Robust Landmark Selection for Mobile Robot Navigation

Maximilian Beinhofer Jörg Müller Andreas Krause Wolfram Burgard

Abstract—Precise navigation is a key capability of autonomous mobile robots and required for many tasks including transportation or docking. To guarantee a robust and accurate localization and navigation performance, many practical approaches rely on observations of artificial landmarks. This raises the question of where to place the landmarks along the desired trajectory of the robot. In this paper, we present a novel approach to landmark selection, which aims at selecting the minimal set of landmarks that bounds the uncertainty about the deviation of the robot from its desired trajectory. At the same time the selected landmark sets are robust against the fact that a certain number of landmarks can be obscured from view during operation. Our algorithm is highly efficient due to a linearization of the whole navigation cycle and employs submodular optimization, for which strong formal bounds on the approximation quality are known. In extensive experiments, also carried out with a real robot, we demonstrate that our approach outperforms several other methods and that it enables robust autonomous robot navigation in practice.

I. INTRODUCTION

For autonomous mobile robots, being able to make distinctive observations of the environment is essential to ensure a reliable navigation performance. Due to possible ambiguities and dynamic changes, the features which are present in the environment are often insufficient for the desired accuracy in navigation. Therefore, especially in industry and logistics, many practical approaches rely on the use of artificial landmarks to achieve a robust navigation performance [8], [20]. Depending on their type, the landmarks or their placement can be expensive. Additionally, the computing power required during navigation increases with the size of the landmark set. On the other hand, landmarks can wear out over time or be obstructed by dynamic objects. Therefore, an ideal landmark placement minimizes their number, while nevertheless allowing for a certain redundancy.

In this paper, we consider the problem of selecting a set of landmarks that is suitable for robust navigation of mobile robots that repeatedly execute the same trajectory. Our approach to landmark selection builds on Bayesian A-optimal design [15], bounding the trace of the covariance of the robot’s deviation throughout the whole trajectory. Thereby, our method effectively bounds the uncertainty about the deviation of the robot from its desired trajectory for all dimensions of the state space. Our approach aims at minimizing the number of landmarks needed, while still satisfying the bound on the trace even if any $k$ of the selected landmarks are not observable. By choosing $k$, the user can trade off the number of landmarks needed against the robustness to missing landmarks.

Our approach has several characteristics that make it especially useful in practice: The robustness against missing landmarks allows the application of smooth fallback procedures: If the robot does not observe a placed landmark for some time, it can send a signal to the maintenance personnel and can still travel safely back to its parking position, leaving the workspace unobstructed for others. For landmark placement, we take into account the desired trajectory of the robot as well as its motion model and its sensor model, making the selected landmarks especially useful for the considered navigation task and the specific robot. As we linearize the model of the entire navigation cycle, our approach is highly efficient and therefore can be utilized even in large scale scenarios. We represent the space of possible landmark locations by a discrete set. Therefore, in contrast to many continuous optimization procedures, our approach does not rely on continuity assumptions for the space of possible landmark locations and is easily adjustable to new scenarios. To achieve the desired robustness in the landmark placement, our approach uses a conservative approximation of the landmark visibility that does only depend on the desired bound specified by the user. Also, we formulate our objective function in a way that allows us to utilize techniques from submodular function optimization, which come with formal approximation guarantees.

II. RELATED WORK

In the context of robot navigation, there exist several approaches to landmark selection for a robot with onboard sensors and to the similar problem of sensor location.
selection for tracking a blind robot. For example, Sala et al. [16] cover an environment with landmarks so that at every position, at least $n \geq 1$ landmarks are observable. Erickson and LaValle [9] derive bounds on the maximum number of deterministically observable color-coded landmarks needed to cover a polygonal region. For a fixed set of possible robot positions, Jouard and Roy [12] place sensors on the walls of buildings to minimize the average position error bound. Lerner et al. [14] select a landmark set that minimizes the weighted trace of the covariance resulting from a single observation of the landmarks. In contrast to these methods, our approach considers the full specification of the robot and its navigation task for landmark selection.

Similar to our approach, Vitus and Tomlin [18] place landmarks along a given trajectory. They use the a-posteriori covariance of the most likely run of the robot to approximate the a-priori covariance. Van den Berg et al. [5] consider a similar linearized system as we do, and choose one out of a set of randomly sampled sensor locations for a single sensor without observability constraints. In our previous work [3], we placed landmarks to minimize the entropy of the joint distribution of all robot states, which is a submodular optimization problem. In that approach, we applied Monte-Carlo simulations for landmark placement, which is computationally more demanding than the linearization we apply in this paper. In a second previous approach [2], we considered an efficient linearized system with a less strict observability constraint than the one presented in this paper. Therefore, that approach is not robust against missing landmarks and is highly non-submodular. Also, both our previous approaches do not consider the case of missing landmarks. See Sec. VI-B for a comparison to these methods.

To our knowledge, there exists no other approach in the landmark selection literature that takes into account the specification of the navigation task and ensures robustness against missing landmarks.

Also on the topic of submodular function optimization, there exists a large body of literature [7], [11], [13], [17], ranging from applications like Gaussian Process model fitting for ship hull inspection [11] to selecting beneficial grasping poses for robotic manipulators [17].

III. BACKGROUND AND PROBLEM STATEMENT

We consider the problem of landmark placement for a mobile robot that repeatedly and autonomously travels along the same trajectory $T$. Hereby, the trajectory $T = (\{x_1^\star, u_1^\star\}, \ldots, \{x_T^\star, u_T^\star\})$ is a time-discrete sequence specifying the desired robot state $x_t^\star$ and control command $u_t^\star$ at each time step $t$.

For localization, the robot has a map of the positions and the unique identities of the landmarks $A = \{L_1, \ldots, L_n\}$ in the area surrounding the trajectory and is equipped with a sensor to observe them. The robot takes noisy observations $z_t^A$ of the landmarks inside its actual field of view according to a sensor model $z_t^A = h(x_t, \text{obs}(x_t, A), u_t)$. Here, $\text{obs}(x_t, A)$ specifies the subset of the landmarks in $A$ which are observable from the state $x_t$ and $u_t \sim \mathcal{N}(0, N_t)$ is the sensor noise, which we assume to be Gaussian distributed.

After taking an observation and updating its state estimate in a localization algorithm, the robot executes a control command $u_t$, propagating its state $x_t$ according to a noisy motion model $x_{t+1} = f(x_t, u_t, m_{t+1})$, where $m_{t+1} \sim \mathcal{N}(0, M_{t+1})$ is the motion noise. The control commands are selected by a linear-quadratic regulator (LQR) feedback controller [6]. At each time step $t$, the LQR controller selects the control command $u_t$ that minimizes the expected quadratic error term

$$E\left[\sum_{t=1}^{T} (\Delta x_t^T C \Delta x_t + \Delta u_t^T D \Delta u_t)\right].$$

where $\Delta x_t = x_t - x_t^\star$, $\Delta u_t = u_t - u_t^\star$, and $C$ and $D$ are positive definite weight matrices.

In this closed-loop system for autonomous navigation, we consider a discrete set of possible landmark locations $\mathcal{V}$, and aim to select the subset $A \subseteq \mathcal{V}$ of landmarks for placement that are most beneficial for the navigation task. We measure the quality of a landmark selection $A$ by considering the a-priori probability distribution of the deviation $x_t - x_t^\star$ of the robot from its desired trajectory

$$p(x_t - x_t^\star \mid \text{obs}(x_{1:t}, A)) = \int \int p(x_t - x_t^\star \mid u_{1:t-1}, z_{1:t}, \text{obs}(x_{1:t}, A)) \cdot p(u_{1:t-1}, z_{1:t} \mid \text{obs}(x_{1:t}, A)) \, du_{1:t-1} \, dz_{1:t}. \quad (2)$$

Note that this distribution is independent of the actual values of the controls $u_{1:t-1}$ and observations $z_{1:t}$ that are not yet known at the moment of landmark placement. The a-priori distribution depends, however, on the number and the positions of the landmarks $\text{obs}(x_{1:t}, A)$ that are observable during operation on the trajectory.

In this work, we consider Bayesian A-optimal design [15], i.e., we aim to select landmarks so that the trace of the estimated covariance matrix $\hat{S}_t^A$ of $p(x_t - x_t^\star \mid \text{obs}(x_{1:t}, A))$ stays below a user-defined threshold for every time step $t \in [1, T]$. Thereby, we effectively bound the uncertainty about the deviation of the robot from its desired trajectory for all dimensions of the state space.

To guarantee safe operation even if up to $k$ landmarks are missing, we aim at finding

$$A^* = \arg\min_{A \subseteq \mathcal{V}} |A| \quad \text{subject to} \quad \max_{B \subseteq A, |B| \leq k} \text{tr} \left( \hat{S}_t^{A \setminus B} \right) \leq \epsilon_t \quad \forall t \in [1, T]. \quad (4)$$

This is the smallest set of landmarks $A^*$ that ensures a bounded trace of the estimator of the a-priori covariance. In particular, we ensure that the trace of the estimator $\hat{S}_t^{A \setminus B}$ of the a-priori covariance stays below $\epsilon_t$ for all $t \in [1, T]$, even if any subset $B \subseteq A^*$ with $|B| \leq k$ is not observable during operation. Here, $\epsilon_t$ is a user-defined bound that can be set for each part of the trajectory individually. This allows
the user to specify lower $\epsilon_t$ values for critical parts of the trajectory in which a path execution with higher accuracy is required.

To efficiently place landmarks in this framework, two important issues have to be addressed: handling the combinatorial structure of the problem stated in Eqs. (3) and (4) and estimating the a-priori covariance $\hat{S}_t^A$. Our efficient solution to the combinatorial optimization problem is presented in the next section. In general, $\hat{S}_t^A$ cannot be estimated in closed form, so one solution that is often applied is to approximate the high-dimensional integral defined in Eq. (2) via Monte-Carlo simulation. Monte-Carlo simulation can deal, for example, with non-linearities due to discontinuities in the observability $\tilde{obs}(x_t, A)$ of landmarks depending on the actual state of the robot, but it is computationally demanding. In contrast to that, using the conservative approximation $\hat{obs}(t, A)$ of the landmark observability defined in the next section, we can estimate $\hat{S}_t^A$ efficiently by applying the linearization proposed by van den Berg et al. [4]. Their approach linearizes the model of the whole navigation system, consisting of observation, localization, control, and motion, resulting in a Gaussian a-priori distribution

$$p \left( x_t - x_t^* \mid \hat{obs}(1:t, A) \right) \sim \mathcal{N}(0, \hat{S}_t^A)$$

that can be calculated efficiently via standard matrix manipulations. This method has been applied successfully in practice [2], [5]. For completeness, it is described in detail in the appendix.

IV. EFFICIENT AND ROBUST LANDMARK SELECTION

We now describe our approach towards landmark placement. We first propose how to address the challenge of uncertain landmark observability. We then reformulate Eq. (4) such that it becomes amenable to efficient optimization.

A. Observability Constraints

For most types of landmark detection sensors (e.g., cameras, RFID readers, and laser range finders), the ability to observe a landmark $L$ changes with the state $x_t$ of the robot due to a limited sensor range or obstacles concealing the landmark. However, in our approach it is necessary to evaluate the quality of $L$ at a stage at which the only knowledge about $x_t$ is the condition on its covariance defined in Eq. (4), which the final landmark set shall guarantee.

Therefore, we define the approximate observability $\hat{obs}(t, A)$ in a conservative way: We consider a landmark as observable at time $t$ only if it is observable with a probability of at least $(1 - \delta)$ according to every a-priori distribution for which the bound in Eq. (4) holds. Fig. 2 shows an example of $\hat{obs}(t, A)$ for a two dimensional robot state $x_t = [x_t, y_t]$. Note that in the experiments, we consider a three dimensional robot state $[x_t, y_t, \theta_t]$.

We calculate $\hat{obs}(t, A)$ using the following insights: As stated in Eq. (5), all estimated a-priori distributions are Gaussians. The $(1 - \delta)$-confidence region of a Gaussian is an ellipsoid whose principal axes have the lengths $c \sqrt{\lambda_{t,i}}$. Here, $\lambda_{t,i}$ are the eigenvalues of the covariance $\hat{S}_t^A$ and

$$c = c(\delta, \dim(x_t))$$

is a constant that depends only on the probability $\delta$ and the dimensionality of the state space. If the bound in Eq. (4) holds, it holds that $tr(\hat{S}_t^A) \leq \epsilon_t$ and therefore $\lambda_{t,i} \leq \epsilon_t$ for all eigenvalues $\lambda_{t,i}$ of $\hat{S}_t$. Consequently, at time $t$, at least $(1 - \delta)$ of the probability mass of every Gaussian that satisfies Eq. (4) lies inside the sphere $K$ with radius $c \sqrt{\lambda_{t,i}}$ and center $x_t^*$. If a landmark $L$ is observable from within every state inside $K$, we define it to be in $\hat{obs}(t, A)$, resulting in

$$\hat{obs}(t, A) = \{ L \mid L \in \tilde{obs}(x_t, A) \forall x_t \in K \} .$$

Having defined $\hat{obs}(t, A)$, we can evaluate $\hat{S}_t^A$. Therefore, we can also evaluate the bound in Eq. (4) for a given set $A$ by iterating over all subsets of $A$ with $|A| - k$ elements. Note that in the definition of $\hat{obs}(t, A)$ we assume that the bound in (4) holds for $A$. Hence, if the bound holds, then also the approximation of the observability, which was applied in the evaluation of the bound, is conservative.

B. Objective Function

Being able of evaluate Eq. (4) for a given landmark set $A$ makes it possible to run a brute force search on the power set $\mathcal{P}(V)$ of all possible landmark positions to find the optimal landmark set satisfying Eq. (4). However, as $\mathcal{P}(V)$ grows exponentially with the number of possible landmark locations $|V|$, we apply an efficient approximation instead. We now show, building on techniques of Krause et al. [13], how the overall problem defined in Eqs. (3) and (4) can be reformulated in a way that admits highly efficient approximation algorithms.

As a first step, we define the reduction of the trace of the a-priori covariance induced by the observations of the landmarks in $A$ as

$$F_t(A) = tr(\hat{S}_t^A) - tr(\hat{S}_t^{A)})$$

for every time step $t$. We truncate this function at the target value $tr(\hat{S}_t^{A}) - \epsilon_t$, leading to the function

$$F_t(\epsilon_t, A) = \min \left( F_t(A), tr(\hat{S}_t^{A}) - \epsilon_t \right) .$$

Fig. 2. Desired robot state $x_t^*$ and landmark $L$. All $(1 - \delta)$ confidence regions of estimated covariances satisfying the bound in Eq. (4) (one example shown in light blue) are inside the circular region with radius $c \sqrt{\lambda_{t,i}}$ (dark blue). According to $\hat{obs}(t, L)$, $L$ is visible at time $t$ if it is inside the sensor range of every pose in the dark blue circle and the shaded area is free of obstacles that would block the robot’s line of sight to the landmark.
Note that this function achieves its maximum value if and only if the target condition $\text{tr}(\hat{S}_t^A) \leq \epsilon_t$ is satisfied. We take into account the robustness against up to $k$ missing landmarks in the objective function by considering the average over $F_t(\epsilon_t, A \setminus B)$ for all possible subsets of missing landmarks $B$: \[
F(k, \epsilon_t, A) = \frac{1}{\sum_{i=0}^{k} (|A|)} \sum_{B \subseteq A, |B| \leq k} F_t(\epsilon_t, A \setminus B). \tag{9}
\]

Similar to the function in Eq. (8), this function achieves its maximum value $\text{tr}(\hat{S}_t^A) - \epsilon_t$ if and only if the target condition under $k$-robustness $\text{tr}(\hat{S}_t^A|B) \leq \epsilon_t$ is satisfied for all $B$. We finally consider multiple time steps or a whole trajectory by using the same averaging procedure, leading to \[
F(k, \epsilon_{1:T}, A) = \frac{1}{T} \sum_{t=1}^{T} F(k, \epsilon_t, A). \tag{10}
\]

Due to its construction, this function takes on its maximum
\[
c = \frac{1}{T} \sum_{t=1}^{T} \text{tr}(\hat{S}_t^A) - \epsilon_t \tag{11}
\]
if and only if the condition defined in Eq. (4) is satisfied. With this, we can re-formulate the problem definition stated in Eqs. (3) and (4) in terms of $F$ as
$$
A^* = \arg\min_{A \subseteq V} \left| A \right| \quad \text{s.t.} \quad F(k, \epsilon_{1:T}, A) = c. \tag{12}
$$

C. Landmark Selection Algorithm

As problems of the type defined in Eq. (12) are typically NP-hard (see Sec. V for details), we apply a greedy iterative landmark selection algorithm that finds an approximate solution to (12). The procedure is stated in Algorithm 1.

Algorithm 1 Iterative approximation algorithm

Input: $V, k, \epsilon_{1:T}$

Output: $A$

$A = \emptyset$

while $F(k, \epsilon_{1:T}, A) < c$ do

$L^* = \arg\max_{L \subseteq V} F(k, \epsilon_{1:T}, A \cup \{L\})$

$A = A \cup \{L^*\}$

end while

return $A$

The computation of the $\arg\max$ operator in the algorithm evaluates each landmark $L$ individually, which makes it well-suited for parallel computing.

Note that due to the usage of $\tilde{\text{obs}}$ in the evaluation of $F$, which is a conservative approximation of the observability only if $F(k, \epsilon_{1:T}, A) = c$, stopping the algorithm before $F(k, \epsilon_{1:T}, A)$ reaches $c$ leads to landmark sets that can perform arbitrarily badly. However, for the final output set $\hat{A}$, the observability, and therefore also the condition in Eq. (4), is approximated conservatively.

D. Practical Considerations

Our algorithm can be used to guarantee a collision-free path execution with high confidence. For that, we use the same insights as for the approximation of the observability. We choose the bound $\epsilon_t$ on the trace such that the nearest static obstacle is at least $c_1 \sqrt{\epsilon_t}$ away from the desired state $x_t^*$ for every $t$. To avoid collisions with moving obstacles without breaking the bound on the trace, the robot needs to stop if its path is blocked and wait until the moving obstacle left the corridor with width $c_2 \sqrt{\epsilon_t}$ around the desired trajectory.

If the robot has to repeatedly execute the same round-trip task, our method can be adjusted to guarantee bounded traces even for a continuous long-term operation of the robot. By setting $\epsilon_T$ to at most the minimum eigenvalue of $\hat{S}_1$, our algorithm produces a landmark set $\hat{A}$ that guarantees that $S_T$ is governed by $\hat{S}_1$, which enables a continuous safe operation.

V. APPROXIMATION BOUND

In this section, we provide a theoretical motivation for our approximation algorithm, which rests on the concept of submodularity, a natural diminishing returns property. For landmark sets, submodularity states that adding a landmark to an already large set of landmarks $C$ results in a smaller increase in the objective function than adding the same landmark to a subset of $C$. Concretely, a function $F$ is called submodular if for all $A \subseteq C \subseteq V$ and all landmarks $L \in V \setminus C$

$$
F(C \cup \{L\}) - F(C) \leq F(A \cup \{L\}) - F(A). \tag{13}
$$

For submodular functions, problems of the type defined in Eq. (12) are called submodular set cover problems, and are NP-hard in general [19]. However, for these problems, Wolsey [19] showed that for greedy solutions $A_{\text{greedy}}$, such as those produced by Algorithm 1, it holds that

$$
|A_{\text{greedy}}| \leq |A^*| \left(1 + \log \max_{L \subseteq V} F(\{L\})\right), \tag{14}
$$

and under natural complexity-theoretic assumptions, no efficient algorithm can provide better solutions. Hence, such greedy solutions $A_{\text{greedy}}$ are near-optimal for submodular set cover problems.

Therefore, the key question is whether (or under which conditions) our objective function $F(k, \epsilon_{1:T}, A)$ for landmark placement is monotonic and submodular. First, note that the function $F_t(A)$ (Eq. (7)), from which our objective function is constructed, can be rearranged to be the sum of the variance reductions in the diagonal elements of the covariance. Das and Kempe [7] show that variance reduction, such as the summands of $F_t(A)$, under certain technical conditions on the covariance matrices is usually monotonic and submodular. In this case, $F(k, \epsilon_{1:T}, A)$ is monotonic and submodular as well, since these properties are preserved under nonnegative linear combinations and truncation [10], [13]. These observations theoretically justify our efficient approximation Algorithm 1. Similar techniques have been successfully used in other domains [11], [13], [17].
VI. Experimental Results

We evaluated our approach in extensive experiments both with a simulated differential drive robot and with a real holonomic drive robot. For these robots, the state $x_k$ of the robot can be described by its pose $[x_k, y_k, \theta_k]$ in the 2D-plane. We assume that the robot is equipped with a landmark detection sensor with a circular field of view and 5 m sensor range. In the different experiments, the sensor can observe either landmarks placed on the walls or landmarks placed on the ceiling of the environment, resulting in different sets $V$ of possible landmark locations. In all experiments, we set the allowed maximum trace $\epsilon_t$ to 0.05 for all $t$ and the probability $\delta$ in the observability constraint to 1%.

A. Evaluation of Robustness

In the first set of simulation experiments, we evaluated the robustness of our landmark sets against missing landmarks. To this end, we considered the two trajectories shown in Fig. 3, corresponding to a sweeping pattern in an obstacle-free environment and a surveillance task in an environment with obstacles. For the sweeping trajectory, our approach selected 10, 15, and 19 landmarks assuming at most zero, one, and two missing landmarks, respectively. For the surveillance task, our approach selected 12, 20, and 27 landmarks. To evaluate the effects of the linear approximation applied in the landmark selection method, we conducted Monte-Carlo simulations using the real, non-linear models. In the simulations, we estimated the a-priori traces $tr_{MC}(S_k)$ using the empirical distributions gained from the deviations $x_k - x'_k$ observed in 1000 simulated executions of the trajectory in each scenario. For all six landmark configurations and all possible combinations of $k$ missing landmarks, the Monte-Carlo simulations resulted in traces that were below the bound $\epsilon_t = 0.05$ for all time steps $t$. The maximum deviation is stated in $m$.

B. Comparison to Other Approaches

In the second set of simulation experiments, we compared the landmark sets selected by our approach to the ones selected by our previous approaches [2], [3], and by two straightforward heuristics. The Deviation method [2] aims at minimizing the translational deviation of the robot from its trajectory in a linearized system. Entropy [3] minimizes the entropy of the joint distribution of $x_{1:T}$ using Monte-Carlo simulations instead of linearized models. The Grid and Random heuristics place a given number of landmarks in the area observable by the robot. Grid in a regular grid pattern and Random at randomly sampled locations. For comparison, we adjusted all methods to select the same numbers of landmarks. To get scenario-independent results, we considered ten randomly chosen trajectories, each connecting two randomly sampled goal points in an area of $15 \times 15 \text{m}$. On these trajectories, our approach selected landmark sets of average sizes 5.7 and 9.3 for $k = 0$ and $k = 1$, respectively. We used Monte-Carlo simulations to evaluate the maximum trace per trajectory, the 99% quantile of the maximum translational deviation per trajectory, and the information gain of the joint distribution of $x_{1:T}$. The results can be seen in Fig. 4. The Entropy approach results in the best information gain values, but yields suboptimal traces and deviations. This is probably due to the fact that minimizing the entropy can lead to flat but elongated covariance ellipses, which have high
trace and deviation values. Paired sample t-tests showed that our approach results in significantly smaller maximum traces and 99% quantiles of the maximum translational deviations than Entropy, Grid, and Random on a 5% level for $k = 0$ as well as $k = 1$. For zero missing landmarks, the Deviation method produces maximum traces and deviations similar to our approach, but seems to behave unreliably in the case of a missing landmark. For $k = 1$, when the most influential landmark is hidden, it results in maximum traces and deviations that are significantly higher than the ones produced by our approach and even the Entropy approach on a 5% level. This is probably due to the fact that the placement algorithm employed in the Deviation approach aims at maximizing the distance between two consecutive landmarks, which is especially unfavorable in the case of missing landmarks, while our approach uses more sophisticated techniques from submodular function optimization to explicitly take missing landmarks into account.

Running multi-threaded on an Intel® Core™ i7 2.8GHz, the runtime of the Entropy approach for computing a single landmark set was 8:10 h on average, while due to the linearization in landmark placement, single-threaded implementations of our approach and Deviation took 28:27 min and 27:22 min, respectively.

C. Landmark Selection for Changing Bounds

To demonstrate the ability of our approach to select landmarks for values of $\epsilon_t$ which vary along the trajectory, we applied our approach on the pick-and-place trajectory shown in Fig. 5. As can be seen in the figure, a higher demand for accuracy in the pick-up and deposit zones and the robustness against a missing landmark in this case lead to the same number of landmarks, but to different locations.

D. Long Term Evaluation on a Real Robot

Finally, we evaluated the landmark sets selected by our approach also on the real robot shown in Fig. 1. The robot is equipped with Mecanum wheels for omnidirectional motion and with two SICK S300 laser scanners mounted on opposite corners of the robot, providing a 360° field of view. The lasers can detect reflective markers, whose unique landmark IDs we calculated using a nearest neighbor heuristic. In a training run, we calibrated the motion noise and sensor noise of this specific robot, and used the calibrated parameters in the linearized models for landmark placement.

VII. CONCLUSIONS

In this paper, we presented a novel method to trajectory-dependent landmark placement, which is robust against missing landmarks. It keeps the trace of the a-priori covariances of all robot states below a user-defined threshold, effectively bounding the uncertainty in all dimensions of the state space. The linearized objective function in our method takes into account the full specification of the navigation task and can be evaluated efficiently. We showed how our problem can be reformulated in a way that enables the use of submodular optimization techniques, which enjoy strong theoretical guarantees. Extensive experiments, also with a real robot, demonstrate that the robustness against missing landmarks resulting from our approach is guaranteed in practice, even in

Fig. 5. Pick-and-place trajectory and landmark sets selected by our approach for a constant $\epsilon_t$ of 0.05 for $k = 0$ (left) and $k = 1$ (right), and for $k = 0$ and $\epsilon_t = 0.03$ for the time steps in which the robot is inside the pick-up and deposit zones (gray areas) and 0.05 outside (middle). The trajectory goes from the start to goal 1, then to goal 2, and back to the start.

Fig. 6. Landmark sets (red triangles), desired robot path (blue) and actual robot paths (red) for $k = 0$ (left), and $k = 1$ (right). For $k = 0$, also the linearized 99% a-priori confidence regions are shown (light blue). For $k = 1$, each landmark was hidden during 10 runs of the robot. The landmark whose absence resulted in the largest deviations is marked in pink. The trajectory goes from the start to goal 1, back to the start, then to goal 2 and again back to the start.
continuous long term operation. The experiments also show that our approach outperforms several other approaches to landmark selection.

**APPENDIX**

**The Linearized System for A-Priori Covariance Estimation**

In our landmark selection algorithm, we apply the efficient method for the estimation of the a-priori covariance $S_t^A$ introduced by van den Berg et al. [4], which is restated here for completeness. This method assumes that the observability of landmarks $\tilde{\text{obs}}(t, A)$ is independent of the concrete state $x_t$ of the robot. It uses first-order Taylor approximations to linearize the sensor model $h(x_t, \tilde{\text{obs}}(t, A), n_t)$ and the motion model $f(x_t, u_t, m_t)$ around the desired state $x_t^*$ and the desired control $u_t^*$. The resulting linearized functions are

$$x_t \approx f(x_{t-1}^*, u_{t-1}^*, 0) + A_t(x_{t-1} - x_{t-1}^*) + B_t(u_{t-1} - u_{t-1}^*) + V_t m_t,$$

$$z_t \approx h(x_t^*, \tilde{\text{obs}}(t, A), 0) + H_t(x_t - x_t^*) + W_t n_t,$$

with the Jacobians

$$A_t = \frac{\partial f}{\partial x}(x_{t-1}^*, u_{t-1}^*, 0), \quad B_t = \frac{\partial f}{\partial u}(x_{t-1}^*, u_{t-1}^*, 0),$$

$$V_t = \frac{\partial f}{\partial m}(x_{t-1}^*, u_{t-1}^*, 0),$$

$$H_t = \frac{\partial h}{\partial x}(x_t^*, \tilde{\text{obs}}(t, A), 0), \quad W_t = \frac{\partial h}{\partial n}(x_t^*, \tilde{\text{obs}}(t, A), 0).$$

In this linearized system, the Kalman filter (KF) is the minimum-mean-square error estimator [1] for the a-posteriori distribution $p(x_t | x_{t-1}, u_{t-1}, z_{1:t}, \tilde{\text{obs}}(1:t, A))$, estimating it as a Gaussian $N(\mu_t | x_t, \hat{\Sigma}_t^A)$. In contrast to the extended Kalman filter (EKF) [1], we linearize around the a-priori known desired states instead of the means in the filter. Therefore, the Kalman gains $K_t$ in the KF can be computed a-priori, without knowledge of the values of $u_{t-1}$ and $z_{1:t}$. Applied on the mean $\mu_t$ in the Kalman filter, the LQR controller selects control commands $u_t$ according to

$$u_t - u_t^* = L_t(\mu_t - x_t^*),$$

where $L_t$ is the feedback matrix that minimizes the quadratic error defined in Eq. (1). With the Jacobians defined in Eq. (17), also the feedback matrices $L_t$ can be calculated a-priori via the recursive formula $E_T = C$,

$$L_t = -(B_{t+1}^T E_{t+1} B_{t+1} + D)^{-1} B_{t+1}^T E_{t+1} A_{t+1},$$

$$E_t = C + A_{t+1}^T E_{t+1} A_{t+1} + A_{t+1}^T E_{t+1} B_{t+1} L_t .$$

As can be seen in Eq. (18), the LQR controller linearly connects the Kalman filter estimate $\mu_t$ with the true robot state $x_t$ via the selected control $u_t$. This fact allowed van den Berg et al. [4] to prove that in the linear navigation system defined above, the joint a-priori distribution of the deviations of $x_t$ and $\mu_t$ from $x_t^*$ is a Gaussian

$$N(x_t^* - x_t^*, \mu_t - x_t^*, \hat{\Sigma}_t^A) \approx \mathcal{N}(\mu_t | x_t, \hat{\Sigma}_t^A),$$

whose covariance $\hat{R}_t^A$ can be computed recursively by

$$\hat{R}_t^A = \begin{bmatrix} P_t^A & 0 \\ 0 & 0 \end{bmatrix}, \quad \hat{R}_t^A = F_t \hat{R}_{t-1}^A F_t^T + G_t \begin{bmatrix} M_t & 0 \\ 0 & N_t \end{bmatrix} G_t^T,$$

with

$$F_t = \begin{bmatrix} A_t & B_t L_{t-1} \\ K_t H_t A_t & A_t + B_t L_{t-1} - K_t H_t A_t \end{bmatrix},$$

$$G_t = \begin{bmatrix} V_t & 0 \\ K_t H_t V_t & K_t W_t \end{bmatrix}. $$

This recursive scheme can be computed independently of the actual values $u_t$ and $z_t$. It efficiently estimates the a-priori covariance $\hat{\Sigma}_t^A$ of the deviation as the upper left block of $\hat{R}_t^A$.

**REFERENCES**


