Virtual Maps for Autonomous Exploration with Pose SLAM

Jinkun Wang, Tixiao Shan and Brendan Englot

***Abstract*— We consider the problem of autonomous mobile robot exploration in an unknown environment taking into account the robot's mapping rate, map uncertainty, and state estimation uncertainty. This paper presents an exploration framework built upon segment-aided pose SLAM adapted for better active localization. We build on our previous work on expectation maximization (EM) exploration, which explicitly models unknown landmarks as latent variables and predicts their expected uncertainty, to resolve the lack of landmark state in denser instances of SLAM. The proposed system comprises path generation, place recognition forecasting, belief propagation and utility evaluation using a virtual map. We analyze the performance in simulated experiments, showing that our algorithm maintains higher coverage speed in exploration as well as lower mapping and localization error. The real-time applicability is demonstrated on an unmanned ground vehicle.**

I. Introduction

Simultaneous localization and mapping (SLAM) has been well studied in theory and applied successfully on real sensing platforms for state estimation and map-building using data collected passively [1]. However, it's still a challenge for an autonomous vehicle to actively map an unknown environment, properly managing the trade-off between exploration speed and state estimation quality. Improving the capability of autonomous exploration is beneficial for many robot mapping tasks, especially in scenarios where teleoperation is limited or infeasible due to constrained communication, e.g., in unknown subsea environments with underwater robots.

The autonomous exploration problem is generally solved in three stages: path generation, utility evaluation and execution. First, we identify candidate waypoints or generate a sequence of actions to follow, which is typically achieved by enumerating frontiers or by employing sampling-based path planning methods. The selected path is usually straightforward to execute using feedback controllers, thus leaving us with a fundamental problem of designing a utility function to measure path optimality. Essentially, it should capture the exploration-exploitation dilemma, i.e., a balancing of visiting unknown areas to reduce map uncertainty, and revisiting known areas to seek better localization (and map accuracy).

The simplified problem, planning with a priori maps, has also been discussed to actively minimize the uncertainty of known landmarks [2]–[4]. Similarly, planning in unknown environments but with predefined waypoints is investigated in [5]. If we don't consider the uncertainty of a robot's state and map, the problem has been approached by following the nearest frontier [6], choosing sensing actions to maximize mutual information [7], [8], using Cauchy-Schwarz quadratic mutual information (CSQMI) to reduce computation time [9], and by exploring on continuous Gaussian process frontier maps [10]. Most of the existing research on exploration in unknown environments takes advantage of occupancy grid maps, considering utility functions involving map entropy and robot pose uncertainty [11], [12]. This paradigm has succeeded in complex applications, including real-time 3D exploration and structure mapping with micro aerial vehicles [13], [14]. However, inaccurate state estimation is likely to result in a complete yet distorted map regardless of exploration speed. The correlation between localization and information gain is often taken into account by integrating over a map's entropy weighted by pose uncertainty [15]–[17]. However, in prior work we have shown that weighted entropy fails to capture the estimation error of landmarks existing in the map, and map inaccuraries may result [18].

Predicing the impact of future actions on system uncertainty remains an open problem [1], and this is particularly true for unobserved landmarks. Our previous work on expectation-maximization (EM) exploration [18] introduced the concept of a *virtual map* composed of virtual landmarks acting as proxy for a real feature-based map, on which we are able to predict the uncertainty resulting from future sensing actions. Since every virtual landmark is deeply connected with robot poses that can observe it, the metrics for exploration and localization are unified as the determinant of the virtual landmarks' error covariance matrix. In this present work, we consider a more realistic dense navigation scenario, in which exploration with range-sensing mobile robots must rely on pose SLAM that does not incorporate features explicity. In such scenarios, SegMap [19], [20] provides a convenient utility for place recognition. Map segments can serve as landmarks, and loop-closures from segment matching allow us to perform active localization efficiently. We improve our previous work on EM exploration to accommodate dense observations using segment-aided LiDAR mapping. The contributions of this paper are twofold:

- A framework for exploration using pose SLAM considering both exploration efficiency and mapping accuracy, that is feasible in dense navigation scenarios;
- A thorough evaluation of our framework's performance in simulated environments, and demonstration of its functionality and real-time viability in experiments using an unmanned ground vehicle (UGV);

The rest of the paper is organized as follows. We define the addressed problem in Sec. II, and provide an overview of the SLAM framework we use in Sec. III. The derivation if EM

J. Wang, T. Shan and B. Englot are with the Department of Mechanical Engineering, Stevens Institute of Technology, Castle Point on Hudson, Hoboken NJ 07030, USA, {jwang92, tshan3, benglot}@stevens.edu.

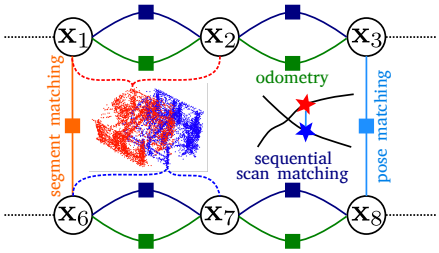


Fig. 1: Overview of the segment-aided LiDAR mapping used for exploration. The factor graph includes sequential factors from odometry (green) and sequential scan matching (blue), and place recognition factors from segment matching (orange) and pose matching (cyan).

exploration on a virtual map is presented in Sec. IV, which is further elaborated in Sec. V. Experiments and results with a range-sensing UGV are presented in Sec. VI.

II. Problem Description

In this paper we address the problem of autonomous exploration for a range-sensing mobile robot in an initially unknown environment. Our robot performs SLAM and constructs an occupancy grid map as it explores. We assume a bounded 2D space $M \subset \mathbb{R}^2$ where all discretized cells $\mathbf{m}_i$ are initialized as *unknown* $P(m_i = 1) = 0.5$. A *frontier* is defined as the boundary where free space meets unmapped space. The exploration is considered complete if no frontier can be detected. However, highly uncertain poses are likely to result in complete, yet inaccurate occupancy grid maps, limiting the usefulness of information gained by exploring unknown space. Assuming the environment contains individual landmarks $L = \{\mathbf{l}_k\}$, apart from discovering more landmarks, minimizing the estimation error is equally crucial. Although landmarks are explicitly incorporated into SLAM in our previous work [18], we relax this requirement here and propose to manage uncertainty through pose graph SLAM.

III. Segment-aided LiDAR Mapping

Although our EM exploration algorithm does not rely on any specific SLAM framework, a fundamental requirement is the need to predict the resulting uncertainty of future robot poses if a sequence of sensing actions is executed. Typically during exploration, loop closure constraints, such as the re-observation of a landmark, are desired with some regularity for better localization. However, not all real-world exploration problems can reason scalably about individual landmarks. Thus we propose to use segment-aided LiDAR mapping [20] to support decision-making in exploration, which optimizes a pose graph but is capable of detecting and handling re-observation as in landmark-based SLAM.

A. System Overview

The diagram of our proposed SLAM system is shown in Fig. 1. The backbone of the pose graph is composed of two sequential factors. The *odometry* factor ($\mathbf{f}^O$) defines the relative motion constraint between two consecutive poses from persistent odometry measurements. Besides odometry, when the robot is equipped with a 3D LiDAR, *sequential scan matching* ($\mathbf{f}^{\text{SSM}}$) also provides a relative transformation by aligning point clouds observed at two positions. The essential component of graph SLAM to ensure accurate estimation is loop closure, which is incorporated in two ways. First, when the current position of the robot is in the vicinity of a previously visited position, *pose matching* ($\mathbf{f}^{\text{PM}}$) is performed by matching two point clouds accumulated around these two positions. Secondly, *segment matching* ($\mathbf{f}^{\text{SM}}$) is utilized for loop-closure, and details on segmentation and segment association are elaborated in Sec. III-B. Overall, the factor graph can be expressed as

$$\mathbf{f}(\mathbf{\Theta}) = \mathbf{f}^0(\mathbf{\Theta}_0) \prod_i \mathbf{f}_i^{\text{O}}(\mathbf{\Theta}_i) \prod_j \mathbf{f}_j^{\text{SSM}}(\mathbf{\Theta}_j) \qquad \text{(sequential)}$$

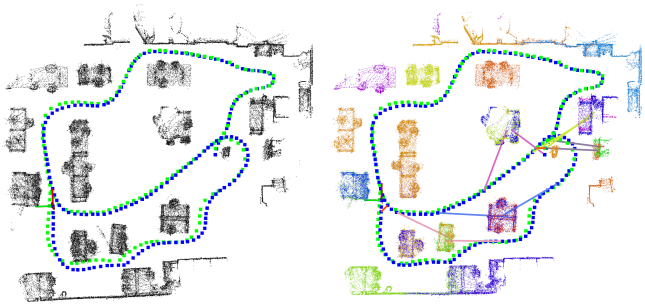
$$\prod_p \mathbf{f}_p^{\text{PM}}(\mathbf{\Theta}_p) \prod_q \mathbf{f}_q^{\text{SM}}(\mathbf{\Theta}_q), \qquad \text{(loop closures)}$$

where variables $\mathbf{\Theta}$ contain 6-DOF robot poses, and every factor $\mathbf{f}_i(\mathbf{\Theta}_i)$ defines a constraint model on a set of variables $\mathbf{\Theta}_i$. The optimization of a factor graph leads to a nonlinear least-squares problem, which can be solved efficiently using iSAM2 [21]. Sequential scan matching is performed using the iterative closest point (ICP) algorithm.

B. Segmentation and Matching

The segmentation and matching scheme follows *SegMatch* as proposed in [22], with some adjustments such that place recognition occurs aggressively and more frequently in indoor environments. We first remove the ground in a point cloud by fitting a plane using points appearing near ground level with the knowledge of sensor displacement and orientation. Euclidean cluster extraction is then performed on non-ground points on the i-th time-step to divide the points into clusters $\{C_1^i, C_2^i, ...\}$, and a voxel grid is constructed from each cluster denoted as $v(C)$. Two clusters detected in sequential frames are considered to be parts of the same segment when their corresponding voxel grids overlap by a certain number of voxels, or $v(C^i) \cap v(C^{i+1}) > v_{\text{thresh}}$. Let C_j^i be the cluster in the j-th segment observed at the i-th time-step, then a complete segment can be represented as $S_j = \{C_j^i, C_j^{i+1}, ..., C_j^{i+N_j}\}$, and all cluster points are anchored at the first step i given relative pose information via state estimation.

Feature descriptors (e.g., Ensemble of Shape Functions (ESF)) computed on a segment's point cloud provide one approach to determining if two segments are from the same object. Similarly, if we assume our state estimation has limited drift in an indoor environment, associated segments are likely to be spatially close to each other. Thus, we intentionally remove geometric consistency verification in [20], and instead S_j and S_k are matched by sequentially verifying the following distances (1) $\Delta c = |\mathbf{c}(S_j) - \mathbf{c}(S_k)|$ where $\mathbf{c}(S)$ is the segment centroid, (2) $\Delta f = |\mathbf{f}(S_j) - \mathbf{f}(S_k)|$ where $\mathbf{f}$ is a feature descriptor vector, and (3) $\Delta v = v(S_j) \cap v(S_k)$ where the voxel grid v is constructed using the entire cluster of points in a segment. A successful segment matching results in a SM factor describing point cloud registration over two segments. An example is given in Fig. 1: two



(a) No place recognition (b) Add SM and PM

Fig. 2: Segment-aided LiDAR mapping. The ground truth trajectory (green) is obtained via LeGO-LOAM [32] using a full-range (100m) LiDAR point cloud, while our estimated trajectory (blue) relies on points with a 3m cutoff distance. The drift is corrected by adding *segment matching* (two lines with the same color connecting pose and segment), and *pose matching* (red lines connecting two poses).

segments, blue and red are tracked at poses $\mathbf{x}_1$ – $\mathbf{x}_2$ and $\mathbf{x}_6$ – $\mathbf{x}_7$ respectively, and segment matching produces a factor connecting $\mathbf{x}_1$ and $\mathbf{x}_6$.

IV. EM Exploration

In the formulation of the SLAM problem as a *belief net* [23], the solution is obtained by maximizing the joint probability distribution,

$$X^*, L^* = \underset{X,L}{\operatorname{argmax}} \ \log P(X, L, Z), \tag{1}$$

where X, L, Z are robot poses, landmarks, and measurements respectively. During exploration, we are confronted with unknown landmarks that haven't been observed yet. Therefore, we introduce the concept of *virtual landmarks* V as latent variables, which describe potential landmark positions that would be observed when following the planned path. Then the objective is to maximize the following marginal model,

$$\begin{aligned} X^* &= \underset{X}{\operatorname{argmax}} \ \log P(X, Z) \\ &= \underset{X}{\operatorname{argmax}} \ \log \sum_V P(X, Z, V). \end{aligned} \tag{2}$$

The above equation involves unobserved variables, which can be approached intuitively using an expectation-maximization (EM) algorithm as follows,

$$\text{E-step:} \quad q(V) = p(V|X^{\text{old}}, Z) \tag{3}$$

$$\text{M-step:} \quad X^{\text{new}} = \arg\max_X \mathbb{E}_{q(V)}[\log P(X, V, Z)]. \tag{4}$$

In the E-step, latent virtual landmarks are computed based on the current estimate of the trajectory and the history of measurements. In the M-step, a new trajectory is selected such that the expected value of joint probability, given the virtual landmark distributions, is maximized. The iterative algorithm alternates between the E-step and M-step, but each iteration is accomplished by the execution of actions and the collection of measurements.

The equation above poses a challenge for efficient solution due to the exponential growth of potential virtual landmark configurations with respect to the number of virtual landmarks. Inspired by classification EM algorithms [24], an alternative solution would add a classification step (C-step) before the M-step to provide the maximum posterior probability estimate of the virtual landmark distributions,

$$\text{C-step:} \quad V^* = \underset{V}{\operatorname{argmax}} \ p(V|X^{\text{old}}, Z) \tag{5}$$

$$\text{M-step:} \quad X^{\text{new}} = \arg\max_X \ \log P(X, V^*, Z). \tag{6}$$

If we further assume measurements are assigned to maximize the likelihood $\operatorname{argmax}_Z h(X, V)$, then the joint distribution can be expressed as a multivariate Gaussian centered at the proposed poses and landmark positions, and the covariance can be approximated by the information matrix inverse,

$$P(X, V, Z) = \mathcal{N}\left(\begin{bmatrix} X \\ V \end{bmatrix}, \begin{bmatrix} \Sigma_{XX} & \Sigma_{XV} \\ \Sigma_{VX} & \Sigma_{VV} \end{bmatrix} \right). \tag{7}$$

The solution of Eq. 6 is equivalent to evaluating the log-determinant of the covariance matrix,

$$\underset{X}{\operatorname{argmax}} \ \log P(X, V^*, Z) = \underset{X}{\operatorname{argmin}} \ \log \det(\Sigma). \tag{8}$$

This implies that the performance metric for our proposed exploration is consistent with the D-optimality criterion in active SLAM [25], except that the subjects considered include unobserved landmarks.

Since we are more interested in the uncertainty of the virtual landmarks and the most recent pose $\mathbf{x}_{T+N}$ at step T with planning horizon N, we can marginalize out irrelevant poses in Σ_{XX}, ending up with $\Sigma_{\mathbf{x}_{T+N}}$. Typically, there exist thousands of virtual landmarks, thus approximation of Σ_{VV} is critical for real-time applications. Combined with pose simplification, we can obtain that, for a positive definite covariance matrix,

$$\log \det(\Sigma) < \log \det(\Sigma_{\mathbf{x}_{T+N}}) + \sum_k \log \det(\Sigma_{\mathbf{v}_k}), \tag{9}$$

where Σ_{V_k} is the diagonal block involving the kth virtual landmark in Σ_{VV}. This approximation is reasonable considering an overestimate of information using the introduced virtual landmarks (see Sec. V-A).

Now we show an approximative estimate of $\Sigma_{\mathbf{v}_k}$. Let $\Sigma_{\mathbf{v}_k}$ be the actual covariance matrix, and let $\Sigma^{(i)}_{\mathbf{v}_k}, \Sigma^{(j)}_{\mathbf{v}_k}$ be the covariance estimates from two individual measurements at poses $\mathbf{x}_i$ and $\mathbf{x}_j$ respectively. As in [26], we define $B_l(\Sigma) \triangleq \{\mathbf{x} \mid \mathbf{x}^T\Sigma^{-1}\mathbf{x} \leq l\}$ as a covariance ellipse of $\Sigma \succ 0$ with level l. Due to additive information, it can be shown that $B_l(\Sigma_{\mathbf{v}_k}) \subset B_l(\Sigma^{(i)}_{\mathbf{v}_k}) \cap B_l(\Sigma^{(j)}_{\mathbf{v}_k})$. The covariance intersection (CI) [26] algorithm was proposed to obtain an upper bound of the covariance estimate without knowledge of the correlation between measurement sources. Given only $\Sigma^{(i)}_{\mathbf{v}_k}, \Sigma^{(j)}_{\mathbf{v}_k}$, CI calculates an upper bound of the actual error covariance as follows,

$$\hat{\Sigma}^{-1}_{\mathbf{v}_k} = \omega {\Sigma^{(i)}_{\mathbf{v}_k}}^{-1} + (1 - \omega){\Sigma^{(j)}_{\mathbf{v}_k}}^{-1}, \tag{10}$$

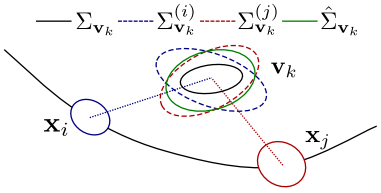


Fig. 3: Covariance intersection (CI). The actual covariance ellipse (black) lies inside the intersection of two individual estimates (blue and red dashed), and CI computes an upper bound (green) of this intersection.

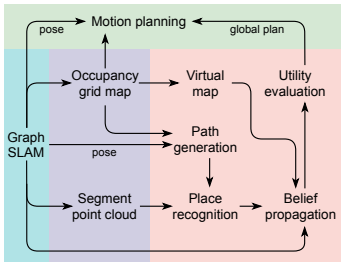


Fig. 4: System overview of our proposed EM-exploration algorithm.

where the weight ω can be optimized to minimize the determinant of $\hat{\Sigma}_{\mathbf{v}_k}$, leading to

$$\omega^* = \frac{2b - ac}{2(a + b - ac)}, \tag{11}$$

$$a = 1/\det(\Sigma^{(i)}_{\mathbf{v}_k}), b = 1/\det(\Sigma^{(j)}_{\mathbf{v}_k}), c = \mathrm{tr}(\Sigma^{(i)}_{\mathbf{v}_k}{\Sigma^{(j)}_{\mathbf{v}_k}}^{-1}). \tag{12}$$

The weighted combination of information matrices implies $B_l(\Sigma_{\mathbf{v}_k}) \subset B_l(\Sigma^{(i)}_{\mathbf{v}_k}) \cap B_l(\Sigma^{(j)}_{\mathbf{v}_k}) \subset B_l(\hat{\Sigma}_{\mathbf{v}_k})$, and it follows that $\det(\Sigma_{\mathbf{v}_k}) < \det(\hat{\Sigma}_{\mathbf{v}_k})$. By iteratively fusing estimates from different poses, we derive an upper bound on the actual covariance in terms of the minimal determinant. The two-step approximation is illustrated in Fig. 3. Therefore, Eq. 9 is transformed to a looser upper bound,

$$\log\det(\Sigma) < \log\det(\Sigma_{\mathbf{x}_{T+N}}) + \sum_k \log\det(\hat{\Sigma}_{\mathbf{v}_k}). \tag{13}$$

From the derivation, we can see the algorithm operates on the basis of three functions: (1) producing a virtual map from the current estimate and measurements, (2) propagating the pose belief under prospective future measurements, and (3) predicting landmark uncertainty. As the last two functions are decoupled, it's perfectly feasible that they employ different sensor models for measuring landmarks.

V. Implementation Details

In this section, we present an implementation of the proposed EM exploration algorithm. The system architecture is illustrated in Fig. 4, and each module contributing to exploration is elaborated in the following subsections.

A. Virtual Map

How can we predict unobserved landmarks without prior knowledge of the characteristics of an environment? We can approach this question by making a conservative assumption that any location which hasn't been mapped yet has a virtual landmark. Therefore, the probability that a location

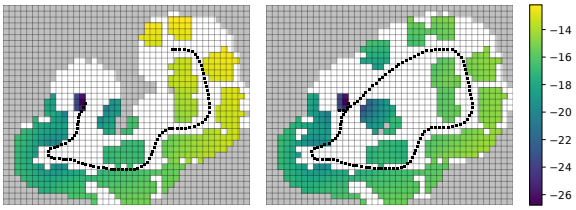


Fig. 5: Virtual landmarks, shown as grid cells with 0.5m resolution. Their uncertainty (measured by covariance log-determinant) is reduced after closing a loop. Unmapped virtual landmarks (gray) have large initial uncertainty outside the spectrum of values depicted.

is potentially occupied with a landmark is strongly related to the traditional occupancy grid map. Let $P(m_i)$ denote the occupancy of a discretized cell, then we define a virtual map V consisting of virtual landmarks with probability

$$P(v_i = 1) = \begin{cases} 1, & \text{if } P(m_i = 1) \geq 0.5 \\ 0, & \text{otherwise.} \end{cases} \tag{14}$$

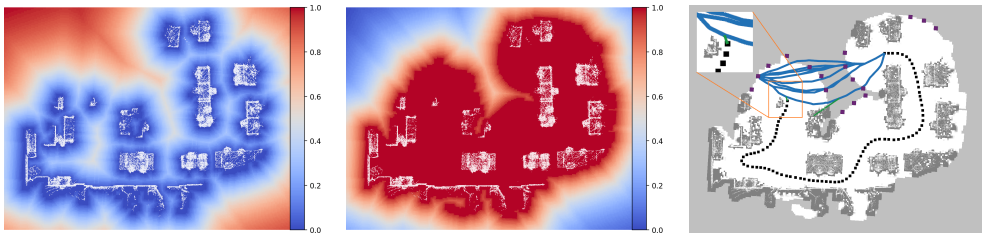
Under circumstances with severe drift, the expected map could be utilized to calculate a weighted average of occupancy grid maps from all trajectory hypotheses, as in our previous work [27]. In Fig. 5, we whiten the grid cells that, once observed, no longer possess virtual landmarks.

In its definition, existing landmarks have been incorporated into the virtual map, which is essential because minimizing the uncertainty of observed landmarks is also desired. What distinguishes an occupancy grid map from the virtual map is the treatment of unknown space, which consequentially determines what metric we use for exploration. In occupancy grid mapping, in order to enable exploration towards unknown space, Shannon entropy is typically leveraged to optimize a path that provides maximum information gain. However, another metric, such as an optimality criterion defined over the covariance matrix, is required to take into account mapping and localization uncertainty. In contrast, virtual landmarks are initialized to have high uncertainty, which is reduced by taking measurements of them, resulting in remarkable information gain. The same metric can be further optimized by improving localization through loop closures, thus unifying the utility measure used in both exploration and localization.

B. Path Generation

In the M-step, given the distribution of virtual landmarks, path candidates are generated and evaluated using our proposed utility function (Sec. V-E). The global paths that are to be followed over a long span of time must take into account two types of actions, exploration and place-revisiting [15]. Exploration actions normally have destinations near frontier locations where mapped cells meet unknown cells, and to reduce localization uncertainty, place-revisiting actions travel back to locations the robot has visited, or where it's able to observe a previously observed map segment. The prevalence of these locations requires us to examine a large number of free grid cells in order to obtain a near-optimal solution.

Therefore, we break the path generation problem into two steps. First, frontiers that are reachable from the robot's



(a) Localization cost $\lambda_l(\mathbf{x})$ (b) Exploration cost $\lambda_e(\mathbf{x})$ (c) Alternative paths

Fig. 6: Alternative paths to a destination are generated from the shortest path on a variety of cost maps by varying weights in a weighted sum of distance, localization and exploration cost. In (c), the alternative paths to a single frontier location are visualized in blue for clarity. Potential segment matching and pose matching are denoted by green lines.

current location are identified as destinations of our path primitives. Next, we refine the path to a frontier location instead of taking the shortest one using multi-objective optimization for path planning. Let $d(\mathbf{x}_i, \mathbf{x}_j)$ be the distance cost between two poses, let $d_{\text{occupied}}(\mathbf{x}), d_{\text{observed}}(\mathbf{x})$ be the distance costs from a pose to the nearest occupied cell and nearest observed cell respectively, and let r_{sensor} be the maximum sensor range. Then we define (1) the cost for localization as $\lambda_l(\mathbf{x}_j)d(\mathbf{x}_i, \mathbf{x}_j)$ where $\lambda_l(\mathbf{x}_j) = 1 - \exp(-a_l d_{\text{occupied}}/r_{\text{sensor}})$, and (2) the cost to explore as $\lambda_e(\mathbf{x}_j)d(\mathbf{x}_i, \mathbf{x}_j)$ where $\lambda_e(\mathbf{x}_j) = \exp(-a_e d_{\text{observed}}/r_{\text{sensor}})$. The cost maps favoring localization and exploration are illustrated in Figs. 6a and 6b.

At current step T, we wish to find a path that minimizes the following weighted sum,

$$\sum_{i=1}^{N} \big(w_0 + w_l\lambda_l(\mathbf{x}_{T+i}) + w_e\lambda_e(\mathbf{x}_{T+i})\big)d(\mathbf{x}_{T+i-1}, \mathbf{x}_{T+i}), \tag{15}$$

where w_0 is added to allow distance to dominate the cost when the other two terms are close to zero. Considering there is no single optimal path for the above objective function, we employ a simple weighted sum approach to explore Pareto optimal solutions [28], by varying weights $w_l \in [0, 1], w_e = 1 - w_l$ and leaving w_0 as constant. The resulting paths are further pruned to retain only those that are well-separated. We build a roadmap in the collision-free configuration space, and Dijkstra's algorithm is used to search for the shortest path with different cost functions. A representative example is shown in Fig. 6c, and it's evident that a few detoured paths are generated to explore unknown space, and to re-observe one map segment and acquire pose matching (green lines).

C. Place Recognition

The place recognition module is designed to achieve accurate prediction of both pose matching factors $\{\tilde{\mathbf{f}}_p^{\text{PM}}\}$ and segment matching factors $\{\tilde{\mathbf{f}}_q^{\text{SM}}\}$ given a predefined trajectory. While predicted pose matching occurs in the same manner as in real SLAM, we don't resolve segment matching by predicting LiDAR measurements using a traditional ray-casting algorithm on a voxel grid associated with a segment $v(S)$. Instead, we approximate the measurement by performing a range search using k-d tree on the entire set of points in a segment, and if we are able to gather enough points within sensor range, we are confident that a cluster will be extracted from this segment. The predicted matching is also validated by subsequent poses, and a segment matching factor will be accepted if the process is successful without interruption following a designated number of sequential poses.

D. Belief Propagation

Belief propagation is concerned with evaluation of Eq. 13 given future sequential odometry factors $\{\tilde{\mathbf{f}}_i^{\text{O}}\}$, and more importantly, loop closure candidates from place recognition $\{\tilde{\mathbf{f}}_p^{\text{PM}}\}$, $\{\tilde{\mathbf{f}}_q^{\text{SM}}\}$. Pose covariance recovery follows a standard update of iSAM2, and we use $\tilde{\Sigma}_{\mathbf{x}_i}$ to denote the covariance estimate after incorporating predicted future factors. Clearly, the segment-aided mapping framework doesn't possess landmarks, so we create an imaginary inverse sensor model $\mathbf{v}_k = h^{-1}(\mathbf{x}_i, \mathbf{z}_{i_k})$ that is able to compute the state of a virtual landmark from measurement z that is corrupted by zero-mean Gaussian noise with covariance Λ. The predicted error covariance for the kth virtual landmark from a measurement at the ith pose is $\Sigma_{\mathbf{v}_k}^{(i)} = A_k^i \tilde{\Sigma}_{\mathbf{x}_i} {A_k^i}^T + B_k^{i_k} \Lambda {B_k^{i_k}}^T$, and $A_k^i, B_k^{i_k}$ are Jacobian matrices of the inverse sensor model with respect to pose and landmark. Eventually, we obtain $\hat{\Sigma}_{\mathbf{v}_k}$ by fusing all individual estimates from poses that can observe the kth landmark using CI in Eq. 10. Unobserved landmarks will still have large initial covariance. We illustrate the uncertainty reduction due to loop closure in Fig. 5.

E. Utility Evaluation

As discussed in Sec.V-B, a variety of candidate paths are generated and we select the best one among them according to a utility function that maps a path to a scalar. In Eq.13, the log-determinant of the covariance matrix is derived from the M-step as the uncertainty metric. Since the estimated covariance has to be fused with a large initial covariance, the log-determinant, or D-optimality, is guaranteed to be monotonically non-increasing during the exploration process, which is consistent with the conclusion in [29]. In addition to uncertainty criteria, it is valuable to incorporate a cost-to-go term to establish a trade-off between traveling cost and uncertainty reduction [15]. Thus, our utility is finalized as,

$$U_{\text{EM}}(X_{T:T+N}) = -\log\det(\tilde{\Sigma}_{\mathbf{x}_{T+N}}) - \sum_k \log\det(\hat{\Sigma}_{\mathbf{v}_k}) - \alpha d(X_{T:T+N}), \tag{16}$$

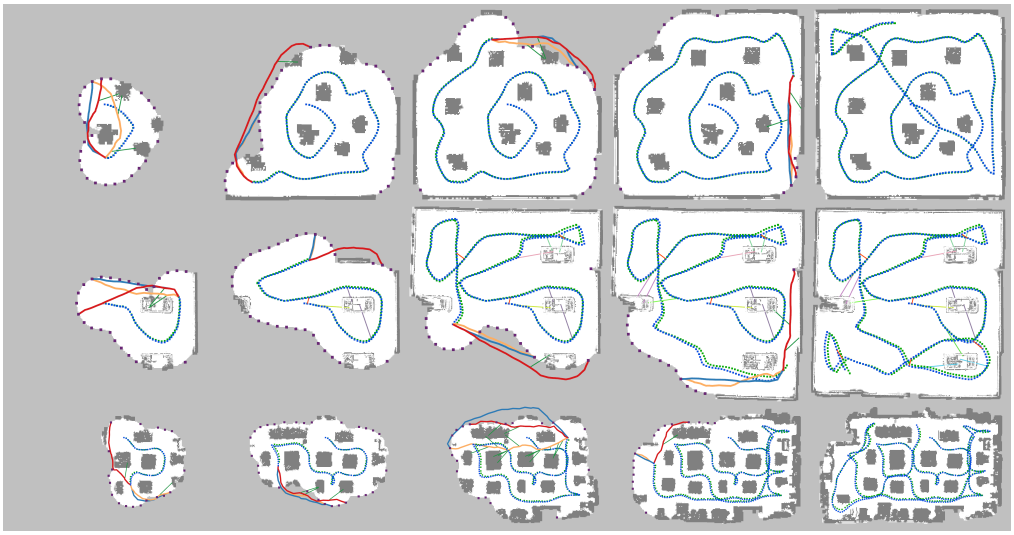


Fig. 7: Three EM exploration examples in simulated office, parking lot and real-world lab environments (from top to bottom). Trajectory estimate and ground truth are plotted with blue/green dashed lines. We also query the optimal path using different algorithms at the same moment off-line, which are shown with solid lines: EM (red), RHEM (orange) and NBV (blue). Frontiers are represented as purple squares, and predicted place recognition constraints are shown as thin green lines. All examples required 8-9 min. of real-time operation.

where α is the weight on path distance $d(X_{T:T+N})$. In our experiments, we adopt a linearly decaying weight function with respect to traveled distance, whose parameters are determined experimentally and applied consistently throughout our algorithm comparisons below. The selected path is executed immediately, but to account for deviation from the nominal trajectory during execution and inaccurate prediction after taking new measurements, the path is discarded when it's blocked by obstacles or the robot has traveled a designated distance. The exploration planning process is repeated until no frontier is detected.

F. Complexity Analysis

The generation of occupancy maps in Sec. V-A and two cost maps in Sec. V-B are trivial. Let N_p be the number of alternative paths to evaluate; N_q is the number of poses on an individual path, N_v is the number of virtual landmarks, T_d is the time to perform Dijkstra's search, T_k is the time to perform range search in k-d tree, and T_c is the time to incrementally update a factor graph and recover the block-diagonal of the covariance matrix. The time consumed to generate alternative paths and place recognition factors is N_pT_d and $N_pN_qT_k$, respectively. The computational cost of the proposed framework is dominated by belief propagation, i.e., estimating pose covariance by executing a plan. Belief propagation is naively achieved by a standard iSAM2 update and covariance recovery in GTSAM, and the total time is N_pT_c, where T_c is approximately $O(n^{2.36})$ [30]. Afterwards, covariance intersection is applied to every virtual landmark, resulting in time of N_vN_k, assuming every landmark is observed by a maximum of N_k poses.

VI. Experiments and Results

We analyze the performance of our proposed algorithm in two simulated environments where we can obtain ground truth data, and in a real environment. The robot is intended to explore the environment given only a bounding box, and terminates when no unknown area is reachable. In all experiments, we use a Clearpath Jackal UGV (Fig. 10a), which is equipped with odometry and a VLP-16 LiDAR with 3-meter range, intended to emphasize uncertainty. We implement segment-aided SLAM and EM exploration in ROS, and the simulation is built upon Gazebo.

A. Comparison

To better assess the proposed algorithm, We compare it with two variants of next-best-view approaches. Throughout the comparison, we use the same path generation method, and the utility function is regarded as the only independent variable. Consequently, our implementations differ somewhat from the documented performance of the original algorithms. We also note that the following utility functions depend entirely on the occupancy grid map, not the virtual map.

Next-best-view (NBV): The NBV planner computes accumulated gain discounted exponentially by distance from start. The gain is defined with regard to the occupancy status of the visible volume at pose $\mathbf{x}_{T+i}$ detailed in [13], and thus

$$U_{\text{NBV}}(X_{T:T+N}) = \sum_{i=1}^{N} \text{Gain}(\mathbf{x}_{T+i}) \exp(-\lambda d(X_{T:T+i})). \tag{17}$$

Uncertainty-aware receding horizon exploration and mapping (RHEM): The RHEM planner [14] improves the result in NBV by taking into account vehicle and feature uncertainty. Specifically, we apply U_{NBV} to search for a goal point, then a nested utility function is evaluated on alternative paths to the designated goal point,

$$U^{(2)}_{\text{RHEM}} = -\log\det(\tilde{\Sigma}_{\mathbf{x}_{T+N}}) - \sum_{k} \log\det(\hat{\Sigma}_{\mathbf{l}_k}), \tag{18}$$

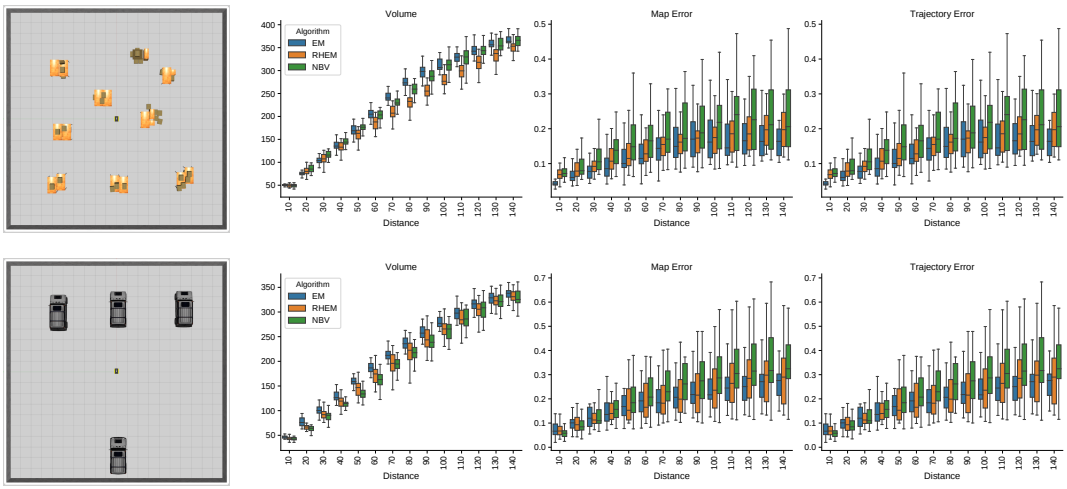


Fig. 8: Gazebo-based simulation environments explored by a simulated Jackal UGV. Results on the right show mapped volume, mapping error and localization error with respect to traveled distance over 30 trials starting from the origin of each map.

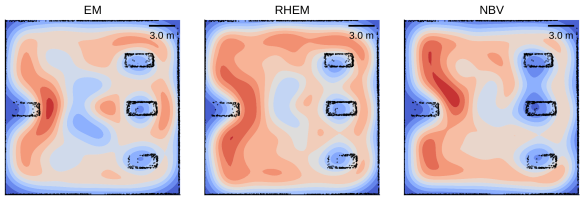


Fig. 9: Heat map of robot positions using kernel density estimation over 30 experiments in the parking lot environment. The 3-meter scale indicates the sensor range simulated.

where we switch to $\mathbf{l}_k$ in the above notation to indicate actual rather than virtual landmarks. Similarly, in a featureless environment, we resort to the same technique to acquire an upper bound of a virtual landmark's covariance, but the utility function in RHEM doesn't include the uncertainty of unknown spaces that have not yet been mapped.

B. Simulation environment

The simulated experiments were conducted in two environments ($10\,\mathrm{m} \times 10\,\mathrm{m}$), an office and parking lot, featuring different densities of objects for localization (see Fig. 8). All algorithms were executed over 30 trials, starting from the same location. We use three statistics with respect to traveled distance to analyze the performance:

1) Exploration progress is computed as the spatial volume considered as free or occupied in a 3D Octomap [31].
2) We measure the localization quality using root-mean-square error (RMSE) of the entire trajectory,

$$E_{\text{Trajectory}} = \sqrt{\frac{1}{T}\sum_{i=0}^{T} ||\hat{\mathbf{x}}_i - \mathbf{x}_i||^2}. \tag{19}$$

3) Map error is computed as the RMSE of all points transformed based on the estimated trajectory,

$$E_{\text{Map}} = \sqrt{\frac{1}{M}\sum_{k=0}^{M} ||\hat{\mathbf{p}}_k - \mathbf{p}_k||^2}. \tag{20}$$

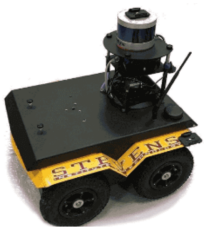
From the results presented in Fig. 8, it is evident that the EM algorithm maintains the highest exploration rate in terms of mapped volume, and more importantly it effectively modulates both trajectory and map error. The outcome can be reasoned from a few moments during EM exploration in Fig. 8. The NBV approach is solely based on unexplored volume, thus prone to erroneous state estimation. Aside from uncertainty, its utility is exponentially discounted by distance cost, which renders information obtained at a greater distance negligible. To remedy localization error, RHEM is inclined to take a detour to where the robot can reacquire previously mapped segments, but generally the improved path resides in known space. Therefore, improving one utility impairs the other one because of the separation of exploration and localization metrics. To better understand the difference in movements during exploration, we produce heatmaps of robot positions in the parking lot environment (see Fig. 9). We can see the robot was more attentive to places close to vehicles in the parking lot using the EM exploration; meanwhile, both RHEM and NBV left a large amount of footprints in the empty area, and the NBV planner had less interest in the right-half, less "informative" space.

C. Real world environment

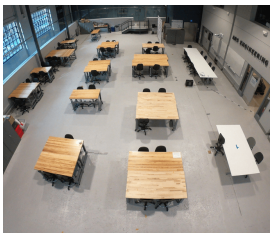
We also wish to demonstrate the applicability of our proposed algorithm to real robot platforms. The real-world experimental scenario was located at the Stevens ABS Engineering Center ($25\,\mathrm{m} \times 15\,\mathrm{m}$), similar to the simulated office, occupied with workbenches and chairs (Fig. 10b). All parameters are kept the same as in the simulated environments. Progressive instances of a representative execution trace are depicted in Fig. 7. We can observe that drift from the ground truth trajectory (obtained from LeGO-LOAM [32]) becomes more severe when the vehicle is further from the starting location. As a result, it traveled through the central region several times for better localization. Inspection of the resulting trajectory illustrates the balance between exploration and exploitation.

VII. Conclusions and Future Work

In this paper, we have adapted an EM exploration algorithm to the widely generalizable context of pose SLAM,



(a) Jackal w/ VLP-16



(b) Stevens ABS laboratory

Fig. 10: Our UGV platform and test environment.

which exploits segmentation of point clouds to enhance place recognition. The proposed framework, comprising virtual map construction and computation of the covariance upper bound, offers an advantageous capability of forecasting future actions without requiring the explicit modeling of features in SLAM. The EM algorithm exhibits superior performance in exploration rate, localization and mapping accuracy in two simulated experiments. The computational complexity is discussed in our original work [18], but throughout our experiments, it is able to achieve near real-time querying, and all experiments required less than 10 minutes of real-time operation. However, our path primitives only consist of paths to frontier locations, thus uncertainty continues increasing as exploration progresses. Improving this routing scheme is a goal for future work.

ACKNOWLEDGEMENTS

This research was supported in part by the National Science Foundation, grants IIS-1652064 and IIS-1723996, and by a grant from Schlumberger Technology Corporation.

REFERENCES

[1] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid and J.J. Leonard, "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE Transactions on Robotics*, vol. 32(6), pp. 1309-1332, 2016.

[2] H.J.S. Feder, J.J. Leonard, and C.M. Smith, "Adaptive mobile robot navigation and mapping," *The International Journal of Robotics Research*, vol. 18(7), pp. 650-668, 1999.

[3] R. Martinez-Cantin, N. Freitas, A. Doucet, and J.A. Castellanos, "Active policy learning for robot planning and exploration under uncertainty," *Robotics: Science and Systems*, vol. 3, pp. 334-341, 2007.

[4] R. Sim, and N. Roy, "Global A-Optimal robot exploration in SLAM," *IEEE International Conference on Robotics and Automation*, pp. 661-666, 2005.

[5] V. Indelman, L. Carlone, and F. Dellaert, "Planning in the continuous domain: A generalized belief space approach for autonomous navigation in unknown environments," *The International Journal of Robotics Research*, vol. 34(7), pp. 849-882, 2015.

[6] B. Yamauchi, "A frontier-based approach for autonomous exploration," *IEEE International Symposium on Computational Intelligence in Robotics and Automation*, pp. 146-151, 1997.

[7] B.J. Julian, S. Karaman and D. Rus, "On Mutual Information-Based Control of Range Sensing Robots for Mapping Applications," *The International Journal of Robotics Research*, vol. 33(10), pp. 1375-1392, 2014.

[8] S. Bai, J. Wang, F. Chen, and B. Englot, "Information-Theoretic Exploration with Bayesian Optimization," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1816-1822, 2016.

[9] B. Charrow, S. Liu, V. Kumar and N. Michael, "Information-theoretic mapping using Cauchy–Schwarz quadratic mutual information," *IEEE International Conference on Robotics and Automation*, pp. 4791-4798, 2015.

[10] M.G. Jadidi, J.V. Miro and G. Dissanayake, "Gaussian processes autonomous mapping and exploration for range-sensing mobile robots," *Autonomous Robots*, vol. 42(2), pp. 273-290, 2018.

[11] F. Bourgault, A. A. Makarenko, S. B. Williams, B. Grocholsky and H. F. Durrant-Whyte, "Information based adaptive robotic exploration," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 540-545, 2002.

[12] A.A. Makarenko, S.B. Williams, F. Bourgault and H.F. Durrant-Whyte, "An experiment in integrated exploration," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 534-539, 2002.

[13] A. Bircher, M. Kamel, K. Alexis, H. Oleynikova, and R. Siegwart, "Receding horizon next-best-view planner for 3D exploration," *IEEE International Conference on Robotics and Automation*, pp. 1462-1468, 2016.

[14] C. Papachristos, S. Khattak, and K. Alexis, "Uncertainty-aware receding horizon exploration and mapping using aerial robots," *IEEE International Conference on Robotics and Automation*, pp. 4568-4575, 2017.

[15] C. Stachniss, G. Grisetti and W. Burgard, "Information gain-based exploration using Rao–Blackwellized particle filters," *Robotics: Science and Systems*, pp. 65-72, 2005.

[16] R. Valencia, J. V. Miro, G. Dissanayake and J. Andrade-Cetto, "Active pose SLAM,", *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1885-1891, 2012.

[17] J. Vallve and J. Andrade-Cetto, "Active pose SLAM with RRT*," *IEEE International Conference on Robotics and Automation*, pp. 2167-2173, 2015.

[18] J. Wang and B. Englot, "Autonomous exploration with Expectation-Maximization," *International Symposium on Robotics Research*, 2017.

[19] R. Dubé, A. Gawel, H. Sommer, J. Nieto, R. Siegwart and C. Cadena, "An online multi-robot slam system for 3d lidars," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1004-1011, 2017.

[20] R. Dubé, D. Dugas, E. Stumm, J. Nieto, R. Siegwart and C. Cadena, "SegMatch: Segment based place recognition in 3D point clouds," *IEEE International Conference on Robotics and Automation*, pp. 5266-5272, 2017.

[21] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. J. Leonard and F. Dellaert, "iSAM2: Incremental smoothing and mapping using the Bayes tree," *The International Journal of Robotics Research*, vol. 31(2), pp. 216-235, 2012.

[22] R. Dubé, A. Cramariuc, D. Dugas, J. Nieto, R. Siegwart and C. Cadena, "SegMap: 3D Segment Mapping using Data-Driven Descriptors," *Robotics: Science and Systems*, 2018.

[23] F. Dellaert and M. Kaess, "Square Root SAM: Simultaneous localization and mapping via square root information smoothing," *The International Journal of Robotics Research*, vol. 25(12), pp. 1181-1203, 2006.

[24] G. Celeux and G. Govaert, "A classification EM algorithm for clustering and two stochastic versions," *Computational statistics & Data analysis,* vol. 14(3), pp.315-332, 1992.

[25] H. Carrillo, I. Reid and J. A. Castellanos, "On the comparison of uncertainty criteria for active SLAM," *IEEE International Conference on Robotics and Automation*, pp. 2080-2087, 2012.

[26] L. Chen, P. O. Arambel and R. K. Mehra, "Estimation under unknown correlation: covariance intersection revisited," *IEEE Transactions on Automatic Control*, vol. 47(11), pp. 1879-1882, 2002.

[27] J. Wang and B. Englot, "Robust Exploration with Multiple Hypothesis Data Association," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3537-3544, 2018.

[28] R.T. Marler and J.S. Arora, "Survey of multi-objective optimization methods for engineering," *Structural and multidisciplinary optimization*, vol. 26(6), pp. 369-395, 2004.

[29] H. Carrillo, Y. Latif, M.L. Rodriguez-Arevalo, J. Neira and J.A. Castellanos, "On the monotonicity of optimality criteria during exploration in active SLAM," *IEEE International Conference on Robotics and Automation*, pp. 1476-1483, 2015.

[30] V. Ila, L. Polok, M. Solony, P. Smrz and P. Zemcik, "Fast covariance recovery in incremental nonlinear least square solvers," *IEEE International Conference on Robotics and Automation*, pp. 4636-4643, 2015.

[31] A. Hornung, K.M. Wurm, M. Bennewitz, C. Stachniss and W. Burgard, "OctoMap: An Efficient Probabilistic 3D Mapping Framework Based on Octrees," *Autonomous Robots*, vol. 34(3), pp. 189-206, 2013.

[32] T. Shan and B. Englot, "LeGO-LOAM: Lightweight and Ground-Optimized Lidar Odometry and Mapping on Variable Terrain," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4758-4765, 2018.