

**Military Technical College  
Kobry El-Kobbah,  
Cairo, Egypt.**



**18<sup>th</sup> International Conference  
on Applied Mechanics and  
Mechanical Engineering.**

## **A MODIFIED SAMPLING METHOD FOR LOCALIZATION ACCURACY IMPROVEMENT OF MONTE CARLO LOCALIZATION**

M. A. Awad-Allah<sup>1</sup>, M. A. Abdelaziz<sup>2</sup>, M. A. Shahin<sup>3</sup> and F. A. Tolbah<sup>4</sup>

### **ABSTRACT**

Unmanned vehicles are devices that can move around and perform tasks without an operator onboard. Such features are essential in many applications. Localization is a very important task in any autonomous mobile robot; in order to reliably navigate, the robot must keep accurate track of where it is. In the past few years Monte Carlo Localization (MCL) has been one of the most successful and popular approaches to solve the localization problem. MCL is a Bayesian algorithm based on particle filters. This paper is an attempt to increase the accuracy of localizing a mobile robot by modifying the way of generating samples from the proposal distribution of the MCL algorithm. Results show improvements in localization accuracy as compared to the basic MCL algorithm.

### **KEY WORDS**

Monte Carlo Localization, Mobile robots; Position estimation; Particle filters.

---

<sup>1</sup> Graduate student, Dept. of Mechatronics, Faculty of Engineering, Ain Shams University, Cairo, Egypt.

<sup>2</sup> Assistant professor, Dept. of Automotive, Faculty of Engineering, Ain Shams University, Cairo, Egypt.

<sup>3</sup> Professor, MTI University, Cairo, Egypt.

<sup>4</sup> Professor, Dept. of Mechatronics, Faculty of Engineering, Ain Shams University, Cairo, Egypt.

## NOMONCLATURE

$M$	number of modified samples
$N$	number of original samples
$u_t$	Control data at time $t$
$\omega_t$	Weight of sample
$x_t$	Robot state at time $t$
$z_t$	Measurement data at time $t$

## INTRODUCTION

Autonomous vehicle may navigate depending on robot capability, namely sensors installed on board and processing resources. The localization problem is the problem of estimating the robot's current position relative to a known map of its environment. Mobile robot localization is one of the most basic perceptual problems in robotics. This is because nearly all robotic tasks require knowledge of the location of the robot and the objects that are being manipulated. Generally, there is a belief that probabilistic approaches are among the most promising candidates to providing a comprehensive and real-time solution to the autonomous system problems. The aim of this paper is to improve Monte Carlo Localization model and its computational approach.

## TYPES OF LOCALIZATION ALGORITHMS

Most of the earlier approaches to robot localization apply Kalman filters which have proven to be robust and accurate for keeping track of the robot's position. Kwon et. al. [1] and Yomchinda [2] used kalman and extended Kalman Filter (EKF) and claimed success of their proposed methods. However, there exist a number of extensions of the basic Kalman filter such as extended kalman filter (EKF) and Multihypothesis Tracking Algorithm (MHT). In practice, localization approaches using Kalman filters typically require that the starting position of the robot is known and the assumption of uncertainty is always Gaussian. Generally, this is well-suited for local position tracking problems with limited uncertainty and in environments with distinct features. They are less applicable to global localization or in environments where most objects look alike.

The straightforward application of Bayes filters to the localization problem is called Markov localization. The central idea of Markov localization is to represent the robot's belief by a probability distribution over possible positions, and use Bayes rule and convolution to update the belief whenever the robot senses or moves. Markov localization employs discrete multi-modal representations for representing the robot's belief, hence can solve the global localization problem. Naseer et. al. [3], successfully applied Markov localization approach by using visual measurements.

Grid localization approaches perform numerical integration over an evenly spaced grid of points. This involves discretizing the interesting part of the state space, and uses it as the basis for an approximation of the state space density. Grid-based methods are powerful, but suffer from excessive computational overhead and a priori commitment to the size and resolution of the state space. In addition, the resolution

and thereby also the precision at which they can represent the state have to be fixed beforehand. Computational requirements have an effect on accuracy as well, since all measurements can be processed in real-time, and valuable information about the state is thereby discarded. An occupancy grid based localization framework is presented in Guo et. al. [4], in order to obtain a precise positioning with relatively low-cost sensor configuration.

Das et. al. [5] presented a novel localization method which can be applied in an environment having orthogonal sets of equally spaced lines to form a grid.

### MONTE CARLO LOCALIZATION

Monte Carlo Localization, or MCL has become one of the most popular localization algorithm in robotics; it represents the belief about the robot's state  $bel(x_t)$  by a set of weighted particles. MCL uses sampling techniques to represent the robot's belief. When the robot moves or senses, importance re-sampling is applied to estimate the posterior distribution.

Table 1 shows the basic MCL algorithm, which represents the belief  $bel(x_t)$  by a set of  $N$  particles  $x_t = \{x_t^{[1]}, x_t^{[2]} \dots \dots \dots, x_t^{[N]}\}$ . Line 4 in the algorithm applies the sampling (from the motion model), using particles from present belief as starting points. The measurement model is then applied in line 5 to determine the importance weight of that particle. The initial belief  $bel(x_0)$  is obtained by randomly generating  $N$  such particles from the prior distribution  $P(x_0)$ , and assigning the uniform importance factor  $N^{-1}$  to each particle.

**Table 1.** Monte Carlo Localization algorithm based on particle filters [6].

1:	<b>Algorithm MCL</b> ( $x_{t-1}, u_t, z_t, map$ ):
2:	$\bar{x}_t = x_t = \phi$
3:	for $m = 1$ to $N$ do
4:	$x_t^{[N]} = \text{sample\_motion\_model}(u_t, x_{t-1}^{[N]})$
5:	$\omega_t^{[N]} = \text{measurement\_model}(z_t, x_t^{[N]}, map)$
6:	$\bar{x}_t = \bar{x}_t + \langle x_t^{[N]}, \omega_t^{[N]} \rangle$
7:	endfor
8:	for $m = 1$ to $N$ do
9:	draw $i$ with probability $\propto \omega_t^{[i]}$
10:	add $x_t^{[i]}$ to $x_t$
11:	endfor
12:	return $x_t$

MCL has several key advantages over earlier work in the field:

1. In contrast to existing Kalman filtering based techniques, it is able to represent multi-modal distributions and thus can globally localize a robot.

2. It drastically reduces the amount of memory required compared to grid-based localization and can integrate measurements at a considerably higher frequency.
3. It is more accurate than Markov localization with a fixed cell size, as the state represented in the samples is not discretized.
4. It is much easier to implement.
5. MCL is an online algorithm.

## LIMITATIONS OF MCL

Beside advantages of MCL there is a range of limitations, such as the inability to estimate posteriors for highly accurate sensors, poor degradation to small sample sets, and the ability to recover from unexpected large state changes (robot kidnapping).

In order to overcome these limitations Fox et. al. [7] and Thrun et. al. [8] introduced a randomized Bayesian algorithm and later a Mixture-MCL algorithm and hence researches continued with such approach.

## MCL SURVEY

Dellaert and Fox et. al. [9] introduced a novel approach to mobile robot position estimation. By using Monte Carlo type methods, they combined the advantages of grid-based localization with the efficiency and accuracy of Kalman filter based techniques. Burgard et. al. [10], proved that MCL leads to a variety of advantages: A significant reduction in computation and memory consumption, which leads to a higher frequency at which the robot can incorporate sensor data, which in turn implies much higher accuracy. Schulz [11] used the Monte Carlo sampling techniques to ensure that the robot is capable of quickly determining the state of objects by taking the robot's uncertainty about its position into account. Röfer et. al. [12] increased the stability of the localization by a slow adaptation of the probabilities of the samples. They improved the precision of localization by using a so-called probabilistic search that moves samples locally dependent on their probabilities.

Pfaff et. al. [13] suggested a learning function that outputs the appropriate variance for each particle based on the estimated area in the state space represented by this particle. Grisetti et. al. [14] presented an optimization that requires less memory and computational resources, an improvement over other state-of-the-art mapping techniques using Rao-Blackwellized particle filters applied to solve the SLAM problem using grid maps. Delius et al. [15] indicated that the improved sample-based maps increase the accuracy during robot localization when compared to the maps before improvement. Beinhofer et. al. [16] applied a Monte Carlo simulation using the real system dynamics to check if the selected landmark placement set of their method satisfies the deviation guarantee also for the possibly non-linear models. Zapata et. al. [17] propose an improved Monte Carlo localization algorithm using self-adaptive samples, abbreviated as SAMCL which is more efficient than regular MCL. Frank et. al. [18] improved their system by replacing the collision detection method which used in [19] and extended the localization module, which is based on Monte Carlo localization. Wolf et al. [20] combined an image retrieval system with Monte-

Carlo localization for robustly localizing a robot in indoor environments to achieve better performance.

Qomaruddin et. al. [21] introduced Monte Carlo localization as method to estimate robot position of robot continuously. Once position of the robot is estimated, dynamic programming would successfully determine the fastest route to the target. Almeida et.al. [22], presented a novel implementation of Monte-Carlo Localization as a localization technique which was applied at humanoid robots whose main sensory input is a camera. Such proposals are useful and were validated by simulated experiments. Rui and Ho [23] proposed an algorithm to locate an object by an asynchronous relay network using time of arrival (TOA) measurements and Markov Chain Monte Carlo method, simulations showed that the proposed method has high accuracy for measurement association and yields a good localization accuracy. Peters [24] developed a localization method that can account for transmitter and receiver location errors, besides sound-speed, time and bearing errors. A Monte Carlo test was conducted to compare the accuracy of the proposed method to that of a more conventional method used as a baseline. The degree of improvement was claimed to increase with a larger region area, larger bearing measurement error, and with a smaller time-of-arrival measurement error. The degree of improvement is also slightly greater with a larger number of receivers.

## MODIFIED SAMPLING METHOD FOR MCL

The idea of the MCL algorithm is to represent the belief  $bel(x)$  by a set of  $N$  weighted samples distributed according to  $bel(x)$ . It is intuitive that increasing the number of samples  $N$  yields a better approximation of the belief distribution and minimizes the mismatch between the target and proposal distribution; therefore, improving the localization accuracy.

The basic MCL algorithms has two basic steps, the prediction and correction step.

### 1- Prediction step:

Using an initial particle set  $(x_{t-1}^i)$  of  $N$  samples apply the motion model to each particle in the set to draw a sample from the density  $p(x|x_{t-1}, u_t)$ . This results in  $N$  predicted samples.

i.e. for  $(x_{t-1}^i) \quad i=1 \longrightarrow N$

draw **1** sample  $(x_t^i)$  from  $p(x|x_{t-1}, u_t)$

### 2- Correction step:

In this step the sensor measurement  $(z_k)$  is taken into account to weigh each particle sample  $(x_t^i)$  obtained from the previous step. At this point each particle  $(x_t^i)$  has a weight according to its importance.

Finally this sample set  $(x_t)$  is resampled according to their particle weights.

i.e. for  $(x_t^i) \quad i=1 \longrightarrow N$

draw one sample  $(x_t^i)$  from  $\{x_t^i, \omega_i\}$  (Importance Resampling)

In this work, a modification on the sampling generation procedure of the prediction step is introduced. Instead of drawing one sample from  $p(x|x_{t-1}, u_t)$  in the particle set, it is suggested to draw  $M$  samples instead of one.

Drawing more samples from the distribution  $p(x|x_{t-1}, u_t)$  increases the chances that one of these samples is the “correct” sample. This fact arises from the uncertainty in

robot motion which increases the spread of the distribution  $p(x|x_{t-1}, u_t)$  hence it is intuitive to increase the number of samples drawn to better represent this distribution.

Sampling modification of MCL:

**3- Prediction step:**

For  $(x_{t-1}^i) \quad i=1 \longrightarrow N$   
 Draw  $M$  samples from  $p(x|x_{t-1}, u_t)$   
 Then the size of  $(x_t^m)$  is  $N * M$

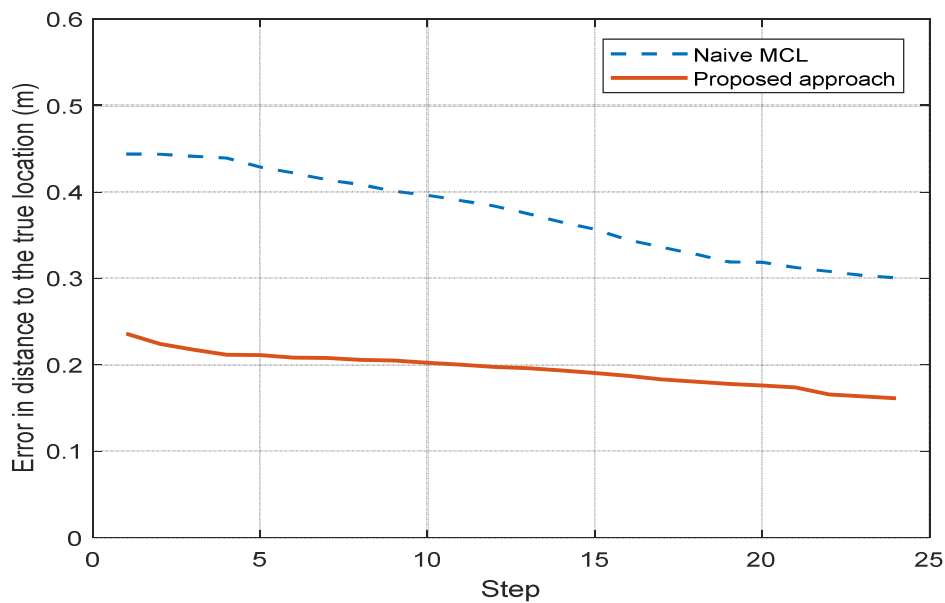
**4- Correction step:**

For  $(x_t^m) \quad m=1 \longrightarrow N * M$   
 Incorporate sensor measurement  $z_k$  to weigh each particle in the sample set.  
 Draw one sample  $(x_t^m)$  from  $\{x_t^m, \omega_m\}$  and down sample to choose the best  $N$  samples, this is to make sure that the size of the sample set will not increase monotonically.

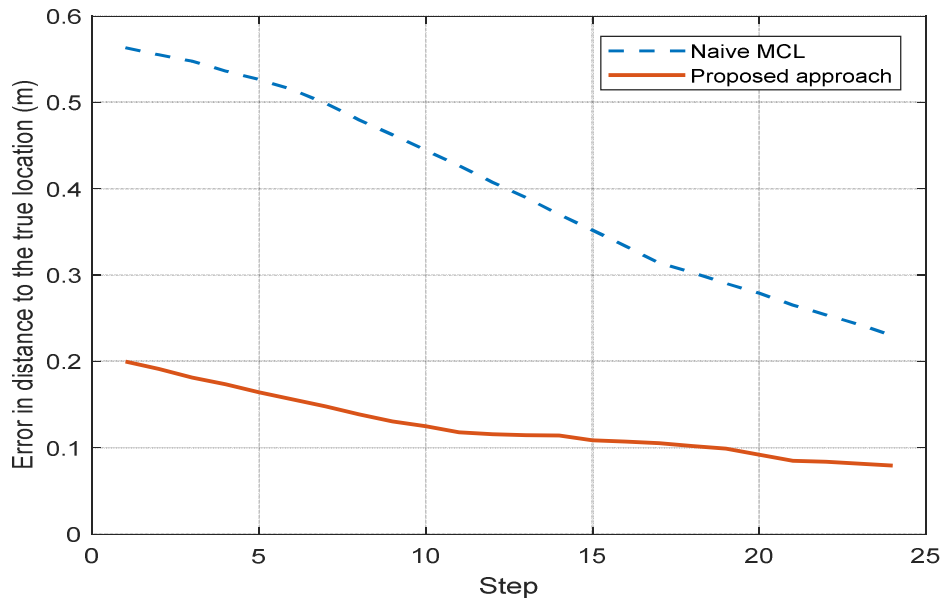
This sampling approach means that for the same number of overall samples it is better to choose  $M > 1$ . In other words, if the required number of samples is 1000 for example, it is better to choose  $N = 100$  and  $M = 10$ , this yields a total number of 1000 samples but improves the overall localization accuracy than if  $M = 1$  and  $N = 1000$ . The results presented next shows this improvement in localization accuracy.

**RESULTS AND ANALYSIS**

In this work, a simulated robot on the Matlab environment was used to verify the proposed technique. The simulated robot is equipped with a laser sensor (Lidar) and wheel encoders. The robot is driven around in a virtual environment of a given map and laser scan data and encoder odometry were collected.



**Fig. 1.** Distance error when using a total of 100 samples.



**Fig. 2.** Distance error when using a total of 1000 samples.

From the previous figures, it may be seen that the proposed modification in the sampling procedure increases the accuracy of localization recorded over 25 robot steps. This improvement is obtained without any other modification to the original basic MCL algorithm, i.e. no modifications were made on the motion or sensor models or even the resampling method. It should be noted that this modification has no effect on the computational complexity of the algorithm nor its run time, since the number of overall samples to be processed remains the same.

The parameter used to evaluate the effectiveness of the proposed modification, is the Euclidean distance between the true pose of the robot (obtained from the simulator) and the estimated position (obtained from the MCL algorithm). On the other hand, the estimated position is chosen as the mean of the particle set after the resampling step.

## CONCLUSION AND FUTURE WORK

In this work, a modification to the sampling procedure from the proposal distribution is introduced. This modification is based on increasing the number of samples drawn from the proposal distribution. Results show an improvement in the local tracking accuracy over the basic Monte Carlo Localization Algorithm.

In future works, the authors intend to apply this modification to the sampling procedure to other versions of the MCL such as the adaptive MCL algorithm. The improvement in accuracy may be further extended to other variants of the MCL algorithms in order to achieve even better performance.

**REFERENCES**

- [1] S. Kwon et. al., "An Effective Kalman Filter Localization Method for Mobile Robots", Intelligent Robots and Systems, IEEE/RSJ International Conference on,(2006).
- [2] T. Yomchinda, " A method of multirate sensor fusion for target tracking and localization using extended Kalman Filter", Defence Technology - Japan (ACDT), Fourth Asian Conference on, (2017).
- [3] T. Naseer et. al., " Vision-based Markov localization across large perceptual changes", Mobile Robots (ECMR)", European Conference on, (2015).
- [4] L. Guo et. al., " Occupancy grid based urban localization using weighted point cloud", Intelligent Transportation Systems (ITSC), IEEE 19th International Conference on, (2016).
- [5] M. P. Das et. al., " 5-DoF monocular visual localization over grid based floor", Indoor Positioning and Indoor Navigation (IPIN), International Conference on, (2017).
- [6] S. Thrun, D. Fox and W. Burgard," PROBABILISTIC ROBOTICS", (2000).
- [7] D. Fox et. al., "Monte Carlo Localization With Mixture Proposal Distribution", American Association for Artificial Intelligence ([www.aaai.org](http://www.aaai.org)),(2000).
- [8] S. Thrun et al., "Robust Monte Carlo localization for mobile robots", / Artificial Intelligence 128, pp. 99–141, (2001).
- [9] F. Dellaert and D. Fox et. al., "Monte Carlo Localization for Mobile Robots", Proceedings of the IEEE International Conference on Robotics & Automation Detroit, Michigan May (1999).
- [10] W. Burgard et. al., "Monte Carlo Localization: Efficient Position Estimation for Mobile Robots", American Association for Artificial Intelligence, ([www.aaai.org](http://www.aaai.org)), (1999).
- [11] D. Schulz and W. Burgard, "Probabilistic state estimation of dynamic objects with a moving mobile robot", Robotics and Autonomous Systems 34, pp.107–115, (2001).
- [12] Röfer, T. and Jüngel, M., "Vision-Based Fast and Reactive Monte-Carlo Localization", In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA-2003), Taipei, Taiwan.pp. 856-861, (2003).
- [13] P. Pfaff et. al., "Robust Monte-Carlo Localization Using Adaptive Likelihood Models", European Robotics Symposium, STAR 22, pp. 181-194, (2006).
- [14] G. Grisetti et. Al, "Fast and accurate SLAM with Rao–Blackwellized particle filters", Robotics and Autonomous Systems Volume 55, Issue 1, pp. 30-38, (2007).
- [15] D. Meyer-Delius et al., "Maximum-likelihood sample-based maps for mobile robots", Robotics and Autonomous Systems, Robotics and Autonomous Systems 58, pp.133\_139, (2010).
- [16] M. Beinhofer et. al., "Effective landmark placement for accurate and reliable mobile robot navigation", Robotics and Autonomous Systems, Volume 61, Issue 10, , pp. 1060-1069,( 2013).
- [17] R. Zapata, L. Zhang and P. Lepinay. "Self-adaptive Monte Carlo Localization for Mobile Robots Using Range Finders", Robotica, Cambridge University Press, pp.229-244, (2012).
- [18] B. Frank et. al., "Learning object deformation models for robot motion planning", Robotics and Autonomous Systems, Volume 62, Issue 8, pp. 1153-1174, (2014).



- [19] M. Montemerlo, N. Roy, S. Thrun, D. Haehnel, C. Stachniss and J. Glover, "Carmen Robot Navigation Toolkit", <http://carmen.sourceforge.net/home.html>, (2008).
- [20] J. Wolf, W. Burgard and H. Burkhardt, "Robust vision-based localization by combining an image-retrieval system with Monte Carlo localization," *Robotics, IEEE Transactions on* 21 (2), pp. 208–216, (2005).
- [21] M. C. Qomaruddin et. al., "Routing algorithm in legged robot with dynamic programming and Monte Carlo localization," *Engineering Technology and Applications (IES-ETA), International Electronics Symposium on*, (2017).
- [22] A. C. Almeida et. al. "Vision-based monte-carlo localization for humanoid soccer robots", *Robotics Symposium (LARS) and Brazilian Symposium on Robotics (SBR), Latin American*,(2017).
- [23] L. Rui and K. C. Ho, "A Markov Chain Monte Carlo Alternating Minimization Algorithm for Asynchronous Relay Network Localization," *IEEE Wireless Communications Letters*, Volume: 6, Issue: 2, (2017).
- [24] D. J. Peters, "A Bayesian Method for Localization by Multistatic Active Sonar", *IEEE Journal of Oceanic Engineering*, Volume: 42, Issue: 1, (2017).