Robots from Nowhere

Hatice Köse and H. Levent Akın

Boğaziçi University,
Department of Computer Engineering,
34342 Bebek, Istanbul, Turkey
{kose, akin}@boun.edu.tr

Abstract. In this study, a new method called Reverse Monte Carlo Localization (R-MCL) for global localization of autonomous mobile agents in the robotic soccer domain is proposed to overcome the uncertainty in the sensors, environment and the motion model. This is a hybrid method based on both Markov Localization(ML) and Monte Carlo Localization(MCL) where the ML module finds the region where the robot should be and MCL predicts the geometrical location with high precision by selecting samples in this region. The method is very robust and fast and requires less computational power and memory compared to similar approaches and is accurate enough for high level decision making which is vital for robot soccer.

Keywords: Global localization, ML, MCL, robot soccer.

1 Introduction

The localization problem is estimation of the position of a robot relative to the environment, using its actions and sensor readings. Unfortunately these sensors and the environment are uncertain, so the results are typically erroneous and inaccurate. Consequently, localization still remains as a nontrivial and challenging problem and from the simplest geometric calculations which do not consider uncertainty at all, to statistical solutions which cope with uncertainty by applying sophisticated models, many solutions have been proposed for this problem [1], [2], [3]. Although some of these approaches produce remarkable results, due to the nature of the typical environments they are not satisfactory because fast solutions with less memory and computational resources are demanded. This is especially true for a real-time application in a dynamical soccer field using robots with onboard computational resources. Generally, solutions producing precise results suffer from slowness, and high memory usage. Whereas a fast solution in practice typically produces only coarse results. Even when they produce precise local results, some approaches like Kalman filters, fail to find the global position.

This work is a part of the Cerberus Team Robot soccer project [4], and aims to localize the legged robots in the soccer field globally, while solving problems mentioned above. There are a several limitations and assumptions related to the rules of the Robocup [5]. In this work, three approaches to solve this problem were developed in parallel. The first of these new approaches is a new geometrical localization algorithm, which is based on just a single landmark observation at a time. This approach is later extended

D. Nardi et al. (Eds.): RoboCup 2004, LNAI 3276, pp. 594–601, 2005.

to a ML based method. In addition, a novel hybrid approach called Reverse Monte Carlo Localization(R-MCL) combining the ML and MCL methods is designed and implemented.

The organization of the paper is as follows: In the second section, a survey of localization methods is presented. In the third section detailed information about the proposed approach can be found. In the fourth section, the results of the application of proposed approach are present. In the fifth section, conclusions and suggestions for future work are given.

2 Localization Methods

The simplest localization method depending on the range and bearing data is triangulation, which uses geometry to compute a single point that is closest to the current location. But in real world applications a robot can never know where it is exactly because of the uncertainty in its sensors, and the environment. Consequently, several different approaches which estimate the position of robot probabilistically were introduced to integrate this uncertainty into the solutions.

Kalman filter (Kalman-Bucy filter) is a well-known approach for this problem. This filter integrates uncertainty into computations by making the assumption of Gaussian distributions to represent all densities including positions, odometric and sensory measurements. Since only one pose hypothesis can be represented, the method is unable to make global localization, and can not recover from total localization failures [6], [7], [3].

Many works consider Markov localization (ML) [1], [8]. ML is similar to the Kalman filter approach, but it does not make a Gaussian distribution assumption and

Table 1. Comparison of Localization Methods

Method	Capability of global localization	Accuracy	Speed	Memory usage	Robustness to noise	Fast recovery from kidnapping
EKF	no	H	H	L	L	L
ML	yes	L**	M	H**	H	H
MCL	yes	M**	M	H**	M	M
SRL1***	yes	M**	M	M**	M	L
SRL2***	yes	M**	M	M**	L	H
A-MCL	yes	M**	M	M**	XH	H
M-MCL	yes	M**	M	M**	H	H
ML-EKF	yes	M**	M	H**	H	H
Fuzzy*	yes	L**	M	H**	H	H
Geometrical	yes	H**	H	L	L	L
R-MCL	yes	M**	H	H**	H	H

* Fuzzy method is the method implemented in [12].

** These are grid based and sample based methods. So accuracy and memory usage changes with the cell size, and the number of samples used. But they still remain in acceptable ranges.

*** SRL1 and SRL2 differ in their wish to accept additional samples on each noisy observation. This leads fast recovery from kidnapping but increase noise and decrease accuracy.

allows any kind of distribution to be used. Although this feature makes this approach flexible, it adds a computational overhead.

Monte Carlo Localization (MCL) is a version of Markov localization that relies on sample-based representation and the sampling/importance re-sampling algorithm for belief propagation [2], [9]. Beliefs are represented by a set of K weighed samples (particles) which are of type $((x, y, \theta), p)$, where p's are positive numerical weighting factors such that sum of all p is 1. Odometric and sensory updates are similar to ML. Most of the MCL based works suffer from the kidnapping problem, since this approach collapses when the current estimate does not fit observations. There are several extensions to MCL that solve this problem by adding random samples at each iteration. Some of these methods are Sensor Resetting Localization (SRL), Mixture MCL (Mix-MCL), and Adaptive MCL (A-MCL). In SRL, when the likelihood of the current observation is below a threshold, a small fraction of uniformly distributed random samples is added [10]. Mix-MCL additionally weights these samples with current probability density. This method has been developed for extremely accurate sensor information [3]. Adaptive MCL only adds samples when the difference between short-term estimate (slow changing noise level in the environment and the sensors) and the long-term estimate (rapid changes in the likelihood due to a position failure) is above a threshold. The key idea is to use a combination of two smoothed estimates (long term and short term) of the observation likelihoods [3].

ML-EKF method is a hybrid method aiming to make use of the advantages of both methods, taking into consideration the fact that ML is more robust and EKF is more accurate. So this method finds the location of the agent coarsely by grid based ML and then inside this area uses EKF to find a more accurate solution [3].

Although there have been only a few fuzzy logic based approaches, they appear to be promising [11], [12]. In these approaches, the uncertainty in sensor readings (distance and heading to beacons) is represented by fuzzy sets. The above mentioned localization approaches and their capabilities are summarized in Table 1.

Some of the comparisons of the known algorithms in this part are based on [3]. There is an ongoing work for testing of these methods with same data set for comparison.

3 The Proposed Approach

In robot soccer, teams of robots, that are capable of seeing and moving, play matches against each other, and the team with the highest goal score win the match. In order to do this, the player robots must detect their location, the goals, the ball, the members of their team and the opponent team members(optional for high level planning), and place the ball in the opponent team's goal to score a goal. A robot is typically expected to find its own location using the six distinguishable unique landmarks in the field, and then use this information to find the location of the ball and goal. Consequently, localization is a vital problem for robot soccer.

Since, as discussed in section 2, no single method satisfies all the needs in terms of single robot localization, three methods including a hybrid algorithm which tend to integrate the advantageous parts of single methods and overcome the deficiencies were developed in this work.

3.1 Geometrical Localization

The geometrical localization method assumes the input data is measured exactly (does not contain noise), and therefore does not need any error modeling. Our previous algorithm in [13] required at least two landmarks to be seen at any time to calculate the position accurately. Although it worked also for the one landmark case, it could not give satisfactory results. This new method is designed to work with one landmark information which is much more realistic within the new field sizes. So even if the robot sees more than one landmark, they are treated separately and one-landmark information is used at each step. The ratio of the distance between the predicted location and the observed location of the landmark is used to predict the new x and y coordinates of the robot.

The bearing is also found by using the new predicted x and y coordinates. Whenever a new visual data comes, the new position is calculated based on the measurement and old position. A point between the newly measured position and the old position is taken as the new position. This new position is placed between the two positions proportional to the belief of the robot on them. The assumption here is that: The more you believe in a position the closer you are to that position. This is used to reduce the effect of inaccurate measurements on the new position. When the odometric data arrives, the position is blurred among the moved distance and heading. The bearing is added to the original heading and it is normalized to give the new heading of the robot. The odometric data consists of the distance moved forward, left and the bearing of turn. This method assumes that the measurements are exact, or noise is below a threshold.

3.2 Markov Localization

The assumption made about error in the previous subsection is not correct in general. So we need a method to handle the uncertainty in the visual and odometric data. Since ML is a grid-based algorithm, it gives a coarse but robust result. Unfortunately it requires complex computations since at every step all grid cells should be taken into account. Although the accuracy of the result might not be sufficient for implementing high level planning which is required for robot soccer. It can converge faster than the sample based algorithms. So a ML based method is designed and implemented. This method works in a similar manner as the geometrical localization algorithm (Figure 1):

```
For all grid cells
        If visual data available
                Apply visual update
        Else if odometric data availabla
                Apply odometric update
        Calculate probability of each cell according to ML
        // use current beliefs and probability, and newly calculated belief and probability
        //of each cell and calculate the weighted average
return the best cells (with maximum probability)
```

Fig. 1. The ML based proposed algorithm

Unfortunately, as the uncertainty increases, the maximum probability decreases and the number of grid cells with the maximum probability increases, so the accuracy of the final position decreases. One solution is to use the averages of the centers of the cells, as the final position. But this parameter will not cover all space uniformly. So it would be useful to use a sample based method within the ML method, in order to increase coverage speed and accuracy. If a method is used to update only several cells but not all at each update, then the computational complexity would decrease drastically, too.

3.3 R-MCL

As indicated in the previous method, ML is robust and converges fast, but coarse and computationally complex. On the other hand, sample based MCL is not as computationally complex as ML, and gives accurate results. However, it can not converge to a position as fast as ML, especially in the case of an external impact on the position of the robot (such as kidnapping). In addition, the number of samples to be used is generally kept very high to cover all space and converge to the right position. There are several extensions for adaptive sample size usage, but these still do not solve the slow coverage problem. So it might be useful to converge to several cells by ML or another grid based method, then inside these bulk of grids, produce a limited number of samples to find the final position. The average of these samples would give the final position and the standard deviation might give the uncertainty of the final position as in the MCL based methods. The algorithm in Figure 2 simply works as shown in Figure 3:

In the original MCL, the number of samples is increased to decrease bias in the result. In R-MCL since we converge by selecting cells with maximum probability, so the bias is already decreased, therefore we do not need this to decrease bias.

After testing this version, some improvements were done on the current version of R-MCL and ML. In the modified ML, not only the distance but also the bearing information is used to find the best grids, so the number of chosen grids decrease and confidence increases. As a result of these modifications, the accuracy of the results improved considerably. Later when the samples are drawn, also the best samples are selected using distance and the bearing from these very good cells, and their average is returned as the current pose. Notice that, samples are taken into consideration only when the position reaches to a certainty level, in other words the number of chosen cells are

```
Apply visual/odometric update to all cells
Choose grid cells with probability > threshold as in ML //only choosing the cells with maximum probability might not be
//adequate due to test field's conditions
Apply resampling
        Calculate the number of cells to be produced according to your level of uncertainty
        Produce random samples of this amount from the chosen cells
        Pick up samples with probability> threshold from this sample set
Find the average of the positions of chosen samples and return as final position
Find standard deviation of the positions of samples and return as the uncertainty of the final position
```

Fig. 2. The R-MCL based proposed algorithm

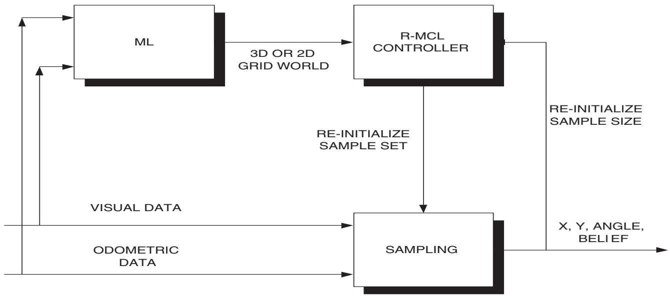

Fig. 3. The R-MCL working schema

below a limit(e.g. 50), and there is at least one very good cell which is below or equal to the minimum error in both distance and bearing limitations. Also if there are no samples which satisfy the minimum bearing and distance error condition then the results of ML are used instead. Also the bearing of the new pose is found by the ML module inside the R-MCL because it is more accurate and robust.

4 Tests and Results

A simulator was developed which is used to produce realistic data for testing the implementations. It produces both odometric and visual data according to the user's choice, besides it enables the addition of random noise to the produced data optionally, to make it more realistic. The new algorithms are tested on a set of tests based on fixed paths in the field. In the first group of tests there is active vision and in the second, the robot could take information about one landmark so it is a very hard and challenging case. Also noise with a magnitude of 20 cm (chosen very high-twice the cell size- to observe the effects easily) is added in some tests, and likewise odometry is included in a group of tests. The tests of R-MCL are repeated 50 times each, because the samples are drawn randomly and this affects the accuracy of the results. The average and standard deviation of the errors that are produced during the tests are presented in the Table 2. As seen from the

Table 2. Experimental results

test no	active vision	noise	odometry	geometrical	ml	r-mcl
1	yes	no	no	18.13±35.60	4.27±1.97	3.68±1.84
2	yes	yes	no	20.12±33.56	16.63±24.10	17.22±24.08
3	yes	no	yes	18.13±35.61	4.27±1.97	3.72±1.55
4	yes	yes	yes	23.48±31.70	11.30±2.73	9.13±3.46
5	no	no	no	168.60±1.68	5.70±0.68	4.06±1.75
6	no	yes	no	169.29±2.65	35.28±27.94	34.41±25.12
7	no	no	yes	168.76±1.75	5.70±0.68	3.37±2.49
8	no	yes	yes	167.94±3.01	5.69±0.68	3.90±1.77

test results, when active vision is used, the geometrical method performs relatively well. ML performs very good, both in case of passive vision and noise. R-MCL outperforms all, and its performance is close to ML as predicted, in the problematic cases, since it uses the output of ML in these cases. The effect of odometry on the results of ML seems ignorable since the step size is ignorable compared to the cell size. In currently ongoing works, these effects are tested in details.

5 Conclusions

Localization in a totally unknown area is a very hard task for autonomous mobile robots. This work aims to propose a fast, reliable, computationally and resource efficient solution to global localization problem. The solution should be successful in environments like the Robocup Games and the challenges which require very high accuracy and speed. For this reason in this paper several new localization algorithms are developed. The first method is a new geometrical localization method which is based on one observation at a time. This property makes this method more robust and realistic. Extending this idea, a ML based method is also developed. Next, to make use of the robustness of ML but to make the results more precise, a hybrid method called R-MCL method is implemented. This method recovers from kidnapping in a very fast manner and it is very robust to even high levels of noise and inadequate environmental information. It is higly accurate and its accuracy can be improved by improving resampling and choosing the best cells and samples. It performs quite well when compared with the outperforming methods such as ML-EKF, and A-MCL, and there is an ongoing research for testing it with these methods on the same data set from ERS 210 quadrupled robots.

Acknowledgements

This project is supported by the Boğaziçi University Foundation and by Boğaziçi University Research Fund project 03A101D.

References

1. W. Burgard, D. Fox, ,D. Hennig, and T. Schmidt, "Estimating the Absolute Position of a Mobile Robot Using Position Probability Grids," *Institut für Informatik III., Universtiat Bonn, Proc. of the Fourteenth National Conference on Artificial Intelligence (AAAI-96)*, 1996.
2. S. Thrun, D. Fox, W. Burgard, and F. Dellaert, "Robust Monte Carlo Localization for Mobile Robots," *Elsevier, Artificial Intelligence,*128, pp. 99-141. 2001.
3. J.S. Gutmann, and D. Fox, "An Experimental Comparison of Localization Methods Continued," *In Proc. of the 2002 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS'02),* Page: 454-459, Lausanne, Switzerland October 2002.
4. Cerberus, *http://robot.cmpe.boun.edu.tr/aibo/home.php3,* 2003.
5. Robocup, , *http://www.robocup2003.org/,* 2003.
6. A.W. Stroupe, T. Balch, "Collaborative Probabilistic Constraint Based Landmark Localization," *Proceedings of the 2002 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems IROS'02),* Page: 447-452, Lausanne, Switzerland, October 2002.

7. A.W. Stroupe, K. Sikorski, and T. Balch, "Constraint-Based Landmark Localization," *Proceedings of 2002 RoboCup Symposium,* 2002.
8. D. Fox, W. Burgard, and S. Thrun, "Markov Localization for Mobile Robots in Dynamic Environments" *Journal of Artificial Intelligence Research 11,*pp. 391-427., 1999.
9. D. Schulz, and W. Burgard, "Probabilistic State Estimation of Dynamic Objects with a Moving Mobile Robot," *Elsevier, Robotics and Autonomous Systems 34,*pp. 107-115. 2001.
10. S. Lenser, and M. Veloso, "Sensor Resetting Localization for Poorly Modelled Mobile Robots" *Proceedings of ICRA 2000, IEEE,* 2000.
11. P. Buschka, A. Saffiotti, and Z. Wasik, "Fuzzy Landmark-Based Localization for a Legged Robot" *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS),* Takamatsu, Japan, pp. 1205-1210, 2000.
12. H. Kose, S. Bayhan, and H. L.Akin, "A Fuzzy Approach to Global Localization in the Robot Soccer Domain" *IJCI Proceedings of International XII Turkish Symposium on Artificial Intelligence and Neural Networks (TAINN 2003),* ISSN 1304-2386 Vol:1, No:1 pp.1-7, July 2003.
13. H.L. Akin, A. Topalov, and O. Kaynak, "Cerberus 2001 Team Description," *Robocup 2001: Robot Soccer World Cup V, Birk, A., Coradeschi, S., and Tadokoro, S. (Eds.),* LNAI 2377, pp.689-692, Springer Verlag. 2001.