Improved Rao-Blackwellized Mapping by Adaptive Sampling and Active Loop-Closure

Cyrill Stachniss[†] Giorgio Grisetti[†‡] Dirk Hähnel[†] Wolfram Burgard[†]

University of Freiburg[†]
Department of Computer Science
D-79110 Freiburg, Germany

Università 'La Sapienza'[‡]
Dipartimento di Informatica e Sistemistica
I-00198 Rome, Italy

Abstract

Recently Rao-Blackwellized particle filters have been introduced as an effective means to solve the simultaneous localization and mapping (SLAM) problem. In such a particle filter each particle carries an individual map of the environment. Accordingly, a key question is how to reduce the number of particles. Additionally, the approach leaves open how to move the robot in order to improve the accuracy of the learned maps. In this paper we present novel solutions to both problems. First we present an efficient way to compute improved proposal distributions in the prediction step which drastically reduces the uncertainty about the robot's pose. Furthermore, we present an approach to effectively reduce the number of re-sampling steps which seriously reduces the particle depletion problem. Finally, we describe a technique allowing the robot to actively close loops during exploration. By re-entering already visited areas our algorithm reduces the localization error and this way produces more accurate maps. Experimental results carried out with mobile robots in large-scale indoor and outdoor environments illustrate the advantages of our methods over previous approaches.

1 Introduction

Simultaneous mapping and localization belongs to one of the fundamental problems of mobile robotics. Robots that are able to concurrently maintain a model of their environment and to localize themselves relatively to this model are regarded as fulfilling a major precondition of truly autonomous mobile vehicles. Recently, Rao-Blackwellized particle filters have been introduced as an effective means for solving the SLAM problem with occupancy grid maps [4]. The key idea of this approach is to use a particle filter in which each particle carries its own map which is computed based on the trajectory of that particle. The major disadvantage of this approach lies in the huge memory requirements, since one map has to be maintained for each particle. Therefore, effective ways to reduce the number of required particles are of utmost importance when environments which large loops have to be mapped.

As many other approaches to solve the SLAM problem, Rao-Blackwellized particle filters do not belong to the integrated techniques since they only estimate the map of the environment and the location of the vehicle and lack a method to actively control the motions of the vehicle. Controlling

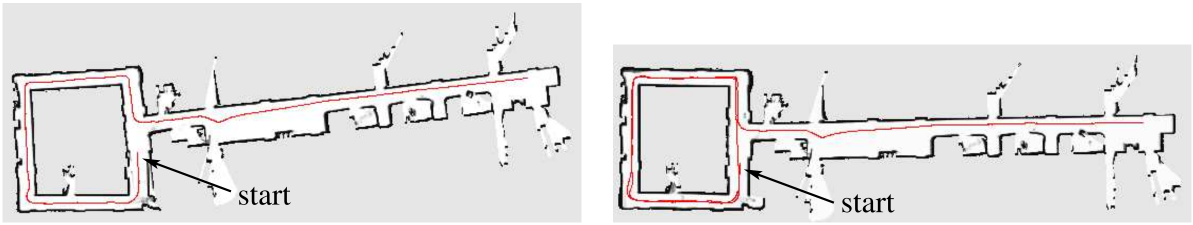


Figure 1. Two different maps obtained from a real world experiment performed in Sieg Hall at the University of Washington. In the left image the robot traversed the loop on the left side only once before it entered the corridor. Accordingly, it was unable to localize itself correctly when it closed the loop. In the right image, the robot traversed the loop twice and therefore could correctly estimate its position in the map. The accuracy of the resulting map therefore is much higher.

the movements of the robot, however, can have a serious influence on the quality of the resulting map. For example, a robot that enters already visited terrain can more accurately localize itself relative to its map compared to an approach which separates the SLAM problem from the control problem and uses an exploration strategy that seeks to visit unknown terrain as fast as possible In general, if the robot uses an exploration strategy that avoids re-visiting known areas of the environment, the probability of making correct associations is reduced.

Figure 1 gives an example that illustrates why an integrated approach that performs active place revisiting provides better results than approaches that do not consider re-entering known terrain during the exploration phase. In the situation shown in the left image the robot traversed the loop just once. The robot was not able to correctly determine the angle between the loop and the straight corridor, because it did not collect enough data to accurately localize itself. The second map shown in the right image has been obtained with the approach described in this paper after the robot traveled twice around the loop before entering the corridor. As can be seen from the figure, this reduces the orientation error from approximately 7 degrees (left image) to 1 degree (right image). This example illustrates that the capability to actively close loops during exploration allows the robot to reduce its pose uncertainty during exploration and thus results in more accurate maps.

In this paper we present a highly efficient variant of the standard Rao-Blackwellized mapping technique which seriously reduces the number of particles needed. This is achieved by using an adaptive motion model which applies a scan-matching procedure during the prediction phase and which estimates the parameters of the motion model based on the uncertainty in the scan-matching process. Additionally we introduce an adaptive resampling scheme that maintains the particle diversity and this way reduces the particle depletion problem. Finally, we present an integrated algorithm that generates trajectories which actively close loops. Our algorithm explicitely takes into account the uncertainty about the pose of the robot during the exploration task. Additionally it avoids that the robot becomes overly confident during the loop closing process. We present practical experiments carried out in large-scale in- and outdoor environments, which can be mapped by our system with 50 particles or even less. Additionally, we present results illustrating that our active loop-closing algorithm yields more accurate maps than a combination of Rao-Blackwellized mapping with a standard exploration behavior that forces the robot to always visit unknown areas.

This paper is organized as follows. After the discussion of related work in the following section, we briefly explain the idea of Rao-Blackwellized mapping in Section 3. Section 4 presents our adaptive techniques to improve the performance this mapping framework. In Section 5 we present our integrated algorithm that actively controls the robot during mapping. Section 6 then presents experiments carried out on real robots as well as in simulation.

2 Related Work

In the past, several techniques have been proposed to reduce the number of necessary particles in the particle filter. One solution is to choose the optimal proposal distribution [3, 19]. Unfortunately, such a distribution is generally unavailable in a form suitable for sampling. However, in several domains it is possible to use Gaussian approximations of the optimal proposal distribution. For example, the unscented particle filter [19] attempts to estimate a Gaussian approximation of the proposal given the model of the system. In FastSLAM-2 [13] the proposal distribution is also approximated by a Gaussian within a Rao-Blackwellized particle filter for landmark-based mapping. In this paper we follow a similar idea in the context of Rao-Blackwellized mapping with grid maps. In particular we utilize a scan-matching procedure for computing a Gaussian approximation of the observation model. This extends our previous work [9] since the proposal distribution is computed on a per-particle basis and dependent on the current observation and the individual maps.

Within the context of exploring unknown environments several previous publications are relevant. Most exploration techniques presented so far focus on generating motion commands that minimize the time needed to cover the whole terrain [1, 10, 20, 21]. Other methods seek to optimize the view-points of the robot to maximize the expected information gain and to minimize the uncertainty of the robot about grid cells [7, 17]. The majority of approaches, however, assumes that the location of the robot is known during exploration. In the area of SLAM, most of the papers focus on the aspect of state estimation as well as belief representation and update [2, 4, 5, 8, 9, 13, 14, 15, 18]. These techniques are passive and do not include means to actively control the motions of the robots. Recently, several integrated approaches have been proposed. For example, Makarenko et al. [12] extract landmarks out of laser range scans and use an Extended Kalman Filter to solve the SLAM problem. They furthermore introduce a utility function which trades-off the cost of reaching frontiers with the utility of selected positions with respect to a potential reduction of the pose uncertainty. The approach is similar to the work done by Feder et al. [6] who consider local decisions to improve the pose estimate during mapping. Both techniques, however, rely on the fact that the environment contains landmarks that can be uniquely determined during mapping. In contrast to this, the approach presented in this paper makes no assumptions about distinguishable landmarks in the environment. It uses raw laser range scans to compute accurate grid maps. It considers the utility of re-entering known parts of the environment and following an encountered loop to reduce the uncertainty of the robot in its pose. This way, the resulting maps become highly accurate.

3 Rao-Blackwellized Mapping

To estimate the map of the environment we use an efficient implementation of the Rao-Blackwellized particle filter for simultaneous localization and mapping proposed by Murphy et al. [4]. The key idea of this approach is to estimate a posterior $p(x_{1:t} \mid z_{1:t}, u_{1:t})$ about potential trajectories $x_{1:t}$ of the robot given its observations $z_{1:t}$ and its odometry measurements $u_{1:t}$ and to use this posterior to compute a posterior over maps and trajectories:

$$p(m, x_{1:t} \mid z_{1:t}, u_{1:t}) \quad = \quad p(m \mid x_{1:t}, z_{1:t}) p(x_{1:t} \mid z_{1:t}, u_{1:t}). \qquad (1)$$

This can be done efficiently, since the quantity $p(m \mid x_{1:t}, z_{1:t})$ can be computed analytically once $x_{1:t}$ and $z_{1:t}$ are known. To estimate the posterior $p(x_{1:t} \mid z_{1:t}, u_{1:t})$ over the potential trajectories Rao-Blackwellized mapping uses a particle filter in which an individual map is associated to each sample. Each map is constructed given the observations $z_{1:t}$ and the trajectory $x_{1:t}$ represented by the corresponding particle.

The particle filter algorithm consists of three major steps. The first step computes the successor state distribution by sampling from a so called proposal distribution π. The second step assigns an individual importance weight to each particle. These weights account for the fact that the proposal distribution π in general is not equal to the true proposal distribution. The third step is the resampling step in which each particle survives with a probability proportional to its importance weight. This resampling process is necessary since only a finite number of particles is used to approximate a posterior and since it allows to apply a particle filter in situations in which the proposal distribution differs from the true one.

In Rao-Blackwellized particle filters for SLAM each particle represents a possible trajectory of the robot. Drawing particles from a proposal distribution is necessary since the exact motion of the robot cannot be determined. Therefore the proposal distribution is often called motion model. The weight $\omega_t^{(i)}$ of particle i being at position $x_t^{(i)}$ is computed according to importance sampling principle:

$$\omega^{(i)} = \frac{p(x_t^{(i)} \mid z_{1:t}, u_{1:t}, x_0)}{\pi(x_t^{(i)} \mid z_{1:t}, u_{1:t}, x_0)}. \tag{2}$$

The effectiveness of a particle filter can be measured in terms of the number of particles required for correctly representing the estimated posterior. Key questions, that have to be solved in practical applications, are how the proposal distribution is computed, when the resampling should be carried out, and how the robot should move through the environment to aquire the necessary data.

Throughout the remainder of this paper, we describe techniques to compute accurate proposal distribution, to adaptively determine when to resample, and how to actively close loops during exploration. All approaches in common lead to a highly efficient Rao-Blackwellized mapping technique that scales to environments that are by one order of magnitude larger than those that could be mapped with previous grid-based variants of Rao-Blackwellized mapping.

4 Adaptive Sampling for Improved Rao-Blackwellized Mapping

In this section we describe two enhancements to Rao-Blackwellized mapping. First we introduce an improved motion model, which allows to draw samples in a highly accurate way. We then present an adaptive resampling strategy, which performs a resampling step based on an estimate about how well the current sample set represents the posterior. As a consequence, the risk of particle depletion is reduced.

As explained above, during the execution of the particle filter one needs to draw samples from a proposal distribution π. This distribution has to be defined by the user and should be as close as possible to the true distribution (see Eq. (1)). Unfortunately, a closed form of this posterior is not known in general. In the context of mobile robot localization, the samples therefore are usually drawn from the motion model $p(x_t \mid x_{t-1}, u_t)$ of the robot. The appropriateness of this approach also depends on the likelihood function $p(z_t \mid x_t)$, which specifies the perception model and is used to compute the importance weights. If the variance of the motion model is significantly larger than the variance of the observation model $p(z_t \mid x_t)$ there is a high risk that the drawn samples represent the posterior in a poor way. Such a situation can typically be observed when laser range finders are used for mobile robot localization. Due to the high accuracy of this sensor the corresponding likelihood function is extremely peaked compared to the model of the odometry.

The current state-of-the-art Rao-Blackwellized mapping algorithms draw particles from a fixed motion model. In order to ensure the convergence of the filter a fixed motion model needs to overestimate the error of the robot's motion. Even if consecutive range measurements are transformed into highly accurate odometry measurements using a scan-matching approach like in our previous work [9] the corresponding motion model has to take into account the worst case scenario. In principle, if all

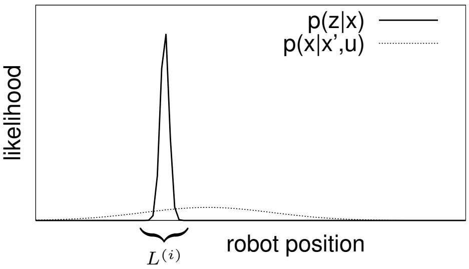


Figure 2. The two components of the motion model. Within in interval $L^{(i)}$ the product of both functions is dominated by the observation likelihood. Accordingly the model of the odometry error can safely be approximated by a constant value.

measurements are maximum range readings this is equivalent to the raw odometry. Conservative approximations typically cause the particle filter to require more samples since the diversity of the particles often is higher than needed. The model introduced in this section is adaptive and adjusts the variance of the proposal according to the accuracy of the scan-matching process. Accordingly, it requires less samples than a conservative approach.

A Rao-Blackwellized mapping technique can be significantly improved by drawing from the optimal proposal distribution (see Doucet [3]):

$$p(x_t^{(i)} \mid m_{t-1}^{(i)}, x_{t-1}^{(i)}, z_t, u_t) = \frac{p(z_t \mid m_{t-1}^{(i)}, x_t^{(i)}) p(x_t^{(i)} \mid x_{t-1}^{(i)}, u_t)}{\int p(z_t \mid m_{t-1}^{(i)}, x') p(x' \mid x_{t-1}^{(i)}, u_t) dx'}. \tag{3}$$

When using a laser range finder, the likelihood function $p(z_t \mid m_{t-1}^{(i)}, x_t^{(i)})$ is usually extremely peaked and dominates the product $p(z_t \mid m_{t-1}^{(i)}, x_t^{(i)}) p(x_t^{(i)} \mid x_{t-1}^{(i)}, u_t)$ within the meaningful area of the likelihood function. Therefore it is possible to approximate $p(x_t^{(i)} \mid x_{t-1}^{(i)}, u_t)$ by a constant k within the interval $L^{(i)}$ given by:

$$L^{(i)} \quad = \quad \left\{ x \mid p(z_t \mid m_{t-1}^{(i)}, x) > \epsilon \right\} \tag{4}$$

Figure 2 illustrates a motion model and the dominant likelihood function as well as the interval $L^{(i)}$. Under this assumption it is possible to formulate an approximation for the right hand side of Eq. (3):

$$\frac{p(z_t \mid m_{t-1}^{(i)}, x_t^{(i)}) p(x_t^{(i)} \mid x_{t-1}^{(i)}, u_t)}{\int p(z_t \mid m_{t-1}^{(i)}, x') p(x' \mid x_{t-1}^{(i)}, u_t) dx'} \quad \simeq \quad \frac{p(z_t \mid m_{t-1}^{(i)}, x_t^{(i)})}{\int_{x' \in L^{(i)}} p(z_t \mid m_{t-1}^{(i)}, x') dx'} \tag{5}$$

In our current system we locally approximate the distribution around the maximum of the likelihood function using a Gaussian so that we obtain

$$p(x_t^{(i)} \mid m_{t-1}^{(i)}, x_{t-1}^{(i)}, z_t, u_t) \quad \simeq \quad \mathcal{N}(\mu_t^{(i)}, \Sigma_t^{(i)}). \tag{6}$$

With such an approximation we obtain a closed form, which is suitable for sampling. The parameters $\mu_t^{(i)}$ and $\Sigma_t^{(i)}$ can be computed by evaluating the likelihood function for a set of points $\{x_j\}$ sampled around the optimal pose obtained from the scan-matching process for particle i

$$\mu_t^{(i)} \quad = \quad \frac{1}{\eta} \cdot \sum_{j=1}^{K} x_j p(z_t \mid m_{t-1}^{(i)}, x_j) \tag{7}$$

$$\Sigma_t^{(i)} \quad = \quad \frac{1}{\eta} \cdot \sum_{j=1}^{K} (x_j - \mu_t^{(i)})(x_j - \mu_t^{(i)})^T p(x_t^{(i)} \mid m_{t-1}^{(i)}, x_j), \tag{8}$$

where $\eta = \sum_{j=1}^{K} p(z_t \mid m_{t-1}^{(i)}, x_j)$ is a normalizer. Note that the scan-matching process as well as the computation of $\mu_t^{(i)}$ and $\Sigma_t^{(i)}$ needs to be carried out for each particle.

Furthermore, we have to specify how the importance weights are computed under this proposal distribution. We can approximate the importance weight $\omega^{(i)}$ for each particle i by:

$$\begin{aligned} \omega_t^{(i)} &= \omega_{t-1}^{(i)} p(z_t \mid m_{t-1}^{(i)}, x_{t-1}^{(i)}) \\ &= \omega_{t-1}^{(i)} \int p(z_t \mid m_{t-1}^{(i)}, x') p(x' \mid x_{t-1}^{(i)}, u_t) dx' \\ &\simeq \omega_{t-1}^{(i)} k \int_{x' \in L^{(i)}} p(z_t \mid m_{t-1}^{(i)}, x') dx' \\ &\simeq \omega_{t-1}^{(i)} k \sum_j p(z_t \mid m_{t-1}^{(i)}, x_j) \\ &= \omega_{t-1}^{(i)} k \eta \end{aligned} \tag{9}$$

The overall idea can be summarized as follows. We use a Gaussian to approximate the optimal proposal distribution based one a scan-matching process which is carried out once per particle. The parameters of this Gaussian are obtained by the maximum likelihood position provided by the scan matcher and by sampling around that position. Given this Gaussian we compute the importance weights based on Eq. (9).

A further aspect that has a major influence on the performance of a particle filter is the resampling step. During resampling particles with a low importance weight $\omega^{(i)}$ are typically replaced by samples with a high weight. On the one hand, this technique is necessary since only a finite number of particles are used to approximate a posterior. Resampling furthermore allows to apply a particle filter in situations in which the true posterior differs from the proposal distribution. On the other hand, the resampling step can delete good samples from the sample set so that the quality of the approximation decreases or, in the worst case, the filter diverges.

Accordingly, it is important to find a criterion when to perform a resampling step. Liu [11] introduced the so-called number of particles N_{eff} to estimate how well the current particle set represents the true posterior. This quantity is computed as

$$N_{\mathit{eff}} = \frac{1}{\sum_{i=1}^{N} \left(w^{(i)}\right)^2}. \tag{10}$$

The intuition behind N_{eff} is as follows. If the samples were drawn from the true posterior the importance weights of the samples would be equal to each other. The worth the approximation the higher the variance of the importance weights. Since N_{eff} can be regarded as a measure of the dispersion of the importance weights, it is a useful measure to evaluate how well the particle set approximates the true posterior. Our approach follows the one proposed by Doucet [3] to determine whether or not a resampling should be carried out. We resample each time N_{eff} drops below a given threshold which was set to $\frac{N}{2}$ where N is the number of particles. In our experiments we found that this technique drastically reduces the risk of replacing good particles, because the number of resampling operations is reduced and resampling operations are only performed when needed.

5 Exploration With Active Loop-Closing for Rao-Blackwellized Mapping

During Rao-Blackwellized mapping, whenever the robot explores new terrain, all samples have more or less the same importance weight, since the most recent measurement is typically consistent with the

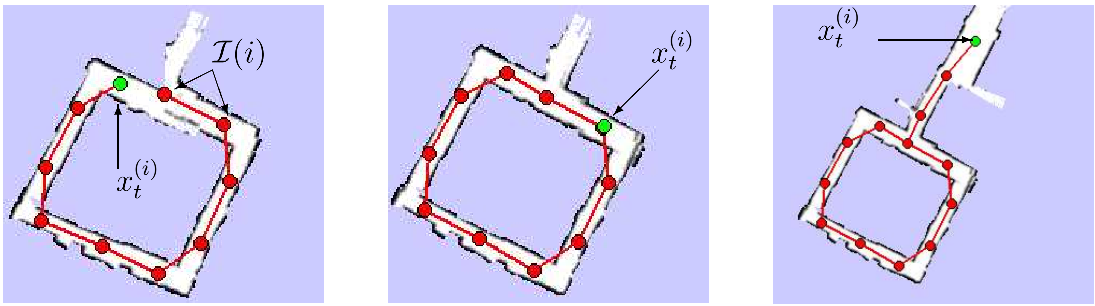


Figure 3. The (red) dots and lines in these three image represent the nodes and edges of $\mathcal{G}^{(i)}$. In the left image $\mathcal{I}(i)$ contains two nodes. In the middle image the robot is closing a loop. Afterwards the robot continues exploring unknown terrain (right image).

part of the map constructed from the immediately preceding observations. As a result, the uncertainty of the particle filter increases. As soon as it re-enters known terrain, however, the maps of some particles are consistent with the current measurement and some are not. Accordingly the weights of the samples differ largely. Due to the resampling step the uncertainty about the pose of the robot usually decreases.

Note that this effect is much smaller if the robot just moves backward a few meters to re-visit previously scanned areas. This is because each map associated to a particle is generally locally consistent. Inconsistencies mostly arise when the robot re-enters areas explored some time ago. Therefore, visiting places seen further back in the history has a stronger effect on the differences between the importance weights and typically also on the reduction of uncertainty compared to places recently observed.

The key idea of our approach is to identify opportunities for closing loops during terrain acquisition. Here closing a loop means actively re-entering the known terrain and following a previously traversed path. To determine whether there exists a possibility to close a loop we consider two different representations of the environment. In our current system we associate to each particle i an occupancy grid map $m^{(i)}$ and a topological map $\mathcal{G}^{(i)}$, which both are updated based on the robot's perceptions and actions while it is performing the exploration task. In the topological map $\mathcal{G}^{(i)}$ the vertices represent positions visited by the robot. The edges represent the trajectory corresponding to the particle i. To construct the topological map we initialize it with one node corresponding to the starting location of the robot. Let $x_t^{(i)}$ be the pose of particle i at the current time step t. We add a new node at $x_t^{(i)}$ to $\mathcal{G}^{(i)}$ if the distance between $x_t^{(i)}$ and all other nodes in $\mathcal{G}^{(i)}$ exceeds a threshold of $c = 2.5m$ or if none of the other nodes in $\mathcal{G}^{(i)}$ is visible from $x_t^{(i)}$

$$\forall n \in \mathit{nodes}(\mathcal{G}^{(i)}) \quad : \quad \left[\mathit{dist}_{m^{(i)}}(x_t^{(i)}, n) > c \quad \vee \mathit{not\_visible}_{m^{(i)}}(x_t^{(i)}, n) \right]. \tag{11}$$

Whenever a new node is added, we also add an edge from this node to the most recently visited node. To determine whether or not a node is visible from another node we perform a ray-casting operation in the occupancy grid $m^{(i)}$.

Figure 3 depicts such a graph for one particular particle during different phases of an exploration task. In each image, the topological map $\mathcal{G}^{(i)}$ is printed on top of metric map $m^{(i)}$. To motivate the idea of our approach we would like to refer the reader to the left image of this figure. Here the robot was almost closing a loop. This can be detected by the fact that the length of the shortest path between the current pose of the robot and previously visited locations in the topological map $\mathcal{G}^{(i)}$ was large whereas it was small in the grid-map $m^{(i)}$.

Thus, to determine whether or not a loop can be closed we compute for each sample i the set $\mathcal{I}(i)$ of positions of interest, which contains all nodes that are close to current pose $x_t^{(i)}$ of particle i based on the grid map $m^{(i)}$ but are far away given the topological map $\mathcal{G}^{(i)}$ of particle i:

$$\mathcal{I}(i) \quad = \{x_{t'}^{(i)} \in nodes(\mathcal{G}^{(i)}) \mid dist_{m^{(i)}}(x_{t'}^{(i)}, x_t^{(i)}) < c_1 \wedge \ dist_{\mathcal{G}^{(i)}}(x_{t'}^{(i)}, x_t^{(i)}) > c_2\} \tag{12}$$

Here $dist_{\mathcal{M}}(x_1, x_2)$ is the length of the shortest path from x_1 to x_2 given the representation $\mathcal{M}$. The distance between two nodes in $\mathcal{G}^{(i)}$ is given by the length of the shortest path between both nodes. The terms c_1 and c_2 are constants that must satisfy the constraint $c_1 < c_2$. In our current implementation the values of these constants are $c_1 = 6m$ and $c_2 = 20m$.

If $\mathcal{I}(i) \neq \emptyset$ there exist so-called shortcuts from $x_t^{(i)}$ to the positions in $\mathcal{I}(i)$. These shortcuts represent edges that would close a loop in the topological map $\mathcal{G}^{(i)}$. The left image of Figure 3 illustrates a situation in which a robot encounters the opportunity to close a loop since $\mathcal{I}(i)$ contains two nodes. The key idea of our approach is to use such shortcuts whenever the uncertainty of the robot in its pose becomes too large. The robot then re-visits portions of the previously explored area and this way reduces the uncertainty in its position.

To determine the most likely movement allowing the robot to follow a previous path of a loop, one in principle has to integrate over all particles and consider all potential outcomes of that particular action. Since this would be too time consuming for online-processing we consider only the particle i^* with the highest accumulated importance weight:

$$i^* \quad = \quad \operatorname*{argmax}_i \sum_{t=1}^{T} \omega_t^{(i)}. \tag{13}$$

Here $\omega_t^{(i)}$ is the weight of sample i at time step t. If $\mathcal{I}(i^*) \neq \emptyset$ we choose the node x_{t_e} from $\mathcal{I}(i^*)$ which is closest to $x_t^{[i^*]}$:

$$x_{t_e} \quad = \quad \operatorname*{argmin}_{x \in \mathcal{I}(i^*)} dist_{m^{[i^*]}}(x_t^{[i^*]}, x). \tag{14}$$

In the sequel x_{t_e} is denoted as the *entry point* at which the robot has the possibility to close a loop. t_e corresponds to the last time the robot was at the node x_{t_e}.

To determine whether or not the robot should activate the loop-closing behavior our system constantly monitors the uncertainty $\mathcal{H}(t)$ about the robot's pose at the current time step. The necessary condition for starting the loop-closing process is the existence of an entry point x_{t_e} and that $\mathcal{H}(t)$ exceeds a given threshold. Once the loop-closing process has been activated, the robot approaches x_{t_e} and then follows the path taken after arriving previously at x_{t_e}. During this process the uncertainty in the pose of the vehicle typically decreases, because the robot is able to localize itself in the map built so far and unlikely particles vanish.

We furthermore have to define a criterion for deciding when the robot actually has to stop following a loop. A first attempt could be to introduce a threshold and to simply stop the trajectory following behavior as soon as the uncertainty becomes smaller than the given value. This criterion, however, can be problematic especially in the case of nested loops. Suppose the robot encounters the opportunity to close a loop that is nested within an outer and so far unclosed loop. If it eliminates all of its uncertainty by repeatedly traversing the inner loop, particles necessary to close the outer loop may vanish. As a result, the filter diverges and the robot fails to build a correct map. Such a situation is shown in Figure 4. To remedy this so-called particle depletion problem [19] we introduce a constraint on the uncertainty of the filter. Let $\mathcal{H}(t_e)$ denote the uncertainty of the posterior when the robot visited the entry point the last time. The new constraint allows the robot to re-traverse the loop only as long as its current uncertainty $\mathcal{H}(t)$ exceeds $\mathcal{H}(t_e)$. If the constraint is violated the robot resumes its frontier-based exploration process. The idea of this constraint is to avoid the depletion of relevant particles during the loop-closing process.

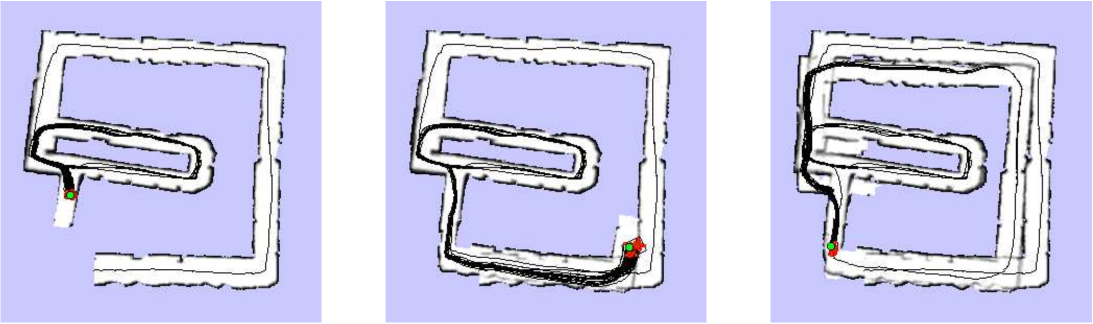


Figure 4. An example for divergence based on particle depletion. A robot traveled through the inner loop several times (left image). After this the diversity of hypotheses about the trajectory outside the inner loop had decreased too much (middle image) and the robot is unable to close the outer loop correctly (right image).

Algorithm 1 The loop-closing algorithm

```
Compute $\mathcal{I}(i^*)$
**if** $\mathcal{I}(i^*) \neq \emptyset$ **then begin**
    $\mathcal{H} \leftarrow \mathcal{H}(t_e)$
    $path \leftarrow x_t^{[i^*]} \cdot shortest\_path_{\mathcal{G}^{[i^*]}}(x_{t_e}, x_t^{[i^*]})$
    **while** $\mathcal{H}(t) > \mathcal{H} \ \wedge \ \mathcal{H}(t) > threshold$ **do**
        $robot\_follow(path)$
**end**
```

To better illustrate the importance of this constraint consider the following example: A robot moves from place A to place B and then repeatedly observes B. While it is mapping B it does not get any further information about A. Since each particle represents a whole trajectory of the robot also hypotheses representing ambiguities about A will vanish when reducing potential uncertainties about B. Our constraint avoids the depletion of particles representing ambiguities about A by aborting the loop-closing behavior at B as soon as the uncertainty drops below the uncertainty stemming from A.

Finally we have to describe how we actually measure the uncertainty in the pose estimate. The most popular way of measuring the uncertainty of a posterior is to calculate its entropy. The entropy, however, has the disadvantage that it does not consider the distance between the individual peaks of multi-modal distributions. In our experiments we figured out that we obtain better results if we use the volume expanded by the samples instead of the entropy of the posterior. We therefore calculate the pose uncertainty by determining the volume of the oriented bounding box around the particle cloud. A good approximation of the minimal oriented bounding box can be obtained efficiently by a principal component analysis.

We use a frontier-based exploration strategy [1] to choose target points for the robot as long as it is localized well enough or no loop can be closed. In our current system we determine frontiers based on the map of the most likely particle i^*. Here a frontier is any known cell that is an immediate neighbor of an unknown, unexplored cell [21].

A precise formulation of the loop-closing strategy is given by Algorithm 1. In our implementation this algorithm runs as a background process, which triggers interrupts of the frontier-based exploration procedure. An application of this algorithm in a simulation run is illustrated in Figure 3.

Note that our loop-closing technique can also handle multiple nested loops. During the loop-closing process the robot follows its previously taken trajectory to re-localize. It does not leave this trajectory until the termination criterion, described in previous section, is fulfilled. Therefore it never starts a

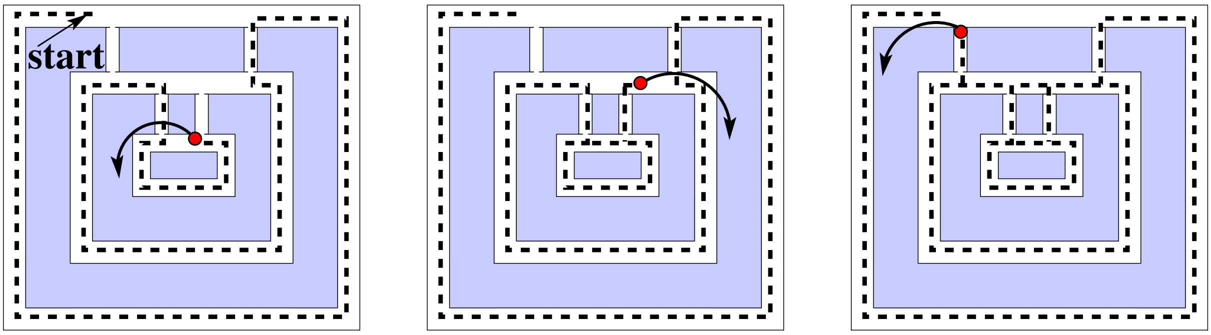


Figure 5. Active loop-closing in an environment with multiple nested loops.

new loop-closing process before the current one is completed. A typical example with multiple nested loops is shown in Figure 5. In the situation depicted in the left image the robot starts with the loop-closing process for the inner loop. After completing this loop it moves to the second inner one and again starts the loop-closing process. Since our algorithm considers the uncertainty at the entry point it keeps enough variance in the filter to close the outer loop. In general, the quality of the solution and whether or not the overall process succeeds depends on the number N of particles used. Since determining N is an open research problem this quantity has to be defined by the user in our current system.

6 Experiments

Our approach has been implemented and evaluated in a series of real world and simulation experiments. For the real world experiments we used an iRobot B21r robot, an ActivMedia Pioneer 2-DX8, and an ActivMedia Pioneer 2-AT outdoor robot. All robots are equipped with a SICK laser range finder. For the simulation experiments we used the real-time simulator of the Carnegie Mellon Robot Navigation Toolkit (CARMEN) [16].

The experiments described in this section are designed to illustrate that our approach can be used to actively learn accurate maps of large environments. It also shows that the improvements of the underlying mapping algorithm have a major influence on quality of the solution. Furthermore, our experiments demonstrate that our integrated approach yields better results than an approach without active loop-closing. Additionally, we analyze how the active termination of the loop-closure influences the result of the mapping process.

6.1 The Improved Motion Model and Adaptive Resampling

To see the enhancements obtained by utilizing our improved motion model and adaptive resampling technique consider Figure 6, which show two maps created from exactly the same data set. This data was acquired using a Pioneer 2-AT robot on our campus at the University of Freiburg. The size of the environment is approximately 250m × 250m and the robot traveled 1.750km. During data acquisition several people walked by, cars passed and also the ground surface was not absolutely flat which makes the mapping task hard. The left image shows a map created with our improved mapper, whereas the right image was constructed with a Rao-Blackwellized mapper lacking the adaptive resampling and the improved motion model but using a scan-matched odometry as input. Both techniques used 30 particles and we used optimized parameters for each approach. As can be seen the sequence of scans in the left map are locally consistent but only the map using the improved techniques is globally

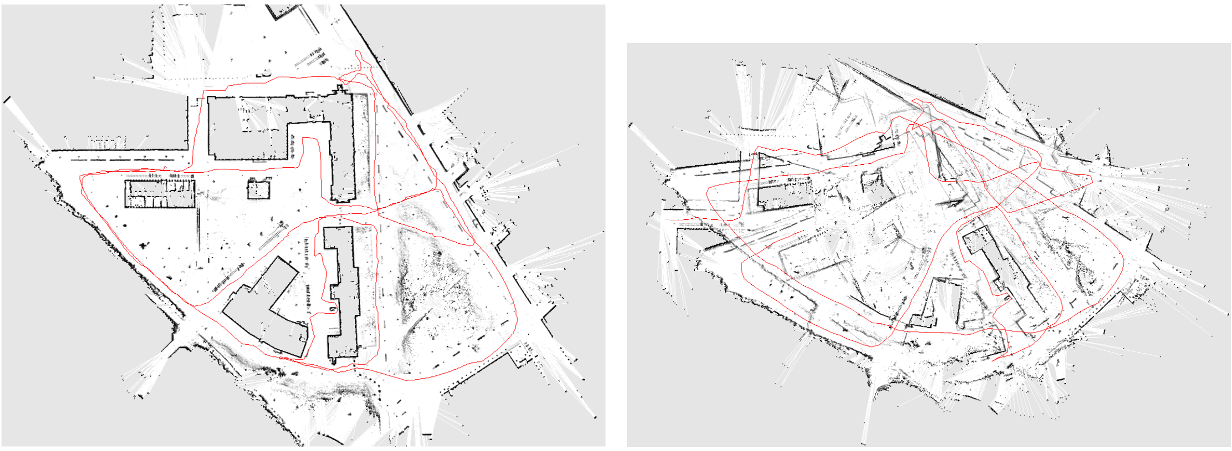


Figure 6. The campus of the Department of Computer Science at the University of Freiburg. The left image shows a map created with the improved Rao-Blackwellized mapping technique, wheres the right image was constructed with a Rao-Blackwellized mapper lacking the improved techniques introduced in this paper.

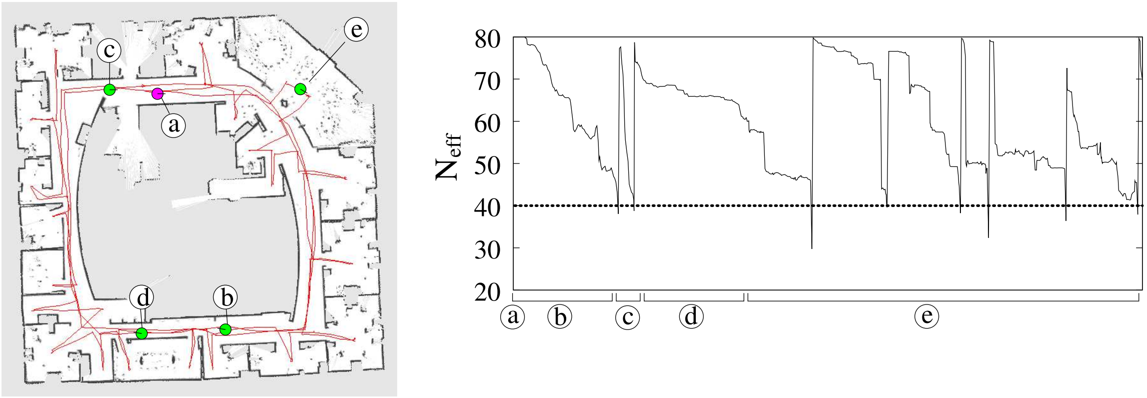


Figure 7. The left image depicts a map of the Intel Research Lab as well as the trajectory of the robot during data acquisition. The robots started at (a), traversed the big loop (b), and closed it (c). It then re-visited the loop again (d) and started exploring the rooms (e). The right plot shows the evolution of N_{eff}. Each time it drops below 40 a resampling is performed.

consistent and at the same time extremely accurate. Note that in general an accurate map could also be build without our improvements, although it would requires seriously more particles.

The next experiment illustrates the evolution of the measure N_{eff} used for our adaptive resampling technique. As mentioned above, we permanently monitor N_{eff} and perform resampling only if this value drops below $\frac{N}{2}$.

Figure 7 plots the value of N_{eff} for a complete mapping task. As can be seen the first resampling is carried out after the first loop-closure, labeled with (c). During the whole experiment only 8 resampling actions were performed, which enabled the filter to keep several hypotheses for a longer period of time. Note the relationship between the value of N_{eff} and whether or not the robot moves through known or unknown terrain. Also note that the value significantly drops when the robots closes a loop (c). This is because the likelihood the the observation varies largely between different particles in such a situation. However, as long as the robot is accurately localized or explores unknown areas this parameter drops only slightly (see (b) and (d)).

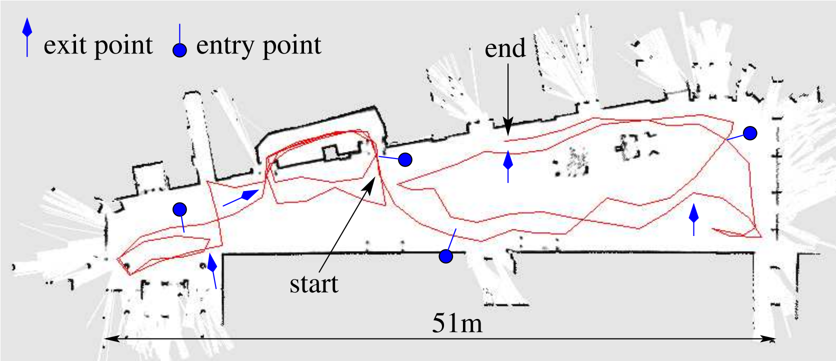


Figure 8. This image shows the resulting map of an exploration experiment carried out using a Pioneer 2 robot equipped with a laser range scanner in the entrance hall of the Department for Computer Science at the University of Freiburg. Also shown is the path of the robot as well as entry and exit points where the robot started and stopped the active loop-closing process.

6.2 Real World Exploration

The first experiment was carried out to illustrate that our current system can effectively control a mobile robot to actively close loops during exploration. To perform this experiment we used a Pioneer 2 robot to explore the main lobby of the Department for Computer Science at the University of Freiburg. The size of this environment is 51m times 18m. Figure 8 depicts the final result obtained by a completely autonomous exploration run using our active loop-closing technique. It also depicts the trajectory of the robot, which has an overall length of 280m. The robot decided four times to re-enter a previously visited loop in order to reduce the uncertainty in its pose. Figure 8 also shows the corresponding entry points as well as the positions where the robot left the loops (“exit points”). As can be seen the resulting map is quite accurate.

6.3 Active Loop-Closing vs. Frontier-Based Exploration

The next experiment was carried out to compare our algorithm with a standard exploration strategy that does not consider loop closing actions. Note that the current implementation of our exploration system does not use the improved mapper and build maps using the standard approach. Nevertheless the problem discussed in the following hold also for the improved technique if reducing the number of particles or increasing the size of the map. Currently we are working on an integration of both system.

The right image of Figure 1 shows the map obtained with a B21r robot in the Sieg Hall at the University of Washington using our algorithm. To eliminate the influence of measurement noise and different movements of the robot we removed the data corresponding to the second loop traversal from the recorded data file and used this data as input to our mapping algorithm. This way we simulated the behavior of a greedy exploration strategy which forces the robot to directly enter the corridor after returning to the starting location in the loop. As can be seen from the left image of Figure 1, an approach that does not actively re-enter the loop fails to correctly estimate the angle between the loop and the corridor which should be oriented horizontally in that figure. Whereas the angular error is 7 degrees with the standard approach it is only 1 degree with our method. Both maps correspond to the particle with the highest accumulated importance factor.

To quantitatively evaluate the advantage of the loop-closing behavior we performed a series of simulation experiments in an environment similar to the Sieg Hall. We performed 20 experiments, 10 with active loop-closing and 10 without. After completing the exploration task we measured the

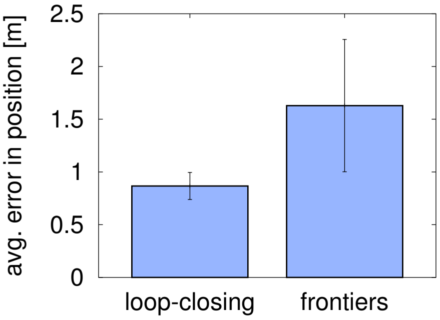


Figure 9. Comparison of our loop-closing strategy with a purely frontier-based exploration technique. The left bar in this graph plots the average error in the pose of the robot obtained with our loop-closing strategy. The right bar shows the average error obtained with a frontier-based approach. As can be seen, our technique significantly reduces the distances between the error in the relative pose estimates.

average error in the relative distances between positions lying on the resulting estimated trajectory and the ground truth provided by the simulator. The results are depicted in Figure 9. As can be seen the active loop-closing behavior significantly reduces the error in the position of the robot.

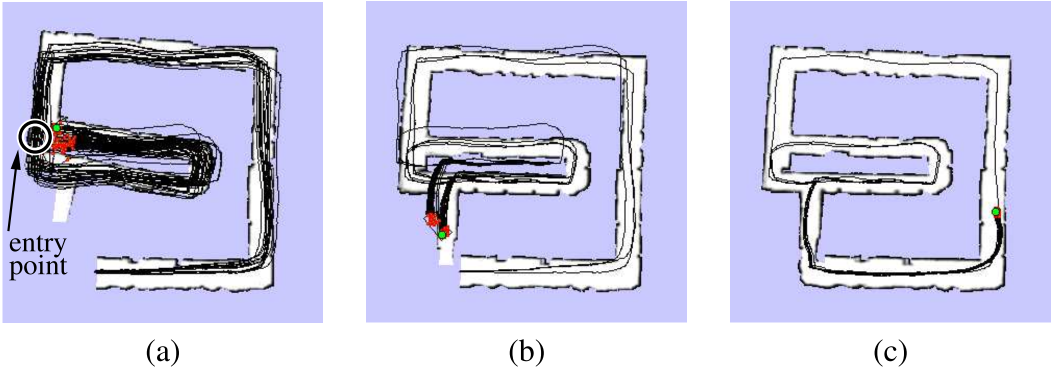
6.4 Importance of the Termination Criterion

In this final experiment we analyze the importance of the constraint that terminates the active loop-closing behavior as soon as the current uncertainty $\mathcal{H}(t)$ of the belief drops under the uncertainty $\mathcal{H}(t_e)$ of the posterior when the robot was at the entry point last time.

In this simulated experiment the robot had to explore an environment containing two nested loops (see Figure 10). In one case we simply used a constant threshold to determine whether or not the loop-closing behavior should be stopped. In the second case we applied the additional constraint that the uncertainty should not become smaller than $\mathcal{H}(t_e)$.

Figure 4 (in Section 5) shows the map of the particle with the highest accumulated importance weight obtained with our algorithm using a constant threshold instead of considering $\mathcal{H}(t_e)$. In this case the robot repeatedly traversed the inner loop (left image) until its uncertainty was reduced below a threshold. After three and a half rounds it decided to again explore unknown terrain, but the diversity of hypotheses had decreased too much (middle image). Accordingly the robot was unable to accurately close the outer loop (right image). We repeated this experiment several times and in no case the robot was able to correctly map the environment. In contrast to that, our approach using the additional constraint always generated an accurate map. One example run is shown in Figure 10. Here the robot stopped the loop-closing after traversing half of the inner loop. In both cases we used 80 particles.

As this experiment illustrates, the termination of the loop-closing is important for the convergence of the filter and to obtain accurate maps in environments with several (nested) loops. Note that similar results in principle can also be obtained without this termination constraint if the number of particles is dramatically increased. Since exploration is an online problem and since every particle carries its own map it is of utmost importance to keep the number of particles as small as possible. Therefore our approach also can be regarded as a contribution to limit the number of particles during Rao-Blackwellized mapping.



(a) (b) (c)

Figure 10. Different situations during the traversal of a nested loop. After exploring unknown areas the robot detected an opportunity to close a loop (a). It then traversed parts of the inner loop until its uncertainty $\mathcal{H}(t)$ did not exceed the uncertainty $\mathcal{H}(t_e)$ of the posterior at the entry point. It then left the loop to explore new terrain (b). After this enough hypotheses were left to correctly close the outer loop (c). In contrast to that, a system considering only a constant threshold fails to correctly close the outer loop (see Figure 4).

7 Conclusion

In this paper we presented several improvements to Rao-Blackwellized particle filtering for simultaneous localization and mapping. First, we presented an approach to compute a highly accurate proposal distribution using scan-matching. Additionally, we dynamically determine whether or not a resampling step needs to be performed. Both approaches in common allow to reduce the number of necessary samples in Rao-Blackwellized mapping and at the same time scale this approach to larger environments. Finally, we introduced a technique to actively close loops. This algorithm forces the robot to re-visit previously traversed loops. Thus, it reduces the uncertainty in the data association and the pose estimate.

Our approach has been implemented and tested on various platforms and in indoor and outdoor environments. Experimental results demonstrate that our current system is able to build large-scale maps with fifty particles or less. Additionally, experiments carried out with our loop-closing algorithm indicate that integrated approaches to exploration and SLAM yield significantly better maps.

Nevertheless, there are several remaining tasks to be solved. One general problem of Rao-Blackwellized mapping is that the number of particles needed to build an accurate map is not known in advance. A further limitation is that there are no means to recover from a divergence of the filter without a complete re-run of the whole algorithm. Finally, it appears interesting to investigate whether the number of effective particles N_{eff} can be used as an indicator to decide whether or not continue the active loop-closing process.

Acknowledgment

This work has partly been supported by the German Science Foundation (DFG) under contract number SFB/TR-8 (A3) and by the Marie Curie program under contract number HPMT-CT-2001-00251.

References

[1] W. Burgard, M. Moors, D. Fox, R. Simmons, and S. Thrun. Collaborative multi-robot exploration. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, 2000.

[2] G. Dissanayake, H. Durrant-Whyte, and T. Bailey. A computationally efficient solution to the simultaneous localisation and map building (SLAM) problem. In *ICRA'2000 Workshop on Mobile Robot Navigation and Mapping*, 2000.

[3] A. Doucet. On sequential simulation-based methods for bayesian filtering. Technical report, Signal Processing Group, Departement of Engeneering, University of Cambridge, 1998.

[4] A. Doucet, J. de Freitas, K. Murphy, and S. Russel. Rao-blackwellized partcile filtering for dynamic bayesian networks. In *Proc. of the Conf. on Uncertainty in Artificial Intelligence (UAI)*, 2000.

[5] A. Eliazar and R. Parr. DP-SLAM: Fast, robust simultainous localization and mapping without predetermined landmarks. In *Proc. of the Int. Conf. on Artificial Intelligence (IJCAI)*, 2003.

[6] H. Feder, J. Leonard, and C. Smith. Adaptive mobile robot navigation and mapping. *International Journal of Robotics Research*, 18(7), 1999.

[7] R. Grabowski, P. Khosla, and H. Choset. Autonomous exploration via regions of interest. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2003.

[8] J.-S. Gutmann and K. Konolige. Incremental mapping of large cyclic environments. In *Proc. of the International Symposium on Computational Intelligence in Robotics and Automation (CIRA)*, 2000.

[9] D. Hähnel, W. Burgard, D. Fox, and S. Thrun. An efficient FastSLAM algorithm for generating maps of large-scale cyclic environments from raw laser range measurements. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2003.

[10] S. Koenig and C. Tovey. Improved analysis of greedy mapping. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2003.

[11] J. Liu. Metropolized independent sampling with comparisons to rejection sampling and importance sampling. *Statist. Comput.*, 6:113–119, 1996.

[12] A. A. Makarenko, S. B. Williams, F. Bourgoult, and F. Durrant-Whyte. An experiment in integrated exploration. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2002.

[13] M. Montemerlo, S. T. D. Koller, and B. Wegbreit. FastSLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges. In *Proc. of the Int. Conf. on Artificial Intelligence (IJCAI)*, 2003.

[14] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. FastSLAM: A factored solution to simultaneous localization and mapping. In *Proc. of the National Conference on Artificial Intelligence (AAAI)*, 2002.

[15] K. Murphy. Bayesian map learning in dynamic environments. In *Neural Info. Proc. Systems (NIPS)*, 1999.

[16] N. Roy, M. Montemerlo, and S. Thrun. Perspectives on standardization in mobile robot programming. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2003.

[17] C. Stachniss and W. Burgard. Exploring unknown environments with mobile robots using coverage maps. In *Proc. of the Int. Conf. on Artificial Intelligence (IJCAI)*, 2003.

[18] S. Thrun. An online mapping algorithm for teams of mobile robots. *International Journal of Robotics Research*, 2001.

[19] R. van der Merwe, N. de Freitas, A. Doucet, and E. Wan. The unscented particle filter. Technical Report CUED/F-INFENG/TR380, Cambridge University Engineering Department, Aug. 2000.

[20] G. Weiß, C. Wetzler, and E. von Puttkamer. Keeping track of position and orientation of moving indoor systems by correlation of range-finder scans. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 1994.

[21] B. Yamauchi. Frontier-based exploration using multiple robots. In *Proceedings of the Second International Conference on Autonomous Agents*, 1998.