Robot Localization Using Particle Filter

|  |  |
| --- | --- |
| *The Second Students Innovative Conference (SIC2013)*  *June 12,2013- Hebron, State of Palestine* | *Palestine Polytechnic University* |

Sabha Abu Sabha, Samah Ratib Badawi and Hasehm Tamimi  
College of Informaion technology and Computer Engineering

Palestine Polytechnic University  
Hebron, Palestine   
[sabha|sbadawi|htamimi]@ppu.edu

**Abstract**

Particle filter have many applications and learning its concepts is critical task to be accomplished. Important application of particle filter in robot localization; where it makes robot localize itself, based mainly on particles updated using a probabilistic model. In this paper we provide an educational application of particle filter, that clarify the idea and make a simulation of robot localization process, show different localization parameters, and the effect of these parameters on the localization process.

Keywords

Particle Filter, resampling, mobile robot loclization, sensor.

# Introduction

Particle filter is one of important and interesting topic nowadays. Particle filter (called also Sequential Monte Carlo method) is an estimation  model based on simulation[1]. Particle filters are usually used to estimate Bayesian models in which the observable variables are connected in a Markov chain, where the state space of the observable variables is continuous rather than discrete, and not sufficiently restricted to make exact inference tractable [2].

We can find particle filter used in many application: In Computer Vision (Object tracking) [3], economics [2], and mobile robot localization [4]. In this paper we use particle filter to demonstrate its concept in mobile robot localization. Given map, we can determine the robot location, and from sensor reading the robot will be able to infer its most likely position on the field and explore the world to determine its structure. [4]

This paper is organized as follows. Section II is the related work, Section III is the theoretical background of particle filter, Section IV describes our experiments and the simulator that we build and Section V we conclude the paper.

# Related Work

There are many works done using particle filter to make optimal solution in a nonlinear state space. Particle filter is an a tractable implementation of Bayes filter [5][6] .

In particle filters, the belief distribution is represented by a set of samples, called *particles*, randomly drawn from the belief itself. The particle filter is in charge of recursively updating the particle set. (Dellaert et al. 1999) [7]. Fox et al. (1999) introduced particle filters in the localization context, defining the so called MCL (*Monte Carlo Localization*) [8] .

Since then, particle filters have been successfully applied to simultaneouslocalization and Mapping (SLAM) (Montemerlo et al. 2002) [9] , multi-robot localization (Fox et al. 2000) [10] , localization given an *a priori* map both using laser (Yaqub & Katupitiya 2007) [11] and sonar sensors (Thrun et al. 2001) [12] ,among many other applications.

# Theory

Robot localization application that we build in this paper depends mainly on the particle filter approach; and the general problem described as follow:

Assume we have a robot move at some known environment but without knowing at what position it is initially located the goal is to make the robot at some instance know its approximate location; where this will be done using N particle according to Algorithm 1[4] :

Some of important steps that will be used to implement Algorithm 1 described in detail below [13]**:**

**Random particles localization:** the first step includes the distribution of particles along the tracked area randomly, according to the random numbers distribution concept.

|  |
| --- |
| ***1.initialization,***  ***set t=0:***  ***robot starts its motion.***  ***for i=1 to N***  ***set random location for particle .***  ***Apply uniform initial weight for particle .***  ***end for***  ***set t=1.***  ***2. robot sensing of current evidence Z.***  ***3. Importance sampling step:***  ***for i=2 to N***  ***P()=P(| Z).***  ***end for***  ***4.normalize the importance weights.***  ***5.resample with replacement N particles.***  ***6.calculate variance of particles location:***  ***if variance < threshold***  ***Then return robot location and stop.***  ***else***  ***move Robot with particles.***  ***end if***    ***set t=t+1.***  ***go to step 2.*** |
| *Algorithm 1: Particle filter algorithm* |

**Uniform initial weight:** all particles will initially have probability that is based on uniform distribution, where all particles have the same weight at the beginning related to the number of whole particles.

**Sensing:** the robot at specific location as shown in the second step, will sense the surrounding area to determine the evidence at that point, which will be used to update the importance weight in next step.

**Importance sampling:** according to the measured value that robot obtained after sensing, we will based on it to update the importance for each particle according to Bayes rule as in Eq. (1) :

**Normalization:** after sampling the whole probability summation must be normalized to be 1 according to the rule of normalization rule Eq.(2) show that:

2

**Resample:** after the assign of importance weight for each particle, we will perform resample with replacement approach, in order to reweight all particles with uniform weights and regenerate the initial number of particles distributed according to the importance, according to the following rule:

**Variance:** each time we finish one iteration we will test to see whether the particles are close to each other enough to say that this is a robot location or not, by measure the variance of the particles location according to the rule as in Eq.(3):

**Movement:** after the decision to repeat the process, and before go back to repeat sensing the robot will change its position and the particles will move accordingly and there will be some uncertainty during movement.

# Expiremnt

We make a simulation of the particle filter for robot localization process, the implementation environment was done using both C# and Java programming languages. We apply the process both on one dimensional map and two dimensions map. A user interface was created to clarify the concepts, mainly for educational purpose. The interface (see Figure 1) includes ability to change the parameters in the algorithm in order to understand their effects on the localization process.

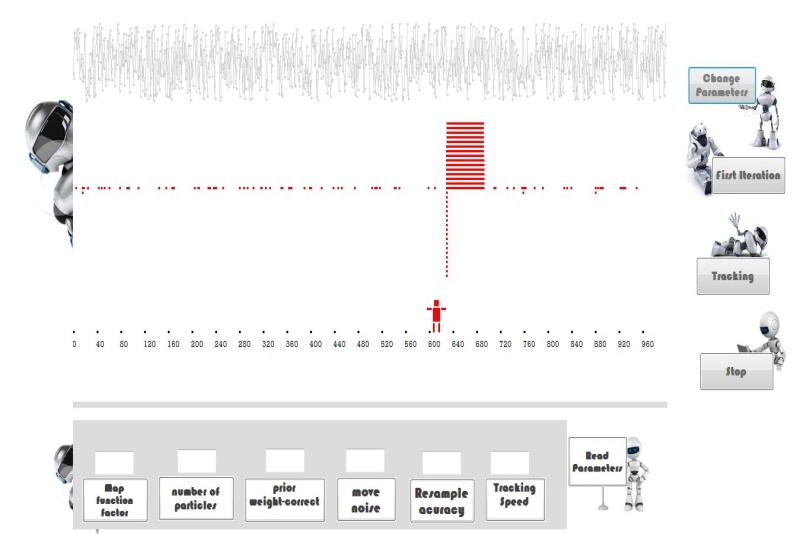


Figure 1: Program User Interface

# Conclusion

Learning Particle filter depends on statistics and uncertainty information about world, so it is considered very important to understand it thoroughly. We proposed an educational application for learning particle filter in robot localization that clearly presents the idea. Also, we showed that the process may be enhanced by control some of the parameters that related to robot and particles nature.

# Acknowledgment

This work done in AI course for 2013, and delivered for The Second Students Innovative Conference (SIC2013),State of Palestine

# References

# [1] A.Doucet, S.J.Godsill, and C.Andrieu, *"On sequential Monte Carlo sampling methods for Bayesian ﬁltering"*, Statist. Comput., 10, 197-208, 2000.

[2] T.Flury & Neil Shephard,. *" On Bayesian inference based only on simulated likelihood: particle filter analysis of dynamic economic model "* ,OFRC Working Papers Series 2008fe32, Oxford Financial Research Centre, 2008.

[3] L. MIhaylova, P. Brasnett, N. Canagarajah and D.Bull, *"On Object Tracking by Particle Filtering*

*Techniques in Video Sequences"*, Department of Electrical and Electronic Engineering, University of Bristol, UK,2005

[4] S.Thrun , *"On Particle Filters in Robotics"*, Computer Science Department ,Carnegie Mellon University,Pittsburgh, PA 15213, 2002

[5] A.Doucet; de.Freitas, J. & N.E Gordon, *"On Sequential Monte Carlo Methods in Practice",* Springer-Verlag, 2001

[6] N.Metropolis, & S.Ulam, (1949). *" On The Monte Carlo method"* , Journal of the American Statistical Association, No. 44, 1949, pp. 335-341.

[7] F.Dellaert , W.Burgard & S.Thrun, *"On Monte Carlo Localization for mobile robots",* Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), 1999

[8] D.Fox, W.Burgard, F. Dellaert, & S.Thrun. *"On Monte carlo localization: Efficient position estimation for mobile robots",* Proceedings of the National Conference on Artificial Intelligence (AAAI), Orlando (USA), 1999

[9] M.Montemerlo , S.Thrun, D. Koller & B.Wegbreit . *"On FastSLAM: A factored solution to the simultaneous localization and mapping problem",* Proceedings of the AAAI National Conference on Artificial Intelligence, 2002.

[10] D.Fox, W.Burgard, H. Kruppa, & S. Thrun*. " On A probabilistic approach to collaborative multi-robot localization"* . Autonomous Robots, Vol. 8, No. 3, 2000.

[11] T.Yaqub & J. Katupitiya , *"On Laser scan matching for measurement update in a particle Filter"* , Proceedings of the IEEE/ASME International Conference on Advanced Intelligent

Mechatronics, pp. 1-6, September 2007.

[12] S.Thrun, D.Fox, W. Burgard & F. Dellaert, *"On Robust monte carlo localization for mobile robots"* . Artificial Intelligence, Vol. 128, No. 1-2, 2001, pp. 99-141.

[13] M. Johannes, N. Polson, "On particle filtering", Columbia University, 2006