Collaborative Multi-Robot Localization

Dieter Fox[†], Wolfram Burgard[‡], Hannes Kruppa[††], Sebastian Thrun[†]

[†] School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

[‡] Computer Science Department III
University of Bonn
D-53117 Bonn, Germany

[††] Department of Computer Science
ETH Zurich
CH-8092 Zurich, Switzerland

Abstract. This paper presents a probabilistic algorithm for collaborative mobile robot localization. Our approach uses a sample-based version of Markov localization, capable of localizing mobile robots in an any-time fashion. When teams of robots localize themselves in the same environment, probabilistic methods are employed to synchronize each robot's belief whenever one robot detects another. As a result, the robots localize themselves faster, maintain higher accuracy, and high-cost sensors are amortized across multiple robot platforms. The paper also describes experimental results obtained using two mobile robots. The robots detect each other and estimate their relative locations based on computer vision and laser range-finding. The results, obtained in an indoor office environment, illustrate drastic improvements in localization speed and accuracy when compared to conventional single-robot localization.

1 Introduction

Sensor-based robot localization has been recognized as one of the fundamental problems in mobile robotics. The localization problem is frequently divided into two subproblems: *Position tracking*, which seeks to compensate small dead reckoning errors under the assumption that the initial position of the robot is known, and *global self-localization*, which addresses the problem of localization with no a priori information about the robot position. The latter problem is generally regarded as the more difficult one, and recently several approaches have provided sound solutions to this problem. In recent years, a flurry of publications on localization—which includes a book solely dedicated to this problem [2]—document the importance of the problem. According to Cox [8], "Using sensory information to locate the robot in its environment is the most fundamental problem to providing a mobile robot with autonomous capabilities."

However, virtually all existing work addresses localization of a *single* robot only. At first glance, one could solve the problem of localizing N robots by localizing each robot *independently*, which is a valid approach that might yield reasonable results in many environments. However, if robots can detect each other, there is the opportunity to do better. When a robot determines the location of another robot relative to its own, both robots can refine their internal believes based on the other robot's estimate, hence improve their localization accuracy. The ability to exchange information during localization is particularly attractive in the context of global localization, where each sight of another robot can reduce the uncertainty in the estimated location dramatically.

The importance of exchanging information during localization is particularly striking for heterogeneous robot teams. Consider, for example, a robot team where some robots are equipped with expensive, high accuracy sensors (such as laser range-finders), whereas others are only equipped with low-cost sensors such as ultrasonic range finders. By transferring information across multiple robots, high-accuracy sensor information can be leveraged. Thus, collaborative multi-robot localization facilitates the amortization of high-end, high-accuracy sensors across teams of robots. Thus, phrasing the problem of localization as a collaborative one offers the opportunity of improved performance from less data.

This paper proposes an efficient probabilistic approach for collaborative multi-robot localization. Our approach is based on *Markov localization* [23, 27, 16, 6], a family of probabilistic approaches that have recently been applied with great practical success to single-robot localization [4, 3, 30]. In contrast to previous research, which relied on grid-based or coarse-grained topological representations, our approach adopts a sampling-based representation [10, 12], which is capable of approximating a wide range of belief functions in real-time. To transfer information across different robotic platforms, probabilistic "detection models" are employed to model the robots' abilities to recognize each other. When one robot detects another the individual believes of the robots are synchronized, thereby reducing the uncertainty of both robots during localization. While our approach is applicable to any sensor capable of (occasionally) detecting other robots, we present an implementation that integrates color images and proximity data for robot detection.

In what follows, we will first introduce the necessary statistical mechanisms for multi-robot localization, followed by a description of our sampling-based Monte Carlo localization technique in Section 3. In Section 4 we present our vision-based method to detect other robots. Experimental results are reported in Section 5. Finally, related work is discussed in Section 6, followed by a discussion of the advantages and limitations of the current approach.

2 Multi-Robot Localization

Throughout this paper, we adopt a probabilistic approach to localization. Probabilistic methods have been applied with remarkable success to single-robot localization [23, 27, 16, 6], where they have been demonstrated to solve problems like global localization and localization in dense crowds.

Let us begin with a mathematical derivation of our approach to multi-robot localization. Let N be the number of robots, and let d_n denote the data gathered by the n-th robot, with $1 \leq n \leq N$. Each d_n is a sequence of three different types of information:

1. **Odometry measurements**, denoted by a, specify the relative change of the position according to the robot's wheel encoders.

2. **Environment measurements**, denoted by o, establish the reference between the robot's local coordinate frame and the environment's frame of reference. This information typically consists of range measurements or camera images.

3. **Detections**, denoted by r, indicate the presence or absence of other robots. Below, in our experiments, we will use a combination of visual sensors (color camera) and range finders for robot detection.

2.1 Markov Localization

Before turning to the topic of this paper—collaborative multi-robot localization—let us first review a common approach to single-robot localization, which our approach is built upon: Markov localization (see [11] for a detailed discussion). Markov localization uses only dead reckoning measurements a and environment measurements o; it ignores detections r. In the absence of detections (or similar information that ties the position of one robot to another), information gathered at different platforms cannot be integrated. Hence, the best one can do is to localize each robot individually, i.e. independently of all others.

The key idea of Markov localization is that each robot maintains a belief over its position. Let $Bel_n^{(t)}(L)$ denote the belief of the n-th robot at time t. Here L denotes the random variable representing the *robot position* (we will use the terms *position* and *location* interchangeably), which is typically a three-dimensional value composed of a robot's x-y position and its orientation θ. Initially, at time $t = 0$, $Bel_n^{(0)}(L)$ reflects the initial knowledge of the robot. In the most general case, which is being considered in the experiments below, the initial position of all robots is unknown, hence $Bel_n^{(0)}(L)$ is initialized by a uniform distribution.

At time t, the belief $Bel_n^{(t)}(L)$ is the posterior with respect to all data collected up to time t:

$$Bel_n^{(t)}(L) = P(L_n^{(t)} \mid d_n^{(t)}) \tag{1}$$

where $L_n^{(t)}$ denotes the position of the n-th robot at time t, and $d_n^{(t)}$ denotes the data collected by the n-th robot *up to* time t. By assumption, the most recent sensor measurement in $d_n^{(t)}$ is either an odometry or an environment measurement. Both cases are treated differently, so let's consider the former first:

1. **Sensing the environment:** Suppose the last item in $d_n^{(t)}$ is an environment measurement, denoted $o_n^{(t)}$. Using the Markov assumption (and exploiting that the robot position does not change when the environment is sensed), the belief is updated using the following *incremental* update equation:

$$Bel_n^{(t)}(L = l) \longleftarrow \alpha \; P(o_n^{(t)} \mid L_n^{(t)} = l) \; Bel_n^{(t-1)}(L = l) \tag{2}$$

Here α is a normalizer which ensures that $Bel_n^{(t)}(L)$ sums up to one. Notice that the posterior belief of being at location l after incorporating $o_n^{(t)}$ is obtained by multiplying the observation likelihood $P(o_n^{(t)} \mid L_n^{(t)} = l)$ with the prior belief. This likelihood is also called the *environment perception model* of robot n. Typical models for different types of sensors are described in [11, 9, 18].

2. **Odometry:** Now suppose the last item in $d_n^{(t)}$ is an odometry measurement, denoted $a_n^{(t)}$. Using the Theorem of Total Probability and exploiting the Markov property, we obtain the following incremental update scheme:

$$Bel_n^{(t)}(L = l) \longleftarrow \int P(L_n^{(t)} = l \mid a_n^{(t-1)}, L_n^{(t-1)} = l') \; Bel_n^{(t-1)}(L = l') \; dl' \tag{3}$$

Here $P(L_n^{(t)} = l \mid a_n^{(t-1)}, L_n^{(t-1)} = l')$ is called the *motion model* of robot n. In the remainder, this motion model will be denoted as $P(l \mid a_n, l')$ since it is assumed to be independent of the time t. It is basically a model of robot kinematics annotated with uncertainty and it generally has two effects: first, it shifts the probabilities according to the measured motion and second it convolves the probabilities in order to deal with possible errors in odometry coming from slippage etc. (see e.g. [12]).

These equations together form the basis of Markov localization, an incremental probabilistic algorithm for estimating robot positions. As noticed above, Markov localization has been applied with great practical success to mobile robot localization. However, it is only designed for single-robot localization, and cannot take advantage of robot detection measurements.

2.2 Multi-Robot Markov Localization

The key idea of multi-robot localization is to integrate measurements taken at different platforms, so that each robot can benefit from data gathered by robots other than itself. At first glance, one might be tempted to maintain a single belief over all robots' locations, i.e.,

$$L = \{L_1, \ldots, L_N\} \tag{4}$$

Unfortunately, the dimensionality of this vector grows with the number of robots: Since each robot position is three-dimensional, L is of dimension $3N$. Distributions over L are, hence, exponential in the number of robots. Thus, modeling the joint distribution of the positions of all robots is infeasible for larger values of N.

Our approach maintains *factorial* representations; i.e., each robot maintains its own belief function that models only its own uncertainty, and occasionally, e.g., when a robot sees another one, information from one belief function is transfered from one robot to another. The factorial representation assumes that the distribution of L is the product of its N marginal distributions:

$$P(L_1^{(t)}, \ldots, L_N^{(t)} \mid d^{(t)}) = P(L_1^{(t)} \mid d^{(t)}) \cdot \ldots \cdot P(L_N^{(t)} \mid d^{(t)}) \tag{5}$$

Strictly speaking, the factorial representation is only approximate, as one can easily construct situations where the independence assumption does not hold true. However, the factorial representation has the advantage that the estimation of the posteriors is conveniently carried out locally on each robot. In the absence of detections, this amounts to performing Markov localization independently for each robot. Detections are used to provide additional constraints between the estimated pairs of robots, which will lead to refined local estimates.

To derive how to integrate detections into the robots' beliefs, let us assume the last item in $d_n^{(t)}$ is a detection variable, denoted $r_n^{(t)}$. For the moment, let us assume this is the only such detection variable in $d^{(t)}$, and that it provides information about the location of the m-th robot relative to robot n (with $m \neq n$). Then

$$\begin{aligned} Bel_m^{(t)}(L = l) &= P(L_m^{(t)}{=}l \mid d^{(t)}) \\ &= P(L_m^{(t)}{=}l \mid d_m^{(t)})\ P(L_m^{(t)}{=}l \mid d_n^{(t)}) \\ &= P(L_m^{(t)}{=}l \mid d_m^{(t)}) \int P(L_m^{(t)}{=}l \mid L_n^{(t)}{=}l', r_n^{(t)}) P(L_n^{(t)}{=}l' \mid d_n^{(t-1)})\, dl' \end{aligned} \tag{6}$$

which suggests the incremental update equation:

$$Bel_m^{(t)}(L=l) \longleftarrow Bel_m^{(t)}(L=l) \int P(L_m^{(t)}=l \mid L_n^{(t)}=l', r_n^{(t)})\; Bel_n^{(t)}(L=l')\; dl' \quad (7)$$

In this equation the term $P(L_m^{(t)}=l \mid L_n^{(t)}=l', r_n^{(t)})$ is the robot perception model. A typical example of such a model for visual robot detection is described in Section 4. Of course, Eq. (7) is only an approximation, since it makes certain independence assumptions (it excludes that a sensor reports "I saw a robot, but I cannot say which one"), and strictly speaking it is only correct if there is only a single r in the entire run. However, this gets us around modeling the joint distribution $P(L_1, \ldots, L_N \mid d)$, which is computationally infeasible as argued above. Instead, each robot basically performs single-robot Markov localization with these additional probabilistic constrains, hence estimates the marginal distributions $P(L_n|d)$ separately.

The reader may notice that, by symmetry, the same detection can be used to constrain the n-th robot's position based on the belief of the m-the robot. The derivation is omitted since it is fully symmetrical.

3 Monte Carlo Localization

The previous section left open how the belief is represented. In general, the space of all robot positions is continuous-valued and no parametric model is known that would accurately model arbitrary beliefs in such robotic domains. However, practical considerations make it impossible to model arbitrary beliefs using digital computers.

3.1 Single Robot MCL

The key idea here is to approximate belief functions using a Monte Carlo method. More specifically, our approach is an extension of **M**onte **C**arlo **L**ocalization (MCL), which was shown to be an extremely efficient and robust technique for single robot position estimation (see [10, 12] for more details). MCL is a version of Markov localization that relies on a sample-based representation and the sampling/importance re-sampling algorithm for belief propagation [25]. MCL represents the posterior beliefs $Bel_n(L)$ by a set $S = \{s_i \mid i = 1..K\}$ of K weighted random samples or *particles*[1]. Samples in MCL are of the type

$$s_i = \langle\langle x_i, y_i, \theta_i\rangle, p_i\rangle \quad (8)$$

where $\langle x_i, y_i, \theta_i\rangle$ denote a robot position, and $p_i \geq 0$ is a numerical weighting factor, analogous to a discrete probability. For consistency, we assume $\sum_{i=1}^{K} p_i = 1$. In analogy with the general Markov localization approach outlined in Section 2, MCL proceeds in two phases:

1. **Robot motion.** When a robot moves, MCL generates K new samples that approximate the robot's position after the motion command. Each sample is generated by

[1] A sample set constitutes a discrete distribution. However, under appropriate assumptions (which happen to be fulfilled in MCL), such distributions smoothly approximate the "correct" one at a rate of $1/\sqrt{K}$ as K goes to infinity [29].

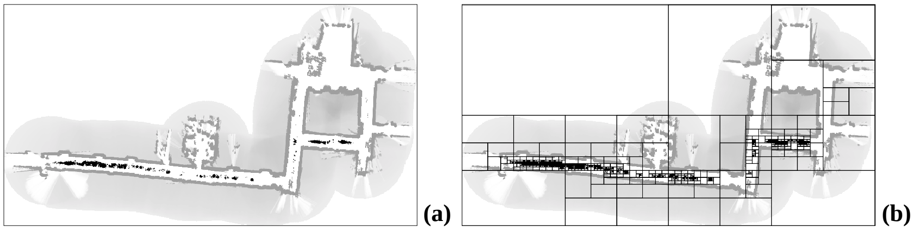

Fig. 1. (a) Map of the environment along with a sample set representing the robot's belief during global localization, and (b) its approximation using a density tree.

randomly drawing a sample from the previously computed sample set, with likelihood determined by their p-values. Let l' denote the position of such a sample. The new sample's position l is then generated by producing a single, random sample from $P(l \mid a, l')$, using the action a as observed. The p-value of the new sample is K^{-1}. An algorithm to perform this re-sampling process efficiently in $O(K)$ time is given in [7].
2. **Environment measurements** are incorporated by re-weighting the sample set, which is analogous to Bayes rule in Markov localization. More specifically, let $\langle l, p\rangle$ be a sample. Then, in analogy to Eq. (2) the updated sample is $\langle l, \alpha\ P(o \mid l)p\rangle$ where o is a sensor measurement, and α is a normalization constant that enforces $\sum_{i=1}^{K} p_i = 1$. The incorporation of sensor readings is typically performed in two phases, one in which p is multiplied by $P(o \mid l)$, and one in which the various p-values are normalized.

3.2 Multi-Robot MCL

The extension of MCL to collaborative multi-robot localization is *not* straightforward. This is because under our factorial representation, each robot maintains its own, local sample set. When one robot detects another, both sample sets are synchronized according to Eq. (7). Notice that this equation requires the multiplication of two *densities* which means that we have to establish a correspondence between the individual samples in $Bel(L_m)$ and the density representing the robot detection.

To remedy this problem, our approach transforms sample sets into density functions using *density trees* [17, 22]. These methods approximate sample sets using piecewise constant density functions represented by a tree. The resolution of the tree is a function of the densities of the samples: the more samples exist in a region of space, the more fine-grained the tree representation. Figure 1 shows an example sample set along with the tree generated from this set. Our specific algorithm grows trees by recursively splitting in the center of each coordinate axis, terminating the recursion when the number of samples is smaller than a pre-defined constant. After the tree is grown, each leaf's density is given by the quotient of the sum of the weights p of all samples that fall into this leaf, divided by the volume of the region covered by the leaf. The latter amounts to maximum likelihood estimation of (piecewise) constant density functions.

To implement the update equation above, our approach approximates the density

$$\int P(L_m^{(t)} = l \mid L_n^{(t)} = l', r_n^{(t)})\ Bel_n^{(t)}(L = l')\ dl' \qquad (9)$$

using samples, just as described above. The resulting sample set is then transformed into a density tree. These density values are then multiplied into the weights (importance

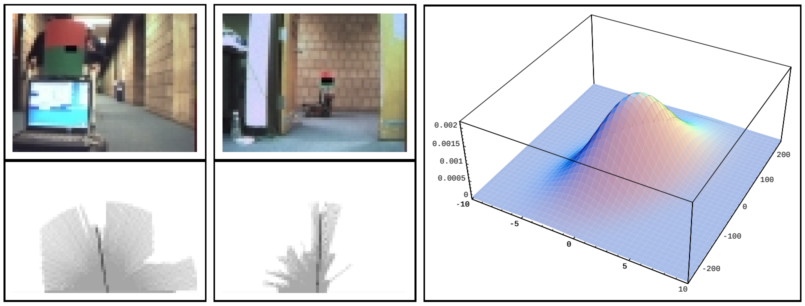

Fig. 2: Examples of successful robot detections and Gaussian density representing the robot perception model. The x-axis represents the deviation of relative angle and the y-axis the uncertainty in the distance between the two robots.

factors) of the samples in $Bel(L_m)$, effectively multiplying both density functions. The result is a refined density for the m-th robot, reflecting the detection and the belief of the n-th robot.

4 Visual Robot Detection

To implement collaborative multi-robot localization, robots must possess the ability to sense each other. The crucial component is the detection model $P(L_m = l \mid L_n = l', r_n)$ which describes the conditional probability that robot m is at location l, given that robot n is at location l' and perceives robot m with measurement r_n. In this section, we briefly describe one possible detection method which integrates camera and range information to estimate the relative position of robots.

Our implementation uses camera images to detect other robots and extracts from these images the relative direction of the other robot. After detecting another robot and its relative angle, it uses laser ranger finder scans to determine its distance. Figure 2 shows two examples of camera images taken by one of the robots. Each image shows another robot, marked by a unique, colored marker to facilitate the recognition. Even though the robot is only shown with a fixed orientation in this figure, the markers can be detected regardless of a robot's orientation. The small black rectangles, superimposed at the center of each marker in the images in Figure 2, illustrate the center of the marker as identified by this visual routine. The bottom row in Figure 2 shows laser scans for the example situations depicted in the top row of the same figure. Each scan consists of 180 distance measurements with approx. 5 cm accuracy, spaced at 1 degree angular distance. The dark line in each diagram depicts the extracted location of the robot in polar coordinates, relative to the position of the detecting robot. The scans are scaled for illustration purposes.

The Gaussian distribution shown in Figure 2 models the error in the estimation of a robot's location. Here the x-axis represents the angular error, and the y-axis the distance error. This Gaussian has been obtained through maximum likelihood estimation based on training data (see [13] for more details). As is easy to be seen, the Gaussian is zero-centered along both dimensions, and it assigns low likelihood to large errors. Please note that our detection model additionally considers a 6.9% chance to erroneously detecting a robot when there is none.

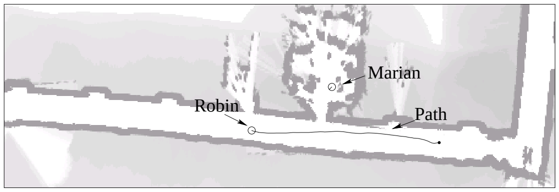

Fig. 3: Map of the environment along with a typical path taken by Robin during an experiment.

5 Experimental Results

Our approach was evaluated using two Pioneer robots (Robin and Marian) marked optically by a colored marker, as shown in Figure 2. The central question driving our experiments was: *Can cooperative multi-robot localization significantly improve the localization quality, when compared to conventional single-robot localization?*

Figure 3 shows the setup of our experiments along with a part of the occupancy grid map [31] used for position estimation. Marian operates in our lab, which is the cluttered room adjacent to the corridor. Because of the non-symmetric nature of the lab, the robot knows fairly well where it is (the samples representing Marian's belief are plotted in Figure 4 (a)). Figure 3 also shows the path taken by Robin, which was in the process of global localization. Figure 5 (a) represents the typical belief of Robin when it passes the lab in which Marian is operating. Since Robin already moved several meters in the corridor, it developed a belief which is centered along the main axis of the corridor. However, the robot is still highly uncertain about its exact location within the corridor and even does not know its global heading direction. Please note that due to the lack of features in the corridor the robots generally have to travel a long distance until they can resolve ambiguities in the belief about their position.

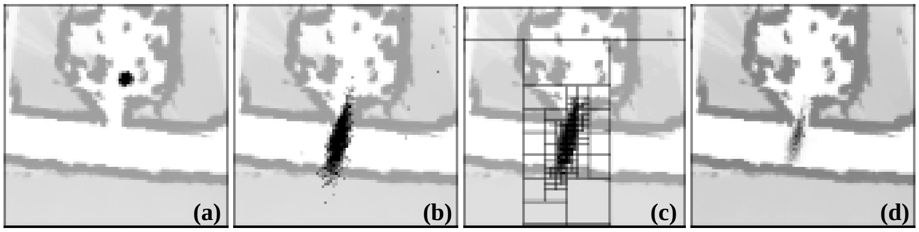

Fig. 4. Detection event: (a) Sample set of Marian as it detects Robin in the corridor. (b) Sample set reflecting Marian's belief about Robin's position (see *robot detection model* in Eq. (7)). (c) Tree-representation of this sample set and (d) corresponding density.

The key event, illustrating the utility of cooperation in localization, is a detection event. More specifically, Marian, the robot in the lab, detects Robin, as it moves through the corridor (see right camera image and laser range scan of Figure 2 for a characteristic measurement of this type). Using the detection model described in Section 4, Marian generates a new sample set as shown in Figure 4 (b). This sample set is converted into a density using density trees (see Figure 4 (c) and (d)). Marian then transmits this density to Robin which integrates it into its current belief. The effect of this integration on

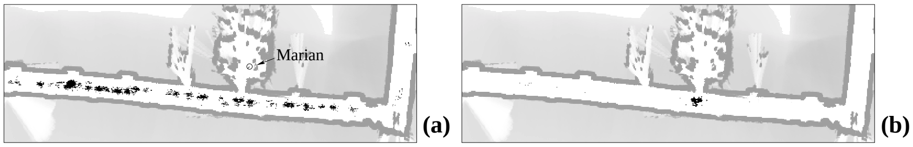

Fig. 5. Sample set representing Robin’s belief (a) as it passes Marian and (b) after incorporating Marian’s measurement.

Robin’s belief is shown in Figure 5 (b). It shows Robin’s belief after integrating the density representing Marian’s detection. As this figure illustrates, this single incident almost completely resolves the uncertainty in Robin’s belief.

We conducted ten experiments of this kind and compared the performance to conventional MCL for single robots which ignores robot detections. To measure the performance of localization we determined the true locations of the robot by measuring the starting position of each run and performing position tracking off-line using MCL. For each run, we then compared the estimated positions (please note that here the robot was not told it’s starting location) with the positions on the reference path. The results are summarized in Figure 6.

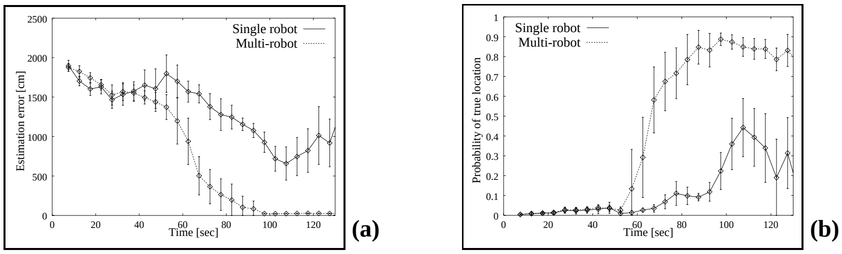

Fig. 6. Comparison between single-robot localization and localization making use of robot detections. The x-axis represents the time and the y-axis represents (a) the estimation error and (b) the probability assigned to the true location.

Figure 6 (a) shows the estimation error as a function of time, averaged over the ten experiments, along with their 95% confidence intervals (bars). Figure 6 (b) shows the probability assigned to the true locations of the robot, obtained by summing over the weighting factors of the samples in an area ± 50 cm and ± 10 degrees around the true location. As can be seen in both figures, the quality of position estimation increases much faster when using multi-robot localization. Please note that the detection event typically took place 60-100 seconds after the start of an experiment.

Obviously, this experiment is specifically well-suited to demonstrate the advantage of detections in multi-robot localization, since the robots’ uncertainties are somewhat orthogonal, making the detection highly effective. A more thoroughly evaluation of the benefits of MCL will be one topic of future research.

6 Related Work

Mobile robot localization has frequently been recognized as a key problem in robotics with significant practical importance. A recent book by Borenstein, Everett, and Feng [2] provides an overview of the state-of-the-art in localization.

Almost all existing approach address single-robot localization only. Moreover, the vast majority of approaches is incapable of localizing a robot globally; instead, they are designed to track the robot's position by compensating small odometric errors. Thus, they differ from the approach described here in that they require knowledge of the robot's initial position. Furthermore, they are not able to recover from global localizing failures. Probably the most popular method for tracking a robot's position is Kalman filtering [15, 20, 21, 26, 28], which represents the belief by a uni-modal Gaussian distribution. These approaches are unable to localize robots under global uncertainty. Recently, several researchers proposed *Markov localization*, which enables robots to localize themselves under global uncertainty [6, 16, 23, 27]. Global approaches have two important advantages over local ones: First, the initial location of the robot does not have to be specified and, second, they provide an additional level of robustness, due to their ability to recover from localization failures. Among the global approaches those using metric representations of the space such as MCL and [6, 5] can deal with a wider variety of environments than the methods relying on topological maps. For example, they are not restricted to orthogonal environments containing pre-defined features such as corridors, intersections and doors.

The issue of cooperation between multiple mobile robots has gained increased interest in the past. In this context most work on localization has focused on the question of how to reduce the odometry error using a cooperative team of robots [19, 24, 1]. While these approaches are very successful in reducing the odometry error, none of them incorporates environmental feedback into the estimation. Even if the initial locations of all robots are known, they ultimately will get lost although at a slower pace than a comparable single robot. The problem addressed here differs in that we are interested in collaborative localization in a global frame of reference, not just reducing odometry error.

7 Conclusions

In this paper, we presented a probabilistic method for collaborative mobile robot localization. At its core, our approach uses probability density functions to represent the robots' estimates as to where they are. To avoid exponential complexity in the number of robots, a factorial representation is advocated where each robot maintains its own, local belief function. A fast, universal sampling-based scheme is employed to approximate beliefs. The probabilistic nature of our approach makes it possible that teams of robots perform *global localization*, i.e., they can localize themselves from scratch without initial knowledge as to where they are.

During localization, detections are used to introduce additional probabilistic constraints between the individual belief states of the robots. As a result, our approach makes it possible to amortize data collected at multiple platforms. This is particularly attractive for heterogeneous robot teams, where only a small number of robots may be equipped with high-precision sensors.

Experimental results, carried out in a typical office environment, demonstrate that our approach can reduce the uncertainty in localization significantly, when compared to conventional single robot localization. Thus, when teams of robots are placed in a known environment with unknown starting locations, our approach can yield much faster localization at approximate equal computation costs and relatively small communication overhead.

The approach described here possesses several limitations that warrant future research. First, in our current system, only "positive" detections are processed. *Not seeing another robot* is also informative, and the incorporation of such negative detections is generally possible in the context of our statistical framework. Another limitation of the current approach arises from the fact that our detection approach must be able to identify individual robots. The ability to integrate over the beliefs of all other robots is a natural extension of our approach although it increases the amount of information communicated between the robots. Furthermore, the collaboration described here is purely passive, in that robots combine information collected locally, but they do not change their course of action so as to aid localization as, for example, described in [14]. Finally, the robots update their belief instantly whenever they perceive another robot. In situations in which both robots are highly uncertain at the time of the detection it might be more appropriate to delay the update and synchronize the beliefs when one robot has become more certain about its position.

Despite these open research areas, our approach provides a sound statistical basis for information exchange during collaborative localization, and empirical results illustrate its appropriateness in practice. While we were forced to carry out this research on two platforms only, we conjecture that the benefits of collaborative multi-robot localization increase with the number of available robots.

References

1. J. Borenstein. Control and kinematic design of multi-degree-of-freedom robots with compliant linkage. *IEEE Transactions on Robotics and Automation*, 1995.
2. J. Borenstein, B. Everett, and L. Feng. *Navigating Mobile Robots: Systems and Techniques*. A. K. Peters, Ltd., Wellesley, MA, 1996.
3. W. Burgard, A. B. Cremers, D. Fox, D. Hähnel, G. Lakemeyer, D. Schulz, W. Steiner, and S. Thrun. Experiences with an interactive museum tour-guide robot. *Artificial Intelligence*, 2000. accepted for publication.
4. W. Burgard, A.B. Cremers, D. Fox, D. Hähnel, G. Lakemeyer, D. Schulz, W. Steiner, and S. Thrun. The interactive museum tour-guide robot. In *Proc. of the National Conference on Artificial Intelligence (AAAI)*, 1998.
5. W. Burgard, A. Derr, D. Fox, and A.B. Cremers. Integrating global position estimation and position tracking for mobile robots: the Dynamic Markov Localization approach. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 1998.
6. W. Burgard, D. Fox, D. Hennig, and T. Schmidt. Estimating the absolute position of a mobile robot using position probability grids. In *Proc. of the National Conference on Artificial Intelligence (AAAI)*, 1996.
7. J. Carpenter, P. Clifford, and P. Fernhead. An improved particle filter for non-linear problems. Technical report, Department of Statistics, University of Oxford, 1997.
8. I.J. Cox and G.T. Wilfong, editors. *Autonomous Robot Vehicles*. Springer Verlag, 1990.
9. F. Dellaert, W. Burgard, D. Fox, and S. Thrun. Using the condensation algorithm for robust, vision-based mobile robot localization. In *Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 1999.
10. F. Dellaert, D. Fox, W. Burgard, and S. Thrun. Monte carlo localization for mobile robots. In *Proc. of the IEEE International Conference on Robotics & Automation (ICRA)*, 1999.

11. D. Fox. *Markov Localization: A Probabilistic Framework for Mobile Robot Localization and Naviagation*. PhD thesis, Dept. of Computer Science, University of Bonn, Germany, December 1998.
12. D. Fox, W. Burgard, F. Dellaert, and S. Thrun. Monte carlo localization: Efficient position estimation for mobile robots. In *Proc. of the National Conference on Artificial Intelligence (AAAI)*, 1999.
13. D. Fox, W. Burgard, H. Kruppa, and S. Thrun. A monte carlo algorithm for multi-robot localization. Technical Report CMS-CS-99-120, Carnegie Mellon University, 1999.
14. D. Fox, W. Burgard, and S. Thrun. Active markov localization for mobile robots. *Robotics and Autonomous Systems*, 25:195–207, 1998.
15. J.-S. Gutmann and C. Schlegel. AMOS: Comparison of scan matching approaches for self-localization in indoor environments. In *Proc. of the 1st Euromicro Workshop on Advanced Mobile Robots*. IEEE Computer Society Press, 1996.
16. L.P. Kaelbling, A.R. Cassandra, and J.A. Kurien. Acting under uncertainty: Discrete bayesian models for mobile-robot navigation. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 1996.
17. D. Koller and R. Fratkina. Using learning for approximation in stochastic processes. In *Proc. of the International Conference on Machine Learning (ICML)*, 1998.
18. K. Konolige. Markov localization using correlation. In *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI)*, 1999.
19. R. Kurazume and N. Shigemi. Cooperative positioning with multiple robots. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 1994.
20. F. Lu and E. Milios. Globally consistent range scan alignment for environment mapping. *Autonomous Robots*, 4:333–349, 1997.
21. P.S. Maybeck. The Kalman filter: An introduction to concepts. In Cox and Wilfong [8].
22. A.W. Moore, J. Schneider, and K. Deng. Efficient locally weighted polynomial regression predictions. In *Proc. of the International Conference on Machine Learning (ICML)*, 1997.
23. I. Nourbakhsh, R. Powers, and S. Birchfield. DERVISH an office-navigating robot. *AI Magazine*, 16(2), Summer 1995.
24. I.M. Rekleitis, G. Dudek, and E. Milios. Multi-robot exploration of an unknown environment, efficiently reducing the odometry error. In *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI)*, 1997.
25. D.B. Rubin. Using the SIR algorithm to simulate posterior distributions. In M.H. Bernardo, K.M. an DeGroot, D.V. Lindley, and A.F.M. Smith, editors, *Bayesian Statistics 3*. Oxford University Press, Oxford, UK, 1988.
26. B. Schiele and J.L. Crowley. A comparison of position estimation techniques using occupancy grids. In *Proc. of the IEEE International Conference on Robotics & Automation (ICRA)*, 1994.
27. R. Simmons and S. Koenig. Probabilistic robot navigation in partially observable environments. In *Proc. of the International Joint Conference on Artificial Intelligence*, 1995.
28. R. Smith, M. Self, and P. Cheeseman. Estimating uncertain spatial relationships in robotics. In I. Cox and G. Wilfong, editors, *Autonomous Robot Vehicles*. Springer Verlag, 1990.
29. M.A. Tanner. *Tools for Statistical Inference*. Springer Verlag, New York, 1993. 2nd edition.
30. S. Thrun, M. Bennewitz, W. Burgard, A.B. Cremers, F. Dellaert, D. Fox, D. Hähnel, C. Rosenberg, N. Roy, J. Schulte, and D. Schulz. MINERVA: A second generation mobile tour-guide robot. In *Proc. of the IEEE International Conference on Robotics & Automation (ICRA)*, 1999.
31. S. Thrun. Learning metric-topological maps for indoor mobile robot navigation. *Artificial Intelligence*, 99(1):27–71, 1998.