches its destination, but it is not very fluent in the process, and it sometimes walks along a curve. It usually takes 10 minutes or so to complete the whole process

4.2 Comparison / Contrast

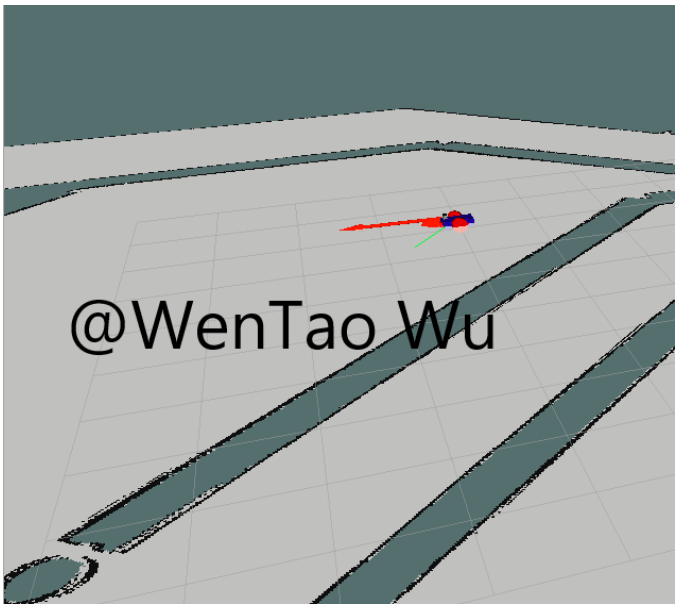

Fig. 2. Comparison.

4.3 Technical Comparison

In the benchmark model, the robot can get to the destination well by adjusting parameters. But if we use the same parameters in the robot we designed, if the model is very different from the benchmark model, the original parameters are not suitable. A robot that only modifies parts can still reach its destination, though it can't follow the process of the original robot. The shape of the robot has a great influence on the parameters.

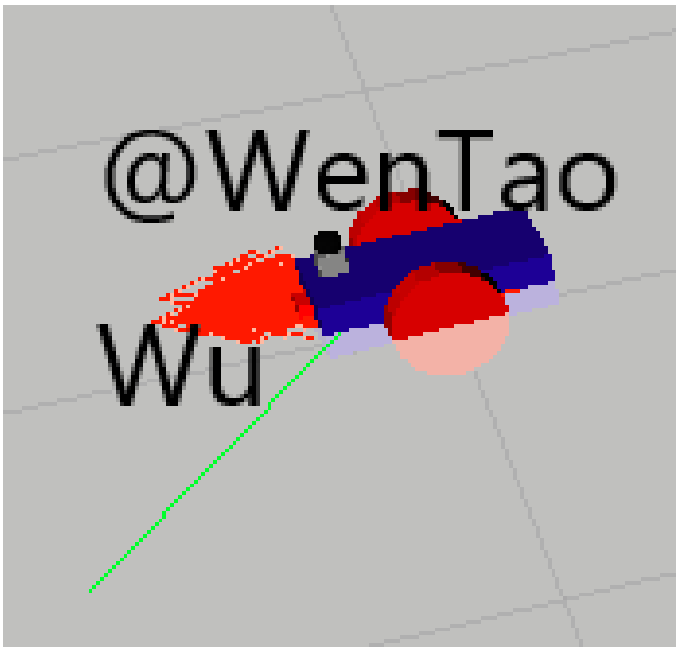

Fig. 3. Comparison.

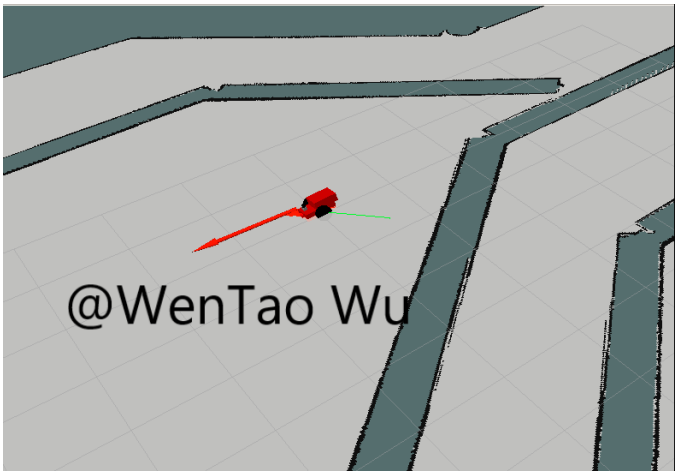

Fig. 4. Comparison.

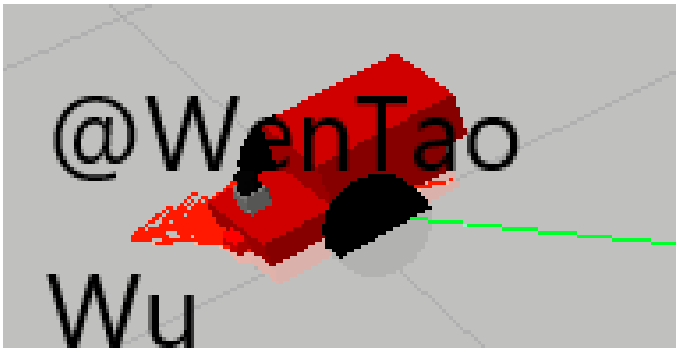

Fig. 5. Comparison.

5 Discussion

It is obvious that in the project, the size and weight of the robot have a great impact on the performance of the robot. It is applicable to the parameters of the previous robot can not adapt well to the next robot. This method does not solve kidnapped robot problem. But if the robot can determine whether it is moved, then restarting the Monte Carlo algorithm can restart its own positioning navigation.

6 Conclusion / Future work

It is shocking that too much time is spent on parameter tuning, and there are not too many hints in this area. Small parameter corrections may make the whole test fail. The relationship between the parameters and the size and weight of the robot will be studied in the future. And the influence of the sensor. The use of rqt reconfigure in the process of tuning the whole parameter can help to quickly adjust the parameters.