Robot Localization with Sparse Scan-based Maps

Alexander Schiotka Benjamin Suger Wolfram Burgard

***Abstract*— Occupancy grid maps are a popular method for representing the environment in the context of robot navigation tasks. However, occupancy grid maps can have a high memory demand that grows quadratically with the range of the sensor. In this paper, we introduce a memory-efficient map representation that is based on a constant set of individual scans. To make these scan-based maps suitable for autonomous robot navigation, we propose probabilistically sound methods for both mapping and localization. To solve the mapping problem, our approach incrementally selects scans based on the additional information they provide relative to the scans previously selected. Using these selected scans, we perform an Monte Carlo Localization (MCL) approach with a sensor model optimized for the scan-based representation of our map. We present extensive experiments in which we evaluate our approach using real world data recorded in a garage parking scenario with an autonomous car as well as a robot localization problem in an indoor environment. The results demonstrate that our approach can cope with high sensor noise and that it achieves comparable localization accuracy while at the same time consuming only a fraction of memory compared to regular occupancy grid maps.**

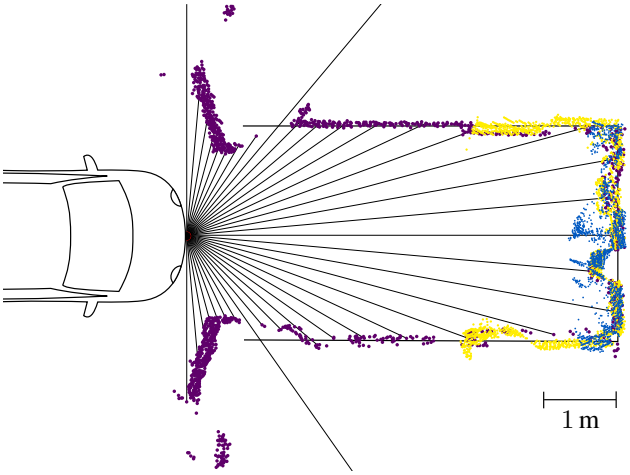

Fig. 1. Visualization of laser scans from varying distances to the parking position, whereas the colors identify different scans. The noise level increases with decreasing distances, from violet (far distance) to blue (short distance).

I. Introduction

The problem of map learning has a long history in mobile robotics. One of the most popular map representation approaches are occupancy grid maps [14]. While these maps have several desirable features including the ability to be updated in a probabilistically sound way, one of their disadvantages is that their memory consumption grows quadratically with the range of the sensor. Whereas this problem can be relaxed by using hierarchical structures like quadtrees, in which cells with the same state or similar values are grouped together [19], such methods do not allow efficient pre-hashing of the maps, i.e., hash-maps for distance queries. Moreover, the quality and usability of grid-based maps highly depends on the chosen discretization level.

In this work, we aim to overcome these problems by using a scan-based map representation, in which we only store a small number of scans from selected poses for localization with a Monte Carlo localization (MCL) process [7]. Our approach is a probabilistically sound method that seeks to find the optimal set of scans so as to maximize the robot localization accuracy. The advantages of our method are twofold: first, it needs substantially less memory than occupancy grid maps; second, the scans can be chosen according to the noise characteristics of the sensor employed, which follows the idea that noise characteristics can be implicitly encoded in the map. The experiments described in this paper show that our method achieves accurate localization estimates within a Monte Carlo localization process. It is particularly suited for situations in which the robot repeatedly needs to localize on similar trajectories such as repeatedly parking in a garage or approaching a work bench. The first of the two targeted application domains, a garage parking scenario with an autonomous car based on a multilayered 3D LiDAR sensor, is shown in Fig. 1. We are convinced that the resource-efficient nature of algorithms will play an important role in the near future, as soon as autonomous systems become part of our daily life and therefore will require mass production.

In summary, the contributions of this work are a memory-efficient map representation based on individual scans, a probabilistic sound approach that incrementally selects scans, and a sophisticated sensor-model for use within the MCL. This paper is organized as follows. After discussing related work in the following section, we present our approach to localization with sparse maps in Sec. III. We then describe different approaches to the mapping problem in Sec. IV. In Sec. V, we evaluate the performance in two real-world experiments.

II. Related Work

A common approach for building a range based representation of the environment is to use grid maps. These rely on calculated occupancy values in a discrete grid, which can be 2D [14] or 3D [19].

Droeschel et al. [6] presented an approach in which the robot stores a multi-resolution grid map of the environment.

All authors are with the Department of Computer Science at the University of Freiburg, Germany. This work has partly been supported by the European Commission under FP7-610917-STAMINA.

They provide a high resolution in nearby areas whereas the resolution decreases with growing distance to the robot. This approach enables a computationally efficient localization. The total amount of points for all potential grid maps and therefore the memory will remain unaffected. Our approach focuses on filtering out information before localizing the robot and thus decreases the memory that is needed in total. Other approaches that potentially decrease the computational effort in localization segment the map in different regions [4, 20]. These can be afterwards used for an efficient multi-robot exploration [18]. A robot only localizes based on a map of a single region after computing a rough estimate of its current position. In comparison to our work, these approaches still keep the full map even if they discard parts of it for different time steps.

Biber and Straßer [3] use a grid structure to represent laser scans by a collection of normal distributions. This representation allows for an analytic formulation of scan matching and relaxes the burden of correspondence search. In comparison to our work, the focus of their method is the sensor data representation and scan matching. Our approach aims to select observations based on the provided information to achieve accurate localization. When selecting laser scans, our approach weights each scan relative to the additional information of the already selected scans. This step is performed on raw laser measurements without any filtering or feature selection techniques [17, 5]. After this selection step, we can localize the robot based on the computed map. The approaches of Kretzschmar and Stachniss [9] and Ila et al. [8] consider the problem of a SLAM pose graph where the amount of nodes is minimized. Kretzschmar and Stachniss [9] take the mutual information between map and measurements into account in order to discard the least informative laser scans. Ila et al. [8] incorporate new measurements during runtime instead of marginalizing out already included ones.

Alternative approaches save landmarks extracted from the measurement data [1, 10, 13]. Beinhofer et al. [1] developed an approach that proposes the placement of artificial landmarks in order to improve localization. Whereas grids consume too much space, the landmarks typically need to be known beforehand and the manual placement of proper landmarks does not appear to be a viable option for the scenario of autonomous parking with a self-driving car.

An alternative to probabilistic approaches to localization are teach-and-repeat scenarios, in which the vehicle follows a demonstrated trajectory [12, 16]. In the approach by Sprunk et al. [16], the robot localizes itself relative to anchor-points, i.e., poses with corresponding laser scans. However, during the selection of anchor-points, the authors do not consider memory requirements and rather store anchor-points after exceedance of a fixed threshold. In contrast to our approach, they neither consider the properties of the environment nor the sensor when determining whether a new scan needs to be recorded. Moreover, the goal of teach-and-repeat approaches is to follow a certain trajectory instead of representing the environment for an arbitrary navigation scenario.

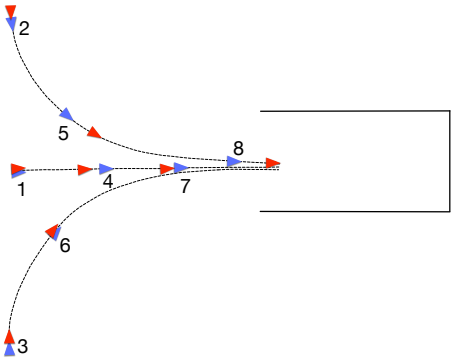

Fig. 2. Resulting positions (markers) of map scans as selected by our probabilistic strategy (blue) and a clustering approach (red) in the garage parking scenario. The dotted lines refer to potential positions at which map scans can be selected. Both approaches choose the scans at similar positions in the environment. The numbers refer to the order of selection from the greedy search of our probabilistic approach.

III. Localization With Sparse Scan Maps

In this paper, we pursue two objectives: first, a memory efficient representation of the map, and second, robustness against sensor noise. To achieve both, we select a (small) set of pose-scan tuples captured in the environment as map representation. Concerning the memory consumption, let us consider a dense grid map that captures an area of $20 \times 20\,\mathrm{m}$ with a resolution of $0.01\,\mathrm{m}$. This results in a memory usage of $15\,\mathrm{MiB}$. In contrast to that, a map consisting of 10 scans, with 540 2D-points each, results in a memory usage of $0.04\,\mathrm{MiB}$. The difference in memory consumption is about two orders of magnitude. Furthermore, we assume that the localization trajectory is similar to the mapping trajectory, meaning that we compare scans taken at similar positions, which then exhibit similar noise characteristics. For an intuition about the extent of noise we face in one of our datasets, refer to Fig. 1, in which points measured from a short distance exhibit differences of $\sim 20\,\mathrm{cm}$.

In the remainder of this section, we formally introduce the SSM (Sparse Scan Map) in Sec. III-A and discuss how to perform a MCL using such maps in Sec. III-B and Sec. III-C.

A. Sparse Scan Map

In this work, we consider a regular mapping with a known-poses scenario. Therefor, we assume a series of known poses $x_{1:t}$, each associated with a sensor measurement $z_{1:t}$, as given. The poses x_i are represented in $\mathrm{SE}(2)$ and the measurements z_i are represented by a set of points in the robot frame. Let $\mathcal{S} = \{(x_i, z_i) \mid i = 1, \ldots, t\}$ be the set of all pose-scan tuples. A Sparse Scan Map (SSM) $\mathcal{S}_n \subset \mathcal{S}$, the map representation we introduce in this work, is a subset of $\mathcal{S}$ that contains n scan-pose tuples, meaning $|\mathcal{S}_n| = n$. The sparseness relates to the fact that we consider a small value for n, which is only a small fraction of the available data. The formal definition of $\mathcal{S}_n$ can be expressed as:

$$\mathcal{S}_n = \{(x_{j_1}, z_{j_1}), \ldots, (x_{j_n}, z_{j_n}) \mid j_i \in \{1, \ldots, t\}\} \quad (1)$$

This defines an element of $\mathcal{P}_n(\mathcal{S})$, which is the set of subsets of $\mathcal{S}$ with cardinality n. When we use such maps for

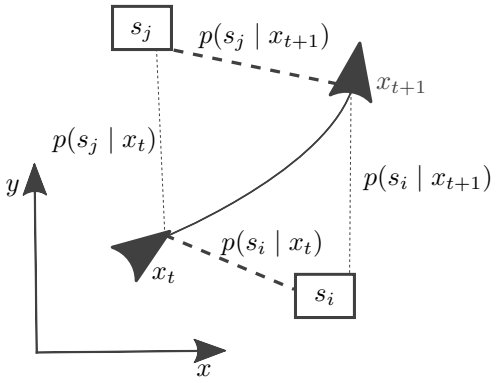

Fig. 3. For every pose x we calculate a weight $p(s \mid x)$ with respect to each map scan s in order to integrate multiple map scans in the measurement model. This weight is based on the spatial relation between x and the pose of s. The stroke width visualizes the weight.

localization, the accuracy will highly depend on the tuples selected to represent the environment. This constitutes a hard combinatorial problem as there exist $|\mathcal{P}_n(\mathcal{S})| = \binom{t}{n}$ possible combinations. The idea is to choose the map scans in a way that, for a given n, the set $\mathcal{S}_n$ represents the environment. This means that we can accurately localize a robot that repeatedly executes similar trajectories as the one used for mapping. We will discuss the selection of map scans in Sec. IV.

B. Localization

In this section, we explain how to use a SSM for mobile robot localization. The probabilistic formulation for localization given the SSM $\mathcal{S}_n$ is to estimate the distribution over the robot's position at time t, $Bel_{\mathcal{S}_n}(x_t)$, given the observations $z_{1:t}$, controls $u_{1:t}$ and the map $\mathcal{S}_n$, specifically $Bel_{\mathcal{S}_n}(x_t) = p(x_t \mid z_{1:t}, u_{1:t}, \mathcal{S}_n)$. With the Bayesian filter we can recursively estimate this distribution.

$$p(x_t \mid z_{1:t}, u_{1:t}, \mathcal{S}_n) = \eta_t p(z_t \mid x_t, \mathcal{S}_n) \cdot \int p(x_t \mid x_{t-1}, u_t) p(x_{t-1} \mid z_{1:t-1}, u_{1:t-1}, \mathcal{S}_n) \mathrm{d}x_{t-1} \quad (2)$$

In our current implementation, we employ the Monte Carlo Localization (MCL) framework [7] to estimate the distribution $Bel_{\mathcal{S}_n}(x_t)$. MCL represents this distribution by a set of weighted samples of pose hypotheses. The population at time t represents the distribution $Bel_{\mathcal{S}_n}(x_{t-1})$. Whenever a motion measurement u_t arrives, we sample the next generation from the current population according to the regular motion model $p(x_t \mid x_{t-1}, u_t)$. When the robot makes a new laser observation, we perform a measurement update, where each particle is weighted with an importance weight $\mathbf{w} \propto p(z_t \mid x_t, \mathcal{S}_n)$. A stochastic universal resampling is triggered if and only if the so-called number of effective particles $n_{\mathit{eff}} = \left(\sum \mathbf{w}_i^2\right)^{-1}$ is less than half the population size.

C. Measurement Model

To implement the measurement model for the SSM representation, we employ a distance measure that is regularly used for scan matching [15]. Given a scan $s \in \mathcal{S}_n$, the observation likelihood of an individual laser beam is based on the minimal distance between the end point of the beam and the points in s. As the map $\mathcal{S}_n$ stores multiple scans, we are faced with an association problem, to which $s \in \mathcal{S}_n$ our current observation needs to be matched. Because this association is unknown, we model it as a hidden variable:

$$p(z \mid x, \mathcal{S}_n) = \sum_{s \in \mathcal{S}_n} p(z \mid x, s) p(s \mid x). \quad (3)$$

Intuitively, we match the current observation z against the scans $s \in \mathcal{S}_n$ and weight each likelihood according to the probability $p(s \mid x)$, which is the probability that scan s is associated to the pose x. This can be interpreted as a prior for seeing scan s from pose x. A graphical example is depicted in Fig. 3. In addition, $p(z \mid x, s)$ is the likelihood of the current observation given the scan $s \in \mathcal{S}_n$.

To compute $p(z \mid x, s)$, we make use of the common assumption that the measurements of individual points q in the scan z are independent.

$$p(z \mid x, s) = \prod_{q \in z} p(q \mid x, s) \quad (4)$$

We calculate $p(q \mid x, s)$ by employing a Gaussian distribution and taking into account the distance to the nearest neighbor in the set of points s:

$$p(q \mid x, s) \sim \mathcal{N}\left(\min_{q_s \in s} \|q_s - x \oplus q\|_2; 0, \sigma\right). \quad (5)$$

$x \oplus q$ transforms the point q from the vehicle frame to the world frame. We use a k-d-tree [2] for the nearest neighbor search. Accordingly, the expected runtime for the evaluation of Eq. (4) is $\mathcal{O}(nK \log(K))$, with $K = |z|$.

In our implementation, the likelihood $p(s \mid x)$ of a map scan s given the pose x is modeled as a Gaussian distribution.

$$p(s \mid x) \propto \mathcal{N}(x, \Sigma) \quad (6)$$

IV. Sparse Scan Mapping

In the previous section, we described a localization technique using SSM. Here, we discuss the mapping problem, which is equivalent to the selection of a scan-set $\mathcal{S}_n$ for a given set size n. We provide three different strategies to solve this task. First, a simple heuristic that selects the scans equidistantly distributed along the trajectory. Second, a method that aims to find clusters of scans, grouping scans that observe similar parts of the environment. Finally, we propose a technique that aims to maximize the observation likelihood along the mapping trajectory.

A. Equidistant

The first strategy for the selection of scans is to distribute them equidistantly over the mapping trajectory. We include this simple strategy because in the past it was successfully applied in teach and repeat settings [16]. Given the trajectory $x_{1:t}$ we compute the path length $l = \sum \|x_{i+1} - x_i\|_2$ and subdivide it in $n+1$ chunks of equal length l/n. Starting from x_1, we recursively select the first pose that has a distance

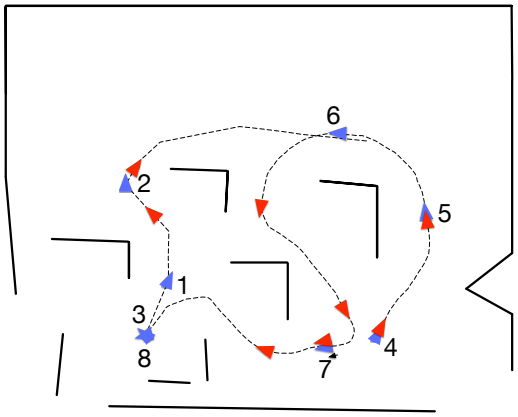

Fig. 4. Resulting selection of map scan poses (markers) of our probabilistic approach (blue) in comparison to a clustering of laser scans (red) for the indoor dataset. The numbers refer to the order of selection from the greedy search of our probabilistic approach.

larger than l/n to the previously chosen pose in trajectory space. This strategy neglects the information contained in the observations and instead purely relies on the mapping trajectory for the selection of $\mathcal{S}_n$.

B. Scan Clustering

The scan clustering strategy takes the observed scans of the mapping data into account and aims to group poses for which the scans observe similar parts of the environment. Therefore, we use the K-medoids clustering technique to determine the n pose-scan tuples that form $\mathcal{S}_n$.

For K-medoids clustering, we need a set of objects, a distance measure for those objects $d(\cdot,\cdot)$, and the desired number of clusters K. The algorithm iteratively assigns the objects to the clusters, which is mutually exclusive, and determines the new medoids. The algorithm terminates if the medoids do not change or a maximum number of iterations is exceeded. Formally, given a set of medoids $c_1,\ldots,c_n$ the scan s is assigned to cluster index

$$i^\star = \operatorname*{argmin}_i d(c_i, s), \tag{7}$$

augmenting $\mathcal{C}_{i^\star} = \mathcal{C}_{i^\star} \cup \{s\}$. After the cluster assignment, the new medoids are defined via:

$$c_i = \operatorname*{argmin}_{c\in\mathcal{C}_i} \sum_{s\in\mathcal{C}_i} d(c, s) \tag{8}$$

Our implementation uses the symmetric average minimal distance between the points in the scans. For two scans s_i and s_j, this is:

$$d(s_i, s_j) = \frac{1}{|s_i| + |s_j|} \left(\sum_{p\in s_i} d(p, s_j) + \sum_{q\in s_j} d(q, s_i) \right) \tag{9}$$

$$\text{with } d(p, s) = \min_{q\in s} \|p - q\|_2 \tag{10}$$

C. Maximum Likelihood Selection

In this section, we present a strategy that aims to find the set of scans for which the observation likelihood given the mapping trajectory is maximal.

$$\mathcal{S}_n^* = \operatorname*{argmax}_{\mathcal{S}_n\in\mathcal{P}_n} p(z_{1:t} \mid x_{1:t}, \mathcal{S}_n) \tag{11}$$

The intuition is that we aim to find the set of scans, for which we expect to achieve the best localization accuracy.

First, we factorize the observation likelihood, assuming that observations are conditionally independent given the poses.

$$p(z_{1:t} \mid x_{1:t}, \mathcal{S}_n) = \prod_{i=1}^{t} p(z_i \mid x_i, \mathcal{S}_n) \tag{12}$$

Now, we can calculate the likelihood for a given set $\mathcal{S}_n$ efficiently. Nevertheless, to solve this discrete optimization problem we would need to compute the value of the objective function for each of the $\binom{t}{n}$ possible selections for $\mathcal{S}_n$. As this is not feasible, we deploy a *greedy* strategy to compute an approximate solution of the optimization problem stated in Eq. (11). We select the scan that maximizes the observation likelihood incrementally. Given the current set $\mathcal{S}_i$, $i < n$, we select a pose scan tuple s_{i+1}^* such that

$$s_{i+1}^* = \operatorname*{argmax}_{s\in\mathcal{S}\setminus\mathcal{S}_i} p(z_{1:t} \mid x_{1:t}, u_{1:t}, (\mathcal{S}_i \cup s)), \tag{13}$$

which is quadratic in the number of the scans available. This results in an overall complexity of $\mathcal{O}(t^2 n^2 K \log(K))$.

Please note that this process can be run offline and on an off-board computer. After finishing the optimization, the selected scans can be transferred to the robot for accurate localization with a small number of scans.

V. Experiments

We implemented the proposed algorithms in C++ and tested them in two different scenarios. First, a parking scenario in which a car drives into a garage (parking dataset) starting from three different locations and second, an indoor environment (indoor dataset), courtesy of Mazuran et al. [11]. The indoor dataset exhibits more complexity, i.e., clutter and higher variation regarding the driven trajectories.

For all experiments, we consider the following settings: The initial pose is known within bounds of 1.5 m and 20°. A ground truth, used for the evaluation and mapping poses, is provided by an optical motion capture system from Motion Analysis Digital that consists of ten high speed Raptor-E cameras. For localization, we utilized a MCL using 1,000 particles. Due to the randomized nature of MCL, we repeated each localization run 25 times and report the root-mean-square error (RMSE). We compare the localization accuracy using SSM with each of the different scan selection strategies and a maximum likelihood occupancy grid map with a resolution of 0.1 m.

A. Autonomous Parking

In this scenario, a car needs to find its parking position in a garage. The car was equipped with a Valeo 3D laser scanner with three physical layers and a distance error of $\sigma_d = 10.0\,\text{cm}$. The laser scanner was mounted at the front of the car with a horizontal field of view of $145°$ and a vertical field of view of $2.4°$. We consider three possible starting areas: approaching from the left, center, and right. The trajectories are depicted in Fig. 2 as dotted lines. The dataset consists of five trajectories from each starting area. To evaluate our approach, we compose five mapping trajectories, which are combined from one trajectory of each starting area. On the remaining trajectories, we evaluate the localization accuracy with 1500 localization runs in total. The length of the trajectories varies from $10\,\text{m}$ to $15\,\text{m}$ on an area of $18 \times 18\,\text{m}$. Considering this extent, the grid map (with a height of $0.5\,\text{m}$) consumes $632.8\,\text{KiB}$ while the SSM representation needs $23.4\,\text{KiB}$ per scan stored. Even the largest map $\mathcal{S}_8$ consumes $70.4\,\%$ less memory with $187.5\,\text{KiB}$.

The statistics of the RMSE from the localization evaluation for each mapping strategy and the occupancy grid map is depicted in Fig. 5. The height of the bars shows the mean RMSE and the error bars the respective standard deviation. For this dataset, the occupancy grid map suffers from the high sensor noise, which results in a mean RMSE of almost $0.1\,\text{m}$. Accordingly, all SSM mapping strategies outperform the occupancy grid map for $n \geq 5$. This underpins our conclusion that the SSM is suitable for localization with high sensor noise. For this environment, the clustering and the probabilistic strategy show similar results for set sizes larger than five. For smaller set sizes, the probabilistic strategy exhibits better results, except for a set size of three. We think this is due to the rather simple structure of the environment. Moreover, the accuracy of the probabilistic strategy is monotonically increasing, while the clustering strategy exhibits higher variance in performance for the different set sizes.

It is worth to note, that the mapping strategies, which take the observations into account, achieve better results than the equidistant strategy for set sizes larger than five.

B. Indoor Localization

The indoor dataset was recorded using a holonomic KUKA omniRob equipped with a 2D SICK S300 laser scanner ($\sigma = 2.8\,\text{cm}$), which has a $270°$ field of view. In contrast to the parking scenario, this dataset has a structurally more complex environment which can not be fully observed from a single pose (see Fig. 4). The grid map for this environment of $11 \times 12\,\text{m}$ consumes $51.6\,\text{KiB}$ of memory while the SSM needs $4.2\,\text{KiB}$ per scan stored. This results in $34.3\,\%$ less memory with $33.9\,\text{KiB}$ for $\mathcal{S}_8$.

For this dataset, we use a single mapping trajectory, as depicted by the dotted line in Fig. 4. The localization trajectories are shorter. They contain distributed starting points along the mapping trajectory. In general, this makes the selection of good map scans less obvious.

In this scenario, the grid map performs best, as one would expect it due to the substantially more accurate sensor, with

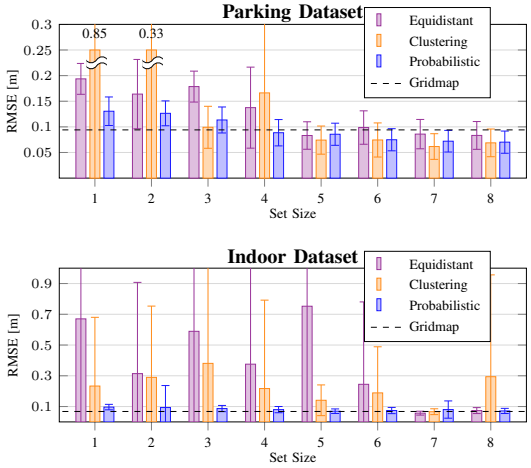

Fig. 5. RMSE of the localization with different sizes of the SSM for the three mapping approaches (equidistant, clustering, probabilistic) and a regular grid map. The bars show the average RMSE with its standard deviation for both datasets. Our probabilistic SSM mapping approach achieves better localization results for multiple map sizes and exhibits consistently low errors with increasing map sizes.

an average RMSE of $9.4\,\text{cm}$. However, as depicted at the bottom of Fig. 5, the probabilistic strategy achieves the best results among the other mapping strategies. Even with a set size of two, the average RMSE is less than $0.1\,\text{m}$ and the standard deviation is with $14.3\,\text{cm}$ comparatively small.

In Fig. 4, we depict the selected map scans for the probabilistic (blue) and the clustering (red) strategy. The probabilistic strategy distributes the scans along the left and right border of the trajectory, since the center part of the map is sufficiently observed. In contrast to that, the clustering suffers from the cluttered structure of the environment's lower part. The RMSE results reflect this, as the probabilistic strategy achieves consistently good results in comparison to the occupancy grid. Depending on the set size, the equidistant and clustering strategy exhibit a high variance in performance.

In combination with the previous experiment, we see a clear advantage for the probabilistic strategy, as it consistently achieves good results for both kind of situations and needs less memory than a dense occupancy grid map.

C. Sensor Model Approximation

In this experiment, we evaluate the gain of our sensor model against the approximation of $p(s \mid x)$ with a Dirac distribution that is equal to 1 if and only if $\|x - s\|_\Sigma = \min_{s'} \|x - s'\|_\Sigma$. In the following, we call this approach the approximate sensor model.

Comparing the localization accuracy (see Fig. 6), the full sensor model shows its advantage for larger set sizes on both datasets. As both models are equivalent for a set size of one, they perform similar for a set size up to three. For larger set sizes, the full sensor model is superior to the approximate sensor model. On the other hand, the full sensor model needs more computation time, as we match more scans for each update. An update of the approximate sensor model takes on average $0.78\,\text{ms}$, which increases for the full sensor model,

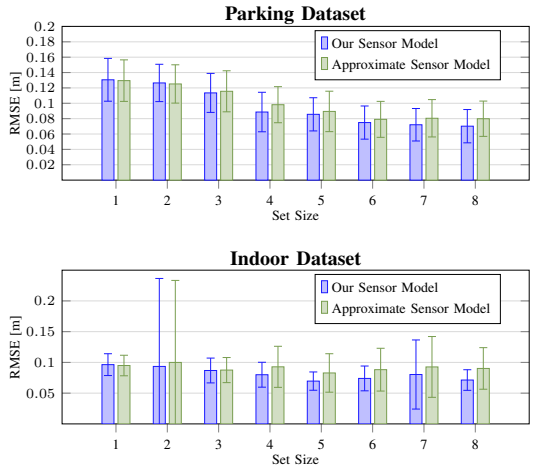

Fig. 6. RMSE of the localization with our probabilistic approach using the full sensor model (blue) and an approximate sensor model (green) that matches only against the most likely scan, according to $p(s \mid x)$.

using a set size of eight, by a factor of 2.5, as we also neglect scans with $p(s \mid x) < 10^{-4}$ in the full sensor model.

D. Scalability to larger areas

We conduct further experiments to show the scalability of our approach in larger areas with more potential scans for the SSM. Therefor, we use the additional indoor dataset fr-079 which consists of a single trajectory of 422 m on an area of 41×17 m. The dataset contains 4789 2D laser scans.

Tab. I shows runtimes of the Maximum Likelihood Selection (see Sec. IV-C) on three different datasets. The experiments were conducted on a Intel Xeon CPU E5-2680 with 12 cores. On the indoor dataset fr-079, the selection of eight scans for the SSM takes only 40.6 min. As stated in Sec. IV-C, the worst-case complexity is quadratic in the number of scans n. The runtime for our approach is in practice less since each observation does not need to be matched against all scans in the SSM. We can localize the robot in the fr-079 environment with an RMSE of 9.8 cm over 25 runs using a SSM with 55 scans. The runtime for the Maximum Likelihood Selection is 7.1 h and results in a memory consumption of 154.7 KiB with $\mathcal{S}_{55}$. The grid map approach achieves a localization accuracy of 6.7 cm and consumes 272.3 KiB of memory. This results in 43.2 % less memory with the SSM approach.

VI. Conclusions

In this paper, we presented a probabilistically sound framework for scan-based mapping and localization. Our approach greedily selects scans based on their additional information relative to the previously selected scans. For localization, we use a sophisticated sensor-model, which is tailored to the proposed map representation. The experimental evaluation shows that our approach, representing the environment with only a few scans, achieves accurate localization results. These are comparable to the performance of grid-based maps while the SSM demands substantially less memory. Moreover, our probabilistic approach outperforms baseline heuristics as it achieves consistently accurate results.

$\lvert\mathcal{S}_n\rvert$	Parking [715]	Indoor [4263]	fr-079 [4789]
1	0.4 min	0.67 min	4.35 min
4	2.08 min	4.13 min	19.13 min
8	5.47 min	10.32 min	40.63 min

TABLE I. Run-times of the probabilistic approach for the computation of SSMs with different sizes on three datasets. The numbers next to the dataset names indicate the amount of potential scans for the SSM. Note, that the number of observations considered for the likelihood calculation varies between the datasets based on the traveled distance. We consider 136 (Parking), 474 (Indoor), and 4001 (fr-079) observations.

References

[1] M. Beinhofer, J. Müller, and W. Burgard. Near-optimal landmark selection for mobile robot navigation. In *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2011.

[2] J. L. Bentley. Multidimensional binary search trees used for associative searching. *Communications of the ACM*, 1975.

[3] P. Biber and W. Straßer. The normal distributions transform: A new approach to laser scan matching. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2003.

[4] E. Brunskill, T. Kollar, and N. Roy. Topological mapping using spectral clustering and classification. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2007.

[5] N. Chehata, L. Guo, and C. Mallet. Airborne lidar feature selection for urban classification using random forests. *Int. Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2009.

[6] D. Droeschel, J. Stückler, and S. Behnke. Local multi-resolution representation for 6d motion estimation and mapping with a continuously rotating 3d laser scanner. In *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2014.

[7] D. Fox, W. Burgard, F. Dellaert, and S. Thrun. Monte carlo localization: Efficient position estimation for mobile robots. *Proc. of the National Conference on Artificial Intelligence (AAAI)*, 1999.

[8] V. Ila, J. M. Porta, and J. Andrade-Cetto. Information-based compact pose slam. *IEEE Transactions on Robotics*, 2010.

[9] H. Kretzschmar and C. Stachniss. Information-theoretic compression of pose graphs for laser-based slam. *Int. J. of Robotics Research*, 2012.

[10] P. Malheiros, P. Costa, A. P. Moreira, and M. Ferreira. Robust and real-time teaching of industrial robots for mass customisation manufacturing using stereoscopic vision. In *Proc. of the IEEE Conf. on Industrial Electronics*, 2009.

[11] M. Mazuran, F. Boniardi, W. Burgard, and G. D. Tipaldi. Relative topometric localization in globally inconsistent maps. In *Proc. of the Int. Symp. of Robotics Research*, 2015.

[12] C. McManus, P. Furgale, B. Stenning, and T. D. Barfoot. Lighting-invariant visual teach and repeat using appearance-based lidar. *Journal on Field Robotics*, 2013.

[13] D. Meyer-Delius, M. Beinhofer, A. Kleiner, and W. Burgard. Using artificial landmarks to reduce the ambiguity in the environment of a mobile robot. In *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2011.

[14] H. Moravec and A. Elfes. High resolution maps from wide angle sonar. In *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, 1985.

[15] A. Segal, D. Haehnel, and S. Thrun. Generalized-icp. In *Proc. of Robotics: Science and Systems*, 2009.

[16] C. Sprunk, G. D. Tipaldi, A. Cherubini, and W. Burgard. Lidar-based teach-and-repeat of mobile robot trajectories. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2013.

[17] G. Vosselman, B. G. Gorte, G. Sithole, and T. Rabbani. Recognising structure in laser scanner point clouds. *Int. Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2004.

[18] K. M. Wurm, C. Stachniss, and W. Burgard. Coordinated multi-robot exploration using a segmentation of the environment. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2008.

[19] K. Wurm, A. Hornung, M. Bennewitz, C. Stachniss, and W. Burgard. OctoMap: A probabilistic, flexible, and compact 3D map representation for robotic systems. In *ICRA Workshop on Best Practice in 3D Perception and Modeling for Mobile Manipulation*, 2010.

[20] Z. Zivkovic, B. Bakker, and B. Krose. Hierarchical map building and planning based on graph partitioning. In *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2006.