Two-Stage Focused Inference for Resource-Constrained Minimal Collision Navigation

Beipeng Mu, *Member, IEEE*, Liam Paull, *Member, IEEE*, Ali-Akbar Agha-Mohammadi, *Member, IEEE*, John J. Leonard, *Fellow, IEEE*, and Jonathan P. How, *Senior Member, IEEE*

***Abstract*—The operation of mobile robots in unknown environments typically requires building maps during exploration. As the exploration time and environment size increase, the amount of data collected and the number of variables required to represent these maps both grow, which is problematic since all real robots have finite resources. The solution proposed in this paper is to only retain the variables and measurements that are most important to achieve the robot's task. The variable and measurement selection approach is demonstrated on the task of navigation with a low risk of collision. Our approach has two stages: first, a subset of the variables is selected that is most useful for minimizing the uncertainty of navigation (termed the "focused variables"). And second, a task-agnostic method is used to select a subset of the measurements that maximizes the information over these focused variables ("focused inference"). Detailed simulations and hardware experiments show that the two-stage approach constrains the number of variables and measurements. It can generate much sparser maps than existing approaches in the literature, while still achieving a better task performance—in this case *(fewer collisions)*. An incremental and iterative approach is further presented, in which the two-stage procedure is performed on subsets of the data, and thus, avoids the necessity of performing a resource-intensive batch selection on large datasets.**

***Index Terms*—Indoor navigation, path planning, simultaneous localization and mapping.**

I. Introduction

ONE of the core enabling capabilities for mobile robots operating in uncertain and GPS-denied environments is simultaneous localization and mapping (SLAM). The built map will subsequently be used to perform tasks. Many of the robotic missions of interest (e.g., autonomous cars, marine robots) often require high-dimensional models to represent the robot poses, landmarks, and obstacles. Graphical models are a powerful tool for modeling high-dimension problems because they can explicitly utilize conditional independencies between variables [1]–[3]. However, when the robot travels longer distance, the variables required to represent robot poses grows and memory demand growths. And when the robot stays longer in the environment, it obtains more and more measurements of the landmarks and the computation demand on processing these measurements grows. Naively applying these methods can result in an unbounded growth in memory and computational requirements.

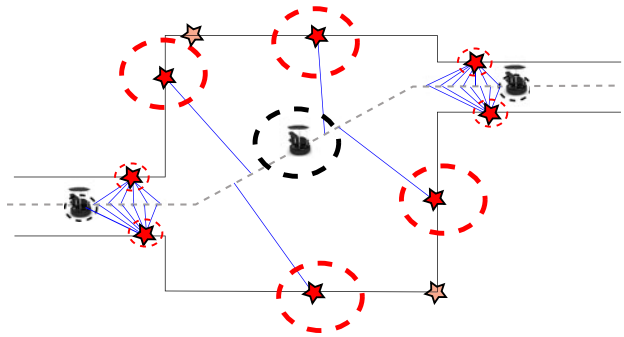


Fig. 1. Resource constrained collision-free navigation: It is more important to localize landmarks in narrow hallways as compared to those in open areas. As a result, along the trajectory of the robot (gray dashed line), far fewer measurements (blue) are required for landmarks (red stars) in the open area as compared with the narrow hallway. (Black and red dashed ellipses represent uncertainty in robot position and landmarks position, respectively.)

On the other hand, the required fidelity and choice of map representation are, in general, domain specific, and furthermore may not even be constant across a given application. Consider the following tasks:

1) a robot is navigating through an unknown environment consisting of both open areas and narrow hallways;
2) an autonomous car is operating on a road network consisting of highways and rarely used local roads;
3) a marine robot is localizing a set of underwater mines amongst clutter.

In these cases, it is important to prioritize which landmarks are maintained in the map. As illustrated in Fig. 1, in an indoor robot navigation scenario, the space consists of both narrow hallways and open areas. It is more important to have accurate estimates of landmark locations in tight corridors, therefore the robot has lower uncertainty in its poses and less chance of collisions compared with open areas. Similarly on the road network, landmarks on highly traveled roads are more useful [4] and in the underwater scenario it is more important that the robot localizes the mines as opposed to the clutter. Consequently, the robot can

Manuscript received May 6, 2016; revised August 23, 2016 and September 22, 2016; accepted October 5, 2016. This paper was recommended for publication by Associate Editor J. M. Porta and Editor T. Murphey upon evaluation of the reviewers' comments. This work was supported in part by ARO MURI Grant W911NF-11-1-0391, in part by ONR Grant N00014-11-1-0688, and in part by NSF Award IIS-1318392.

B. Mu, A. A. Agha-Mohammadi, and J. P. How are with the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: mubp@mit.edu; aliagha@mit.edu; jhow@mit.edu).

L. Paull and J. J. Leonard are with the Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: lpaull@mit.edu; jleonard@mit.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TRO.2016.2623344

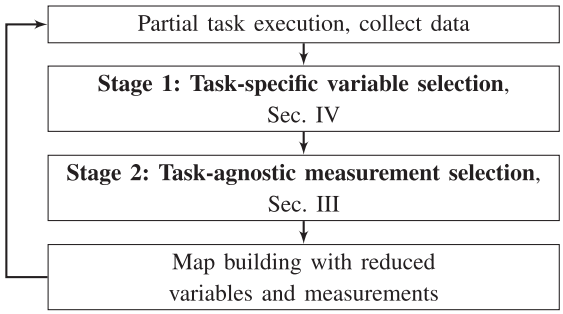


Fig. 2. Two-stage focused inference.

save resources, such as memory and computation, by focusing the mapping operation to more explicitly support the task.

Recent work on map reduction has focused on minimizing the impact on the overall quality of the map and robot trajectory. For example, using some criteria to discard incoming measurements [5], [6], or marginalizing out nodes followed by a sparsification procedure to maintain efficiency [7]–[9]. These works do not consider the robot task performance achieved with the resulting maps, nor do they answer the variable selection question of what are the right nodes to retain.

This paper first presents a more flexible batch-selection framework that supports task-specific prioritization. The common task of mobile robot navigation is used in this study. By focusing on the important parts of the map, our approach reduces the resource requirements without significantly impacting the task performance (navigation).

Batch-selection operates on the full dataset. In a typical scenario, as explored space grows, the number of variables used to represent the map grows, and as exploring time grows, the number of measurements grows. Both of these scalability issues will make the batch selection slow or infeasible for robots, which necessarily have finite resources. We further propose an incremental selection process to operate on sequences of streaming data. As new data arrives, we perform the two-stage selection based on the result from previous selection, thus reducing the resource requirements for processing a single-batch dataset.

The workflow for the general case is shown in Fig. 2. The robot first explores to gather data. The next step is to select the focused variables that are deemed important for subsequent navigation tasks. Section IV discusses the specific case of selecting landmarks to support collision-free navigation (the "focused variables"). The third step is to select the subset of measurements that are most useful for estimating the focused variables, as described in Section III. The last step is to build a map using the reduced set of variables and measurements. The robot continues in this loop until the task is complete. In the incremental approach, the robot will obtain new data each time it performs the task execution. The reduced map produced from previous data is taken as an input to the two-stage variable and measurement reduction algorithm with new collected data to further update the map.

In summary, we claim the following contributions:

1) A generic framework for measurement selection in the case that some variables are deemed higher priority than others.
2) A method for landmark (focused variable) selection to support the specific task of collision-free navigation.
3) An incremental algorithm for selecting variables and measurements from new data to reduce processing and storage requirements.
4) Simulations and hardware experiments which demonstrate that the approach reduces collisions given fixed resources.

A previous version of this paper was presented in [10]. This paper additionally provides a detailed discussion of navigation with uncertain landmark locations (see Section V). The framework has also been upgraded to enable incremental operations, where robots can sequentially select variables and measurements to expand existing maps (see Section VI). Finally, much more extensive experiments are conducted to test the variable/measurement selection as well as the incremental approach (see Section VII-B).

II. Related Work

This study is at the intersection of a number of robotics subdomains, so the following reviews some of the key recent results in the areas of map reduction, landmark selection and its application to autonomous navigation, and planning with uncertainty.

A. Map Reduction

The map reduction literature typically assumes the variables to be retained are provided as input, and focuses only on measurement reduction to make the map sparse. For example, the original filtering-based approaches to SLAM marginalize old poses at every time step, and significantly delay inference on large scale problems. The sparse extended information filter (SEIF) [11] breaks weak links in the graph to achieve sparsity and speed up SLAM solutions. A consistent alternative, the exactly SEIF, selectively discards data during the measurement update step [12]. A less conservative, but still consistent approach, formulates the sparsification process as a constrained convex optimization to minimize the Kullback–Leibler divergence (KLD) between the sparse and true estimates subject to the consistency and sparsity constraints [13].

Graph-based optimization approaches [14] are also widely used for solving SLAM problems. These methods provide a naturally sparse representation of the SLAM problem that can be solved efficiently [15]. Nevertheless, these methods do not scale constantly with time and space, and ultimately require some form of graph reduction to enable prolonged operation.

Given the variables to be removed, marginalization induces a fully connected subgraph over the Markov blanket of the marginalized variable. A convex optimization can be employed on the subgraph for sparsification. Similar to the method introduced in [13], the KLD between the dense subgraph and a sparse approximation is minimized subject to a consistency constraint. Carlevaris-Bianco *et al.* [7] present a method called generic linear constraints (GLC) that sparsifies the dense subgraph using a Chow–Liu tree (CLT) approximation. Alternately, sparsity can be enforced through an ℓ_1-regularization term in the KLD minimization [8], which has the advantage that it does not impose a specific graph structure on the

sparse approximation (e.g., a CLT). Mazuran *et al.* [9] recently improved upon previous methods by allowing nonlinear measurements to approximate the dense subgraph, and then formulating the KLD minimization with respect to the *measurement*, rather than the *state*, information matrix. These graph reduction techniques are not concerned with selecting the nodes to be removed from the graph. A further issue is that performance can degrade if the wrong landmarks are removed through marginalization since they are no longer available for subsequent loop-closures.

Another approach is not to remove any variables, but instead discard measurements before they are processed by the SLAM optimizer. Kretzschmar and Stachniss [6] propose a pose-graph compression where laser scans are selectively removed by utilizing an approximate marginalization based on a CLT.

The method proposed in this paper selects *both* variables and measurements. The only other known works to consider both are [16] and [5], both of which are task agnostic. In [16], a graphical method is employed to select the variables to remove which will result in the minimum KLD after the subsequent sparsification. In [5], measurements and poses are selectively added as new data arrives based on an information-theoretic metric, which can be efficiently computed. Also note that, in [5], there is no limit on the number of measurements or poses that can be added to the state space, only their rate of growth will be limited. Here, we choose to retain a fixed number of variables that will maximally impact the robot's ability to perform a given task, and then, select a fixed number of measurements to preferentially estimate these more important variables.

B. Landmark Selection

Techniques to perform the subselection of previously mapped landmarks to support localization have been proposed to accomplish a number of different objectives for example, actively placing sensors to maximize coverage [17] or reducing navigation uncertainty [18]. A standard approach in vision-based systems is to downsample landmarks based on a measure of visual saliency in an effort to improve the detection of loop closures. Specific applications include map compression [19], active gaze control [20], area coverage [21], and lifelong operation of service robots [22]. Closely related to our motivation is the application of resource-constrained inference. For a localization and mapping objective, previously proposed approaches include uniform landmark selection [23] (henceforth will be referred to as "downsampling") and entropy-based landmark selection [24].

Several researchers have considered selection of landmarks to support navigation. Strasdat *et al.* [25] proposed a reinforcement-learning-based landmark selection policy to minimize the robot position error at the goal. Lerner *et al.* [26] considered single-camera frame-based landmark selection in terms of a "severity function." And Sala *et al.* [27] proposed to choose a minimal set of landmarks such that at least k are viewable from every point in the configuration space. Note, however, that none of these previous works consider the obstacles, obstacle uncertainty, or probability of collision in the landmark selection process. Our method, in contrast, chooses landmarks in a SLAM setting without a prior map, with an explicit goal of minimizing the probability of collision during navigation by inherently accounting for metric properties of the map, such as constrictions and tight corridors. This technique also limits the resources required to map the landmarks simultaneously.

C. Planning Under Uncertainty

This paper specifically considers navigation under map and pose uncertainty as the motivating application for the focused inference framework. A vast amount of literature addresses the problem of finding collision free paths in the presence of robot pose uncertainty assuming the landmark map is given. A standard measure in finding a safe path is to define a probability of collision with an obstacle, and the resulting path can be chosen that balances optimality and risk [28], [29]. In [30], an optimal path is found subject to a maximum allowable probability of collision (typically called a "chance constraint"). Measurement uncertainty is taken into account in [31] and[32] to compute a more accurate estimate of the robot pose and collision probabilities. The path is planned in advance assuming accurate stochastic models for motion dynamics and sensor measurements. Finally, there is a small class of planning algorithms that consider motion, measurement, and map uncertainty (e.g., [33]), but these approaches are mainly limited to problems with small discrete state, action, or measurement spaces.

The most relevant work to our present approach is by Lambert and Le Fort-Piat [34] who use a set-bounding technique to ensure that the 3σ ellipse of the robot's pose estimate never comes into contact with an obstacle. This paper is one of the few to explicitly consider the pose, control, and map uncertainty. The known shortcoming of such an approach is that it tends to be overly conservative. **Particularly, in the case of highly cluttered environments or tight corridors, the algorithms will fail to produce a feasible solution. This paper presents a mapping scheme that is specifically designed to be utilized in a probabilistic navigation module. We provide a rigorous treatment of the coupling between trajectory and landmark map uncertainties, which is achieved within memory and computational constraints, making our approach applicable to low-cost robots operating with limited sensing in realistic environments.

III. Measurement Selection for Inference on Focused Variables

We begin by assuming that the selection of focused variables has already been performed, and proceed to formulate the problem of measurement selection to support inference over these focused variables. We keep the formulation as general as possible at this measurement-selection stage. The following sections will discuss task-specific focused variable selection and task execution.

In general, there are two ways to sparsify measurements. The first one is *forward selection* where the map is initially empty and measurements are added when they are deemed important [5]. The second way is *backward map reduction* in which the map is initialized with an optimized full graph and measurements are removed [13]. However, an optimized graph with the full set of variables and measurements would be difficult to compute for

a real robot with limited onboard computation. Furthermore, the number of measurements to retain is typically very small compared to the total number of measurements that are available, so the backward reduction is relatively inefficient. If each step only adds or removes one measurement, then backwards map reduction will involve many more operations than that of forward selection. As a result, a forward selection approach is used in this paper.

A. Problem Formulation

Denote $X = \{X_1, \ldots, X_N\}$ as a set of hidden random variables that are (partially) observable through measurements $z = \{z_1, \ldots, z_K\}$ which are collected in the initial data collection phase, where K can be very large. Then, in its most general form, the measurement selection problem can be posed as follows.

Problem 1: Unfocused Measurement Selection: Select the subset of measurements $z^R = \{z_1^R, \ldots, z_{K^R}^R\} \subset z$, such that some information metric $f(\cdot)$ over the hidden variables X is maximized, subject to some cost function $g(\cdot)$ constraint on the measurement set

$$\begin{aligned} \max_{z^R \subset z} \quad & f(X; z^R) \\ \text{s.t.} \quad & g(z^R) \leq c. \end{aligned} \tag{1}$$

Commonly used information metrics include entropy, mutual information, and KLD. In this paper, the cost function used is the cardinality of the set $g(z^R) = K^R$. In practice, the number of measurements often relates to the model complexity, and thus the computational cost. Therefore, the resource budget c could be selected based on computational resource available.

Here, instead of Problem 1, we have a set of "focused" variables, $\widetilde{X} = \{\widetilde{X}_1, \ldots \widetilde{X}_{\widetilde{N}}\}$, which is a compact representation of the variables in X, with $\widetilde{N} \ll N$.[1] To maintain generality, we represent the mapping from the unfocused variables to the focused variables by a *prioritization function*.

Definition 1: Prioritization Function: The function $w : \mathcal{R}^N \rightarrow \mathcal{R}^{\widetilde{N}}$ that maps the full set of variables onto the set of focused variables.

For example, in the degenerate case of variable selection, $\widetilde{X} = w(X) = WX$, where W is an $\widetilde{N} \times N$ matrix with a single 1 in each row. However, the formulation allows for more complex mappings from the full set to the focused set. Notice that the prioritization function gives a mapping relationship between original and focused variables, but does not marginalize original variables: all the original variables X are still maintained in the model at this stage.

The problem of focused measurement selection consists of choosing the best subset of the full measurement set z to optimally estimate the focused variables.

Problem 2: Focused Measurement Selection: Select the subset of measurements $z^R = \{z_1^R, \ldots, z_{K^R}^R\} \subset z$, such that some information metric $f(\cdot)$ over the **focused** variables $\widetilde{X}$ is maximized, subject to some cost function $g(\cdot)$ constraint on the measurement set

$$\begin{aligned} \max_{z^R \subset z} \quad & f(\widetilde{X}; z^R) \\ \text{s.t.} \quad & |z^R| \leq c. \end{aligned} \tag{2}$$

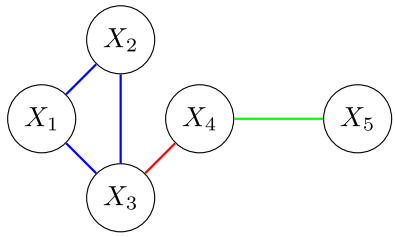


Fig. 3. *Graphical model:* Edge colors denote factors: blue $\psi_{123}(X_1, X_2, X_3)$, red $\psi_{34}(X_3, X_4)$, and green $\psi_{45}(X_4, X_5)$.

Graphical models, such as factor graphs in Fig. 3 are a compact way of representing dependencies between variables [1]. For a factor graph, the joint posterior can be expressed as a product of factors, $\psi_c(x_{\{c\}})$

$$p(x|z^R) \propto \prod_{c \in \mathcal{C}} \psi_c(x_{\{c\}}) \tag{3}$$

where $\mathcal{C}$ is the set of all factors, $x_{\{c\}}$ are the variables in factor c. Because each factor $\psi_c(x_{\{c\}})$ is strictly positive, $p(X = x|z^R)$ can be written equivalently in logistic form

$$p(x|z^R) \propto \exp\left\{ \sum_{c \in \mathcal{C}} \phi_c(x_{\{c\}}) \right\} \tag{4}$$

where $\phi_c(x_{\{c\}}) = \log \psi_c(x_{\{c\}})$.

The reduction from the full set of variables to the focused variables is achieved by mapping the posterior through the prioritization function $\tilde{x} = w(x)$ to produce a new posterior over the focused variables

$$p(\tilde{x}|z^R) \propto \exp\left\{ \sum_{c \in \widetilde{\mathcal{C}}} \tilde{\phi}_c\left(\tilde{x}_{\{c\}}\right) \right\}, \quad \tilde{x} = w(x) \tag{5}$$

where $\widetilde{\mathcal{C}}$ is the new (smaller) set of factors over $\tilde{X}$, and $\tilde{\phi}_c$ are the resulting factors.

If we define the function $f(\cdot)$ in (2) to be the Shannon entropy of the conditional distribution

$$H(\widetilde{X}|z^R) = \mathbb{E}_{\widetilde{X}|z^R}\left[-\log p(\tilde{x}|z^R)\right] \tag{6}$$

where $\mathbb{E}$ is the expectation operator, then we obtain the resulting equation for the entropy of the focused variables, $\widetilde{X}$ conditional on the subset of measurements z^R as

$$f(\widetilde{X}; z^R) = H(\widetilde{X}|z^R) = \mathbb{E}_{\widetilde{X}|z^R}\left[-\sum_{c \in \widetilde{\mathcal{C}}} \tilde{\phi}_c(\tilde{x}_{\{c\}}) \right] + C \tag{7}$$

where C is a constant. A more concrete example will be given in next Section III-B.

Note that computing the transformation from ϕ to $\tilde{\phi}$ can be hard in general. Furthermore, the graph over $\widetilde{X}$ will be more dense than the graph of X, and computing $H(\widetilde{X}|z^R)$ can be computationally expensive. However, it will be shown

[1] The "$\widetilde{\cdot}$" notation is used in this paper to refer to the set of focused variables.

that $H(\widetilde{X}|z^R)$ can be computed in closed-form given two assumptions:

Assumption 1: The graphical model can be approximated as a Gaussian distribution.

Assumption 2: The prioritization function $w(\cdot)$ is an affine transformation.

Assumption 1 is, in fact, less limiting than the standard additive Gaussian noise assumption in the SLAM literature, since even in the case of robust cost functions, we can always approximate the posterior as a Gaussian distribution using the Laplacian approximation [35]. Assumption 2 essentially requires the focused variables to be linear combinations of the original variables, which is still more general than other variable selection methods [23], [24] which restrict the set of focused variables to be a strict subset of the original set. For example, given corners/edges of an object, a focused variable could be the center of the object, which is a linear combination of the corners/edges.

B. Gaussian Approximation

We begin by applying the standard method [36] of approximating a posterior over the unfocused variables (4) using a second-order Taylor series expansion of the factors ϕ_c at some initial guess x^* and denote the approximated factors as $\hat{\phi}_c$[2]

$$\begin{aligned} p(x|z^R) \approx\ \hat{p}(x|z^R) &\propto \exp\left\{\sum_{c\in\mathcal{C}} \hat{\phi}_c(x_{\{c\}})\right\} \\ &= \exp\left\{\sum_{c\in\mathcal{C}} \phi_c(x^*_{\{c\}}) + (x-x^*)^T \sum_{c\in\mathcal{C}} \frac{\partial}{\partial x}\phi_c(x^*_{\{c\}}) \right. \\ &\quad \left. + \frac{1}{2}(x-x^*)^T\left(\sum_{c\in\mathcal{C}} \frac{\partial^2}{\partial x^2}\phi_c(x^*_{\{c\}})\right)(x-x^*)\right\}. \quad (8) \end{aligned}$$

Note that the exponential component in (8) is quadratic in x, therefore the approximation is a Gaussian distribution with information matrix, Λ_{z^R} given by the Hessian

$$\hat{p}(x|z^R) = \mathcal{N}^{-1}(\zeta, \Lambda_{z^R}),\ \Lambda_{z^R} = \sum_{c\in\mathcal{C}} -\frac{\partial^2}{\partial x^2}\phi_c(x^*_{\{c\}}) \quad (9)$$

which can be further decomposed in the case of single and pairwise factors as

$$\Lambda_{z^R} = \frac{1}{2}\sum_{i,j=1..N} x_i^T \Lambda_{i,j} x_j \quad (10)$$

where $\Lambda_{i,j} \neq 0$ only if $i = j$ or X_i and X_j are connected in the graph. We write Λ_{z^R} with subscript z^R to explicitly represent that the information is dependent on the choice of selected measurements which will impact the structure of the graph and the resulting factors. One point of note is that Λ_{z^R} is inherently dependent on the linearization point chosen x^*.

C. Affine Prioritization Function

In Def. 1, we defined the prioritization function $w(\cdot)$, which is a task-specific function that maps the set of all variables onto the set of focused variables. In this section, we impose the restriction that this function is affine in order to provide a closed-form solution of getting from (4) to (5):

$$\widetilde{X} = w(X) = WX \quad (11)$$

where $W \in \mathcal{R}^{\widetilde{N}\times N}$. For example, in Fig. 4 we have

$$\begin{bmatrix}\widetilde{X}_1\\ \widetilde{X}_2\\ \widetilde{X}_3\end{bmatrix} = \underbrace{\begin{bmatrix}0 & 1 & 0 & 0 & 0\\ 0 & 0 & 1/3 & 2/3 & 0\\ 0 & 0 & 0 & 0 & 1\end{bmatrix}}_{W} \begin{bmatrix}X_1\\ X_2\\ X_3\\ X_4\\ X_5\end{bmatrix}. \quad (12)$$

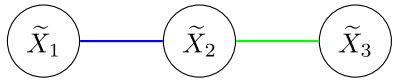


Fig. 4. Transformed graphical model. New variables $\widetilde{X}_1 = X_2$, $\widetilde{X}_2 = \frac{1}{3}X_3 + \frac{2}{3}X_4$, $\widetilde{X}_3 = X_5$. New factors: blue $\widetilde{\psi}_{12}(\widetilde{X}_1, \widetilde{X}_2)$, green $\widetilde{\psi}_{2,3}(\widetilde{X}_2, \widetilde{X}_3)$.

This restriction on the prioritization to be affine guarantees that the posterior over the focused variables will still be Gaussian: $\hat{p}(\tilde{x}|z^R) = \mathcal{N}^{-1}(\tilde{\zeta}, \Lambda_{z^R})$ [3]. Furthermore, we can easily write an expression for the information matrix

$$\Lambda_{z^R} = (W\Lambda_{z^R}^{-1}W^T)^{-1} \quad (13)$$

as a result, the approximate entropy of the focused variables given the selected measurements can be written in closed form as

$$\hat{H}(\widetilde{X}|z^R) = -\frac{1}{2}\log\left|\widetilde{\Lambda}_{z^R}\right| + C = \frac{1}{2}\log\left|W\Lambda_{z^R}^{-1}W^T\right| + C. \quad (14)$$

We finish by restating Problem 2 based on the Gaussian approximation and restriction to affine prioritization functions.

Problem 3 (Approximate Focused Measurement Selection): Select the subset of measurements $z^R = \{z_1^R, \ldots, z_{K^R}^R\} \subset z$, such that approximate entropic information over the **focused** hidden variables $\widetilde{X}$ is maximized, subject to the same constraint as (2) as

$$\begin{aligned} \max_{z^R\subset z}\ & -\log\left|W\Lambda_{z^R}^{-1}W^T\right| \\ \text{s.t.}\quad & |z^R| \leq c. \end{aligned}$$

D. Efficiently Solving Problem 3

Notice Problem 3 optimizes an objective function that is defined on a set. Such a set function is discrete and combinatorial, and computing the optimal solution is typically intractable on SLAM-size problems. We use a greedy selection procedure that maximizes the incremental information gain on the next measurement.

Each new measurement z_k^R added to the set will introduce a new factor, $\phi_k(x^*_{\{k\}})$, into the joint posterior, where the set of variables $X_{\{k\}}$ are the ones affected by measurement z_k^R. We denote the intermediate set of $k \leq K^R$ measurements that have already been selected as $z^{R_k} = \{z_1^R, \ldots, z_k^R\}$.

The approximate entropy reduction (or information gain) over the focused variables brought about by adding a new measure-

[2]The ˆ notation is used throughout to refer to the Gaussian approximation.

ment z_k^R that we want to maximize is

$$\Delta\hat{H}(\widetilde{X}|z_k^R) = \hat{H}(\widetilde{X}|z^{R_{k-1}}) - \hat{H}(\widetilde{X}|z^{R_k}). \tag{15}$$

In the following theorem, we show that this quantity can be efficiently computed.

Theorem 1: The approximate reduction in entropy over the focused variables brought about by introducing the new measurement z_k^R will be

$$\Delta\hat{H}(\widetilde{X}|z_k^R) = -\frac{1}{2}\log\left|I - (I + J_k^T\Lambda_{z^{R_{k-1}}}^{-1}J_k)^{-1} \times L_k^T\widetilde{\Lambda}_{z^{R_{k-1}}}L_k\right| \tag{16}$$

where $L_k \triangleq W\Lambda_{z^{R_{k-1}}}^{-1}J_k$, J_k is the measurement covariance weighted stacked Jacobian [8], and W is the affine prioritization function.

Proof: We proceed similarly to [5], but with the added complication that there is a transformation from the unfocused to the focused variables ($\widetilde{X} = WX$) required to evaluate (15).

From (9), the information matrix after the introduction of z_k^R will be

$$\Lambda_{z^{R_k}} = \Lambda_{z^{R_{k-1}}} - \frac{\partial^2\phi_k(x_{\{k\}}^*)}{\partial x^2} = \Lambda_{z^{R_{k-1}}} + J_kJ_k^T \tag{17}$$

where $J_k \in \mathcal{R}^{N\times|x_{\{k\}}|}$, with $|x_{\{k\}}|$ the number of variables in clique k and only blocks corresponding to the variables in the clique $x_{\{k\}}$ being nonzero. The new entropy after adding measurement z_k^R can be evaluated as

$$\hat{H}(\widetilde{X}|z_k^R) = \frac{1}{2}\log\left|W(\Lambda_{z^{R_k}})^{-1}W^T\right|. \tag{18}$$

Then, the reduction in entropy, $\Delta\hat{H}(\widetilde{X}|z_k^R)$, is

$$-\frac{1}{2}\log\left|W\Lambda_{z^{R_k}}^{-1}W^T\right| + \frac{1}{2}\log\left|W\Lambda_{z^{R_{k-1}}}^{-1}W^T\right| = -\frac{1}{2}\log\frac{\left|W(\Lambda_{z^{R_{k-1}}} + J_kJ_k^T)^{-1}W^T\right|}{\left|\left(W\Lambda_{z^{R_{k-1}}}^{-1}W^T\right)\right|}. \tag{19}$$

By applying the matrix inversion lemma

$$(A + BB^T)^{-1} = A^{-1} - A^{-1}B(I + B^TA^{-1}B)^{-1}B^TA^{-1}$$

and the determinant property $|I + AB| = |I + BA|$, (19) reduces to

$$-\frac{1}{2}\log\left|I - (I + J_k^T\Lambda_{z^{R_{k-1}}}^{-1}J_k)^{-1}L_k^T\Lambda_{z^{R_{k-1}}}L_k\right|$$

which is the required result. ■

1) Computational Complexity of (16): Similar to [5], we have avoided the need to compute the entropy over the entire variable set. However, unlike in [5] where the calculation of information gain scales only with the size of the measurements, we have a slightly more complicated scenario because of the prioritization transformation W. Upon further inspection of (16), calculating $J_k^T\Lambda_{z^{R_{k-1}}}^{-1}J_k$ has a computational complexity of $\mathcal{O}(|x_{\{k\}}|^4)$ (similar to [5]), where $|x_{\{k\}}|$ is the cardinality of $x_{\{k\}}$. In addition, note that calculating the ith element of L_k (which is computed as $L_k^i = W^i\Lambda_{z^{R_{k-1}}}^{-1}J_k$, where W^i is ith row of W) requires us to check the submatrix of $\Lambda_{z^{R_{k-1}}}^{-1}$ that corresponds to the nonzero elements of W^i *and* J_k. Therefore, the overall complexity of computing L_k is $\mathcal{O}(||W||_0|x_{\{k\}}| + |x_{\{k\}}|^4)$, where $||W||_0$ is the number of nonzero elements in the prioritization transformation. Typically the clique size, $|x_{\{k\}}| \ll N$, and since $\widetilde{N} \ll N$ we should expect that $||W||_0 \ll N$, therefore the overall complexity of computing L_k is much less than the problem size N. Algorithm 1 summarizes the measurement selection approach.

Algorithm 1: Measurement Selection for Prioritized Landmarks.

Input: Initial information matrix $\Lambda_{z^{R_0}}$, focused variables $\widetilde{X}$, all measurements z, budget c
Output: measurements z^R

```
k ← 0, z^{R_0} ← ∅
while |z^{R_k}| < c do
    k ← k + 1
    z_k^* = arg max_{z_k^R ∈ z\z^{R_{k-1}}} ΔĤ(X̃|z_k^R)
    z^{R_k} = z^{R_{k-1}} ∪ {z_k^*}
    Λ_{z^{R_k}} = Λ_{z^{R_{k-1}}} + J_k J_k^T
    Λ̃_{z^{R_k}} = (W Λ^{-1}_{z^{R_k}} W^T)^{-1}
end while
```

IV. Focused Mapping for Navigation

In this section, we return to the question of variable selection that generates an input to the measurement selection scheme described in the previous section. The selection of focused variables mainly depends on the robot's tasks and goals. Here, we specifically consider the task of collision-free navigation.

Assume that the robot gathered some data, and needs to map a set of landmarks that can be used to localize itself and navigate through the environment. The dataset contains both the robot's path as well as measurements of landmarks along the path. Denote the robot's trajectory as a sequence of random variables $X = \{X_1, \ldots, X_T\}$. In GPS-denied environments, X is not directly observable. However, the robot can always measure the incremental change between two sequential poses (odometry), for example from an IMU or wheel encoder. There also exists a set of landmarks from which focused landmarks can be selected. Denote the set of landmarks as $L = \{L_1, L_2, \ldots L_N\}$.

A. Selection of Focused Landmarks

While there may exist thousands of landmarks, typically a small, carefully chosen subset can result in sufficiently accurate navigation. This indicates we can reduce the amount of landmarks without significantly degrading navigation performance. Reducing the number of landmarks is desirable, as it will significantly reduce the required computation for data association, which will in turn enable faster and more efficient on-line trajectory planning and navigation.

In navigation, narrow passages are especially challenging for collision-free motion planning. However, in the case of high map and robot uncertainty, the "narrowness" should be redefined. As shown in Fig. 5, a "geometrically" wide passage might still be problematic for a robot that does not have access to accurate landmark information, and thus, has poor localization accuracy.

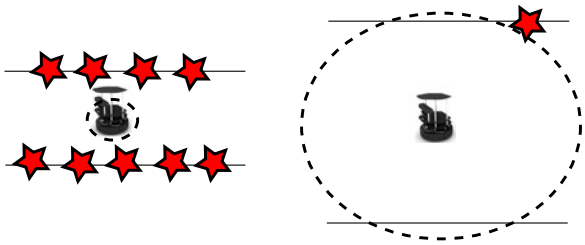


Fig. 5. *Probabilistic narrowness:* A measure of the probability of collision while navigating the environment. *Left:* This geometrically narrow corridor is probabilistically wide because there are many landmarks with which to localize. *Right:* Conversely, this geometrically wide hallway is probabilistically narrow.

We refer to this passage as being "geometrically wide" but "probabilistically narrow" and we will formalize these terms below.

The evaluation of probabilistic narrowness involves two key components: 1) Calculating an estimate of the robot's position uncertainty and 2) Calculating the probability of collision based on the robot's uncertainty and the distance to obstacles as determined by the robot.

1) Robot Position Uncertainty: To generate an estimate of the robot's position uncertainty at any given point x along the robot's path X, we use the concept of belief stabilization since it is a trajectory-independent measure [31].

Assume we have a closed-loop controller that can stabilize the robot's state to belief state x. Such a controller is typically comprised of an estimator and a separated controller. The estimator generates an *a posteriori* distribution over all robot poses based on the existing map of landmarks and the local observation of landmarks. Given these estimates, the separated controller will generate a control signal that drives the robot toward x. To design an analytic measure of narrowness, we rely on a simple Linear-Quadratic-Gaussian (LQG) controller, which combines a Kalman filter and a linear quadratic regulator. It can be shown [31] that starting from any $\Sigma_0 > \Sigma^*(x)$, the estimation covariance decreases monotonically and approaches the covariance $\Sigma^*(x)$, which is the fixed point of a Riccati recursion at location x

$$\{\Sigma^* = Q + A(\Sigma^* - \Sigma^* H^T (H\Sigma^* H^T + R)^{-1} H\Sigma^*) A^T\}_x. \quad (20)$$

The Jacobians, A and H are computed by linearizing the process and measurement model at point x on the path, where Q and R are the process and measurement noise, respectively. A visual depiction of the stabilization process is shown in Fig. 6. We associate a set of visible landmarks $L(x)$ (the ones connected to the robot by blue lines in Fig. 6) with each pose x. The value of the steady-state covariance will depend on the set of visible landmarks as expressed through the measurement noise covariance R. Note that the set of visible landmarks and Jacobian matrices would depend on the point x on the path. However, they do not depend on the path that leads to x. Therefore, main computational advantages can be gained by parallelizing the computation of $\Sigma^*(x)$.

2) Collision Probability: Using $\mathcal{N}(x, \Sigma^*(x))$ as a measure of uncertainty for each point x on the path, the collision probability can be defined by a Monte Carlo method. For each x,

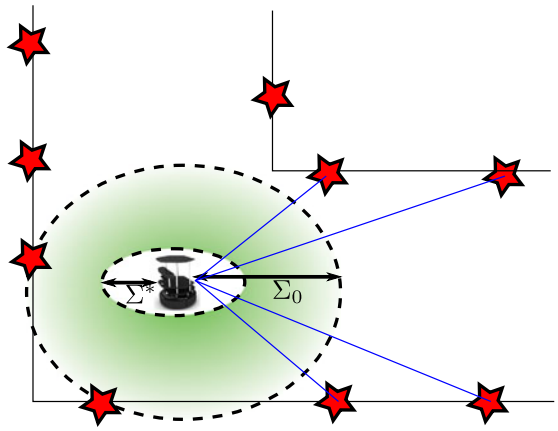


Fig. 6. *Stabilization*: Based on the landmark estimates, we solve a Riccati recursion to determine the minimum robot uncertainty at each point along the robot's path. This value is used as the measure of "probabilistic narrowness" introduced in Fig. 5.

sample the normal distribution and denote I as the set samples drawn from $\mathcal{N}(x, \Sigma^*(x))$. Denote O_{bs} as the set of obstacles, and $I \cap O_{bs}$ as the set of samples that collide with obstacles. Then, the collision probability can be defined as

$$P_c(x) = \lim_{|I|\to\infty} \frac{|I \cap O_{bs}|}{|I|}. \quad (21)$$

Monte Carlo methods are typically computationally expensive, and thus, not well-suited for real-time implementation. There have been many approximate methods, such as [18], [37], [38]. Here, we utilize an approximate measure that is computationally cheaper. First, denote the set of readings received from the laser range finder in its local frame as $D = \{d^i\}$ where each d^i is a 2-D vector connecting x to a point on the obstacle surface. Then, we simply compute the minimum Mahalanobis distance between x and the obstacle surface as

$$P_c(x) = \min_{d^i \in \bar{D}} \{(d^i)^T \Omega d^i\} \quad (22)$$

where $\Omega = (\Sigma^*(x))^{-1}$ is the information matrix corresponding to x. $\bar{D}$ could be equal to D or could be a subsampled version of D based on the available computational resources. In the extreme case, $\bar{D}$ could be a single point $\bar{D} = x^{\text{obst}} = \arg\min_{d^i \in D} \|d^i\|$ on the obstacle surface. In other words, $P_c(x)$ measures how near the closest obstacle point is from any possible poses of x in terms of standard deviations from the mean.

Notice the approximation retains the ability to favor some unbalanced covariance choices. For example, when one eigenvalue is significantly larger than the other, resulting in a "collision" in the direction of the larger one, the approximation will still have a high cost along that direction.

The landmark selection problem is framed as finding the poses along the trajectory with the highest approximate collision probability.

Problem 4 (Minimum Collision Probability Landmark Selection): Select the α landmarks $L_f \subset L$, such that the worst case

Algorithm 2: Minimum Collision Probability Landmark Selection.

Input: Robot poses X, landmarks L, odometry noise Q, measurement noise R, budget α
Output: Selected landmarks L_k^f
1: $k \leftarrow 0$, $L_0^f \leftarrow \emptyset$
2: Compute $P_c(x)$ for all $x \in X$ by (20) and (22)
3: $P(l) = \min_{x \text{ s.t. } l \in L(x)} P_c(x)$ for all $l \in L$
4: **while** $k < \alpha$ **do**
5: $k \leftarrow k + 1$
6: $l_k^* \leftarrow \text{argmin}_{l \in L \setminus L_{k-1}^f} P(l)$
7: $L_k^f = L_{k-1}^f \cup \{l_k^*\}$
8: update $P_c(x)$ for x that can observe l_k^*
9: update any $P(l)$ that may have changed
10: **end while**

probability of collision is minimized as

$$\max_{L_f \subset L} \min_x P_c(x) \quad \text{s.t.} \quad |L_f| \leq \alpha. \tag{23}$$

The problem is solved approximately by greedily selecting landmarks as summarized in Algorithm 2. At each iteration, pick the landmark l^* that is associated with minimal $P_c(x)$, then update $P_c(x)$ for all the x that can observe l^*. Since the number of poses that can observe any individual landmark is low, this greedy approximation can be computed efficiently.

B. Focused Measurement Selection for Navigation

Here we detail how to apply this landmark selection scheme resulting from solving Problem 4 to the measurement selection process described in Section III. Recall random variables X to represent robot poses and L to represent landmarks. Recall that the joint log probability of a factor graph is proportional to the sum of the factors $\phi(X_c, L_c|o, z)$

$$p(X, L) \propto \exp\left(\sum_{c \in \mathcal{C}} \phi_c(X_c, L_c|o, z)\right). \tag{24}$$

An odometry measurement, o, adds a factor between two subsequent poses $\phi_t(X_t, X_{t+1}; o_t)$ and a landmark measurement, z, adds a factor between a pose and a landmark $\phi_t^i(X_t, L_i; z_t^i)$.

The focused variables are selected focused landmarks: $\widetilde{X} = L_f \subset L$. The rest of the unfocused variables, including the rest of the unfocused landmarks and robot poses, are denoted as $\widetilde{X}' = \{L \setminus L_f, Y\}$. Then, the affine prioritization function represented by W is

$$\widetilde{X} = W \begin{bmatrix} L_f \\ \widetilde{X}' \end{bmatrix}, \quad W = \begin{bmatrix} I_{\tilde{N} \times \tilde{N}} & \mathbf{0}_{\tilde{N} \times N - \tilde{N}} \end{bmatrix}. \tag{25}$$

Using the standard assumption that the odometry and landmark measurement are corrupted by additive. Gaussian noise [14], then the factors are

$$\begin{aligned} X_0 &\sim \mathcal{N}(\mu_0, \Sigma_0) \\ X_{t+1} &= f(X_t, o_t) + \eta\,, \eta \sim \mathcal{N}(0, Q_t) \\ \phi_t(X_t, X_{t+1}) &= -\frac{1}{2}\left(X_{t+1} - f(X_t, o_t)\right)^T \\ &\quad \times Q_t^{-1}(X_{t+1} - f(X_t, o_t)) \\ z_t^i &= g(X_t, L_i) + \nu\,, \nu \sim \mathcal{N}(0, R_{t,i}) \\ \phi_t^i(X_t, L_i) &= -\frac{1}{2}(z_t^i - g(X_t, L_i))^T R_{t,i}^{-1}(z_t^i - g(X_t, L_i)). \end{aligned}$$

The W matrix and the factors can be used in Algorithm 1 now for the second stage (focused measurement selection).

Notice that the Gaussian approximation of the joint likelihood (9) is subject to the linearization point. An ideal linearization point would be the maximal likelihood value where the gradient is 0. However, computing this optimal value would involve optimizing the full graph with all variables and edges, which would become slow and resource intensive quickly when the problem gets bigger. Instead, we use the odometry based initial value as the linearization point. While it may subject to performance loss, this approach significantly reduces the computational burden.

For this choice of W, (16) can be further simplified. First decompose $\Lambda_{z^R}^{-1}$ into blocks of focused and unfocused variables

$$\Lambda_{z^R}^{-1} = \begin{bmatrix} A & B \\ B^T & C \end{bmatrix} \tag{26}$$

with $A = A^T > 0$ and $C = C^T > 0$. Then

$$\begin{aligned} L_k^T \tilde{\Lambda}_{z^{R_{k-1}}} L_k = J_k^T \Lambda_{z^{R_{k-1}}}^{-1} W^T \left(W \Lambda_{z^{R_{k-1}}}^{-1} W^T\right)^{-1} \\ \times W \Lambda_{z^{R_{k-1}}}^{-1} J_k. \end{aligned} \tag{27}$$

Inserting (26) into (25) gives

$$\left(W \Lambda_{z^{R_{k-1}}}^{-1} W^T\right)^{-1} = A^{-1} \tag{28}$$

and

$$\begin{aligned} L_k^T \tilde{\Lambda}_{z^{R_{k-1}}} L_k &= J_k^T \begin{bmatrix} A & B \\ B^T & C \end{bmatrix} \begin{bmatrix} I \\ 0 \end{bmatrix} A^{-1} \begin{bmatrix} I & 0 \end{bmatrix} \begin{bmatrix} A & B \\ B^T & C \end{bmatrix} J_k \\ &= J_k^T \begin{bmatrix} A & B \\ B^T & B^T A^{-1} B \end{bmatrix} J_k \\ &= J_k^T \left(\Lambda_{z^{R_{k-1}}}^{-1} - \begin{bmatrix} 0 & 0 \\ 0 & C - B^T A^{-1} B \end{bmatrix} \right) J_k \\ &= J_k^T \left(\Lambda_{z^{R_{k-1}}}^{-1} - \begin{bmatrix} 0 & 0 \\ 0 & (\Lambda_{z^{R_{k-1}}}^{C})^{-1} \end{bmatrix} \right) J_k \end{aligned} \tag{29}$$

where $\Lambda_{z^{R_{k-1}}}^{C}$ denotes the submatrix of $\Lambda_{z^{R_{k-1}}}$ corresponding to the unfocused variables and $(\Lambda_{z^{R_{k-1}}}^{C})^{-1}$ is the marginal covariance matrix of unfocused variables. Inserting (29) into (16)

yields

$$\begin{aligned}
&\Delta\hat{H}(\tilde{X}|z_k^R) \\
&= -\frac{1}{2}\log\left|I-(I+J_k^T\Lambda^{-1}_{z^{R_{k-1}}}J_k)^{-1}L_k^T\tilde{\Lambda}_{z^{R_{k-1}}}L_k\right| \\
&= \frac{1}{2}\log\left|I+J_k^T\Lambda^{-1}_{z^{R_{k-1}}}J_k\right| \\
&\quad -\frac{1}{2}\log\left|I+J_k^T\Lambda^{-1}_{z^{R_{k-1}}}J_k-L_k^T\tilde{\Lambda}_{z^{R_{k-1}}}L_k\right| \\
&= \frac{1}{2}\log\left|I+J_k^T\Lambda^{-1}_{z^{R_{k-1}}}J_k\right| \\
&\quad -\frac{1}{2}\log\left|I+J_k^T\begin{bmatrix}0 & 0\\ 0 & (\Lambda^C_{z^{R_{k-1}}})^{-1}\end{bmatrix}J_k\right|. \qquad (30)
\end{aligned}$$

Note that the form of (30) is intuitive. The first term is the information gain on all variables. The second term has a similar form, but is computed only on unfocused variables. The difference of these two terms is the information gain on focused variables. Depending on whether the new measurement is of a focused landmark or unfocused landmark, (30) can be further simplified, as outlined in the following.

Case 1: Measurement of focused landmark, $l_i \in L_f$. In this case, J_k is nonzero only at the ith row corresponding to the focused landmark and the jth row corresponding to the robot pose, $J_k = [\ldots, J_k, \ldots, J_j, \ldots]_k^T$. Denote Λ^{-1}_{ij} as the element corresponding to i, j location of matrix $\Lambda^{-1}_{z^{R_{k-1}}}$, and $(\Lambda^C_{ij})^{-1}$ as the element corresponding to i, j location of matrix $(\Lambda^C_{z^{R_{k-1}}})^{-1}$, then (30) can be further simplified to

$$\begin{aligned}
\Delta\hat{H}(\tilde{X}|z_k^R) &= \frac{1}{2}\log\left|I+\begin{bmatrix}J_i\\ J_j\end{bmatrix}^T\begin{bmatrix}\Lambda^{-1}_{ii} & \Lambda^{-1}_{ij}\\ \Lambda^{-1}_{ji} & \Lambda^{-1}_{jj}\end{bmatrix}\begin{bmatrix}J_i\\ J_j\end{bmatrix}\right| \\
&\quad -\frac{1}{2}\log\left|I+J_j^T(\Lambda^C_{jj})^{-1}J_j\right|. \qquad (31)
\end{aligned}$$

Case 2: Measurement of unfocused landmark, $l_i \notin L_f$

$$\begin{aligned}
\Delta\hat{H}(\tilde{X}|z_k^R) &= \frac{1}{2}\log\left|I+\begin{bmatrix}J_i\\ J_j\end{bmatrix}^T\begin{bmatrix}\Lambda^{-1}_{ii} & \Lambda^{-1}_{ij}\\ \Lambda^{-1}_{ji} & \Lambda^{-1}_{jj}\end{bmatrix}\begin{bmatrix}J_i\\ J_j\end{bmatrix}\right| \\
&\quad -\frac{1}{2}\log\left|I+\begin{bmatrix}J_i\\ J_j\end{bmatrix}^T\begin{bmatrix}(\Lambda^C_{ii})^{-1} & (\Lambda^C_{ij})^{-1}\\ (\Lambda^C_{ji})^{-1} & (\Lambda^C_{jj})^{-1}\end{bmatrix}\begin{bmatrix}J_i\\ J_j\end{bmatrix}\right|. \qquad (32)
\end{aligned}$$

Note that the first terms of (32) and (31) are identical. However, only J_j contributes to the second term in (31) but both J_i and J_j contribute to the second term in (32). Case 2 has larger information gain over the unfocused variables as compared to Case 1, and consequently, the change in total information is much smaller.

C. Robot Pose Sparsification

The variables in the graph consist of both landmarks and robot poses. After marking certain landmarks as focused variables and selecting measurements, the graph structure is significantly sparsified. However, the graph size is still big, as all the variables including both landmarks and robot poses are still maintained in the graph, which will exceed the robot memory constraint quickly.

Observe that after c measurements are selected, there are at most c robot poses that are connected to any landmarks in the graph. Most robot poses are only connected to the previous and subsequent poses (an odometric chain). Marginalizing out such robot poses is an odometry composition. Robot poses without any landmark measurements are marginalized out to reduce graph size. If an unfocused landmark is not connected to any robot poses, it is also marginalized out. The same linearization point is maintained during the variable and measurement selection, to minimize information loss.

After these two marginalizations, the number of variables in the graph is linear in the measurement constraint, which is given as an input as a memory and computation constraint.

V. Navigation With Uncertain Landmarks

Different from classical path planning algorithms with known map of landmarks for localization, the environment map here is given as a set of stochastic landmarks $L \sim \mathcal{N}(\widehat{L}, R^L)$ learned from SLAM. In this section, we discuss how to use such a map to navigate. In particular, an LQG controller is designed that tracks a trajectory connecting robots initial point to its goal. First, the trajectory is discretized to T steps as $(x_i^d)_{i=0}^T$ and $(u_i^d)_{i=0}^{T-1}$ where, x_k^d and u_k^d denote the desired state and control signal at the kth time step on the trajectory. Assume that the robot starts at $x_0 \sim \mathcal{N}(\widehat{x}_0, \Sigma_0)$.

We formulate the problem by incorporating a stochastic map into the LQG framework. Assume that the locations of the landmarks are random variables with a mean and covariance given by the map data structure.

The nonlinear partially observable state-space equations of the system are

$$x_{k+1} = f(x_k, u_k, w_k), \quad w_k \sim \mathcal{N}(0, Q_k) \qquad (33a)$$

$$z_k = h(x_k, L, v_k), \quad v_k \sim \mathcal{N}(0, R_k), \quad L \sim \mathcal{N}(\widehat{L}, R^L). \qquad (33b)$$

Represent the control problem as

$$\begin{aligned}
\min_{\mu_t(\cdot)}\ &\mathbb{E}\left[\sum_{t=0}^{N}(x_t-x_t^d)^T W_x(x_t-x_t^d)\right. \\
&\left.\quad +(u_t-u_t^d)^T W_u(u_t-u_t^d)\right] \\
&x_{k+1} = f(x_k, u_k, w_k), w_k \sim \mathcal{N}(0, Q_k) \\
&z_k = h(x_k, L, v_k), v_k \sim \mathcal{N}(0, R_k), L \sim \mathcal{N}(\widehat{L}, R^L) \\
&u_t = \mu_t(b_t) \\
&b_t = p(x_t|z_{0:t}, u_{0:t-1}). \qquad (34)
\end{aligned}$$

To transform the problem in (34) into an LQG problem, we first compute a time-varying linear system by linearizing the

nonlinear system about the nominal trajectory $(x_k^d, u_k^d)_{k\geq 0}$

$$x_{k+1} = f(x_k^d, u_k^d, 0) + A_k(x_k - x_k^d) + B_k(u_k - u_k^d) + G_k w_k, \quad w_k \sim \mathcal{N}(0, Q_k) \tag{35a}$$

$$z_k = h(x_k^d, \widehat{L}, 0) + H_k(x_k - x_k^d) + M_k^L(L - \widehat{L}) + M_k v_k, \quad v_k \sim \mathcal{N}(0, R_k), L \sim \mathcal{N}(\widehat{L}, R^L) \tag{35b}$$

where

$$A_k = \frac{\partial f}{\partial x}(x_k^d, u_k^d, 0), \quad B_k = \frac{\partial f}{\partial u}(x_k^d, u_k^d, 0) \tag{36a}$$

$$G_k = \frac{\partial f}{\partial w}(x_k^d, u_k^d, 0), \quad H_k = \frac{\partial h}{\partial x}(x_k^d, \widehat{L}, 0) \tag{36b}$$

$$M_k = \frac{\partial h}{\partial v}(x_k^d, \widehat{L}, 0), \quad M_k^L = \frac{\partial h}{\partial v^L}(x_k^d, \widehat{L}, 0). \tag{36c}$$

Now define the following errors:

1) LQG error (main error): $e_k = x_k - x_k^d$.
2) Map error: $v^L = L - \widehat{L}$.
3) Kalman filter error (KF estimation error): $\widetilde{e}_k = x_k - \widehat{x}_k^+$.
4) LQR error (estimation of LQG error): $\widehat{e}_k^+ = \widehat{x}_k^+ - x_k^d$

where $\widehat{x}_k^+$ refers to the mean of estimated state at the kth time step. Let $\delta u_k = u_k - u_k^d$ and $\delta z_k = z_k - z_k^d := z_k - h(x_k^d, \widehat{L}, 0)$, then the linearized models in (35) can be rewritten as

$$e_{k+1} = A_k e_k + B_k \delta u_k + G_k w_k, \quad w_k \sim \mathcal{N}(0, Q_k) \tag{37a}$$

$$\delta z_k = H_k e_k + M_k^L v^L + M_k v_k, \quad v_k \sim \mathcal{N}(0, R_k), v^L \sim \mathcal{N}(0, R^L). \tag{37b}$$

Defining $\bar{M}_k = [M_k^L M_k]$ and $\bar{v}_k = [{v^L}^T v_k^T]^T$ and $\bar{R}_k = \text{diag}[R^L, R_k]$, the observation equation can be written as

$$\delta z_t = H_k e_k + \bar{M}_k \bar{v}_k, \bar{v}_k \sim \mathcal{N}(0, \bar{R}_k). \tag{38}$$

The last step is to write the control problem in the linear error space as

$$\begin{aligned} \min_{\mu_t(\cdot)} \; & J = \mathbb{E}\left[\sum_{t=0}^{N} e_t^T W_x e_t + \delta u_t^T W_u \delta u_t\right] \\ & e_{k+1} = A_k e_k + B_k \delta u_k + G_k w_k, w_k \sim \mathcal{N}(0, Q_k) \\ & \delta z_t = H_k e_k + \bar{M}_k \bar{v}_k, \bar{v}_k \sim \mathcal{N}(0, \bar{R}_k) \\ & \delta u_t = \mu_t(b_t) - u_t^d \\ & b_t = p(e_t + x_t^d | \delta z_{0:t}, \delta u_{0:t-1}). \end{aligned} \tag{39}$$

Now, the problem in (39) is in the standard LQG form. Note that $e_k = \widetilde{e}_k + \widehat{e}_k^+$ and, based on the separation principle [39], it can be shown that minimizing the quadratic objective in (39) can be divided into two separate minimizations over the estimation error $\widehat{e}_k^+$ and the separated controller error $\widetilde{e}_k$. In the following, we discuss how a KF and an LQR can be designed for this linearized system and finally combine them to construct a time-varying LQG controller.

Kalman Filter: In Kalman filtering, we aim to provide an estimate of the system's state based on the available partial information we have obtained until time k, i.e., $z_{0:k}$. The error estimate is a random vector denoted by e_k^+, whose distribution is the conditional distribution of the state on the obtained observations so far, which is called belief and is denoted by b_k as

$$b_k = p(x_k^+) = p(x_k | z_{0:k}) = \mathcal{N}(\widehat{e}_k^+ + x_k^d, P_k) \tag{40}$$

$$\widehat{e}_k^+ = \mathbb{E}[e_k | \delta z_{0:k}, \delta u_{0:k-1}] \tag{41}$$

$$P_k = \mathbb{C}[e_k | \delta z_{0:k}, \delta u_{0:k-1}] \tag{42}$$

where $\mathbb{E}[\cdot|\cdot]$ and $\mathbb{C}[\cdot|\cdot]$ are the conditional expectation and conditional covariance operators, respectively.

Kalman filtering consists of two steps at every time stage: prediction step and update step. In the prediction step, the mean and covariance of prior e_k^- is computed. For the system in (37) prediction step is

$$\widehat{e}_{k+1}^- = A_k \widehat{e}_k^+ + B_k \delta u_k \tag{43}$$

$$P_{k+1}^- = A_k P_k^+ A_k^T + G_k Q_k G_k^T. \tag{44}$$

In the update step, the mean and covariance of posterior e_k^+ is computed. For the system in (37), the update step is

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + \bar{M}_k \bar{R}_k \bar{M}_k^T)^{-1} \tag{45}$$

$$\widehat{e}_{k+1}^+ = \widehat{e}_{k+1}^- + K_{k+1}(\delta z_{k+1} - H_{k+1} \widehat{e}_{k+1}^-) \tag{46}$$

$$P_{k+1}^+ = (I - K_{k+1} H_{k+1}) P_{k+1}^-. \tag{47}$$

LQR Controller: Once we obtain the belief from the filter, a controller can generate an optimal control signal accordingly. In other words, we have a time-varying mapping μ_k from the belief space into the control space that generates an optimal control based on the given belief $u_k = \mu_k(b_k)$ at every time step k. LQR controller is of this kind and it is optimal in the sense of minimizing cost

$$J_{LQR} = \mathbb{E}\left[\sum_{k\geq 0} (\widehat{e}_k^+)^T W_x (\widehat{e}_k^+) + (\delta u_k)^T W_u (\delta u_k)\right].$$

The linear control law that minimizes this cost function for a linear system is of the form

$$\delta u_k = -F_k \widehat{e}_k^+ \tag{48}$$

where the time-varying feedback gains F_k can be computed recursively as

$$F_k = (B_k^T S_{k+1} B_k + W_u)^{-1} B_k^T S_{k+1} A_k \tag{49}$$

$$S_k = W_x + A_k^T S_{k+1} A_k - A_k^T S_{k+1} B_k F_k. \tag{50}$$

If the nominal path is of length N, then the $S_N = W_x$ is the initial condition of above recursion, which is solved backwards in time. Note that the final controller is

$$\begin{aligned} u_k &= u_k^d + \delta u_k \\ &= u_k^d - (B_k^T S_{k+1} B_k + W_u)^{-1} B_k^T S_{k+1} A_k \widehat{e}_k^+. \end{aligned} \tag{51}$$

It should be especially noticed that when stochastic landmarks are incorporated in the navigation mechanism, we can predict the navigation accuracy based on the map accuracy. This information is utilized in the map generation phase. In other words, because we especially optimized the uncertainties on key landmarks in the map, the robot will have high confidence to go through narrow passages and successfully reach the goal.

VI. Incremental Mapping

Mobile robot autonomy will typically require operation over long time periods and large distances. In such cases, batch selection on a big dataset would be extremely slow, and would likely exceed the resources onboard the robot. As such, it is particularly important to develop algorithms that can perform both variable and measurement reduction incrementally using smaller sets of data. The variable reduction would enable the use of a small model of the environment, thus saving memory. The measurement reduction enables the use of a sparse graphical model, which reduces the computational effort and reduces the inference computation times.

The incremental pipeline is illustrated in Fig. 2, with an arrow connecting map building step back to data collection step. After some operation in the environment collecting some data, the robot first performs the two-stage landmark and measurement selection procedures in Sections III and IV to reduce that dataset. The reduced dataset is then used to update the partial map of the environment constructed by the robot. The robot can navigate with the updated partial map to newly defined goal locations, such as frontiers. More data would be collected during that further operation, which enables the robot to incrementally interleave the mapping/exploration operation with the measurement/landmark reduction.

At some time point during operation, assume the robot has finished $t-1$ operations of two-stage selection. Denote the graph to $t-1$ steps of operation as G_{t-1}, the data the robot gathers during operation t as z^t, and the landmarks observed during operation t as L^t, then the incremental two-stage selection process is described by Problem 5.

Problem 5: A. Incremental Focused Measurement Selection: Given graph G_{t-1}, select a minimal set of measurements z^{R_t} from newly obtained measurements z^t during operation t, such that the information metric $f(\cdot)$ over the **focused** variables $\widetilde{X}^t$ is bounded

$$\max_{z^{R_t} \subset z^t} \; g(z^{R_t})$$
$$\text{s.t.} \quad f(\widetilde{X}^t; z^{R_t} | G_{t-1}) \geq c^t. \tag{52}$$

B. Incremental Landmark Selection: Given graph G_{t-1}, select a minimal set of landmarks L_f^t from landmarks L^t observed during operation t, such that the maximum probability of collision is bounded

$$\min_{L_f^t \subset L^t} \; |L_f^t|$$
$$\text{s.t.} \quad \max_{x_k} P_c(x_k | G_{t-1}) \leq \alpha^t. \tag{53}$$

Algorithm 3: Incremental Focused Mapping.

Input: Initial graph $G_0 = \emptyset$, $t = 0$
1: **while** not stopped **do**
2: $t = t + 1$
3: Operate robot and get data z^t, L^t
4: Select landmarks L_f^t with (53) and Algorithm 2
5: Select measurements z^{R_t} with (52) and Algorithm 1
6: $G_t = G_{t-1} \cup \{z^{R_t}, L_f^t\}$
7: **end while**

Note that Problem 5 differs from Problems 2 and 4 in two aspects. First, the new metrics in (52) and (53) are conditioned on the reduced graphical model from the previous operation G_{t-1}. Second, Problem 5 enforces the collision probability and information gain as constraints, instead of optimizing information gain and collision probability given a landmark and measurement budget. In a single batch setting, the robot has access to all of the data, which represents the entire environment. In an incremental setting, the robot only has access to a single subset of the data at a given time. Each subset only represents a part of the environment and different subsets may correspond to different geometric shapes of the environments. For example, one subset represents narrow spaces while others corresponds to more open space. If the incremental problem is set up as Problems 2 and 4, then a uniform resource budget is enforced for all different data subsets. As a result, it will lead to wasted resources on open spaces and low accuracy on narrow places. Therefore, Problem 5 enforces collision probability and information gain as constraints, then greedily selects landmarks and measurements until the constraints are satisfied. The incremental algorithm is summarized in Algorithm 3.

VII. Experiments

We present simulation and real-world results.[3] First, experiments are run in a simulated environment, where ground truth is available to compare accuracies. In the real-world experiment, we ran a turtlebot in a cluttered office space.

A. Simulations

Fig. 7 shows the simulation environment. It includes both open areas on the left-hand side and narrow passages on the right-hand side. The parameters and statistics are listed in Table I. We first run the robot once to get the initial dataset, then reduce robot poses and landmark measurements with various approaches. The sparsified maps are optimized with iSAM [40]. Repeated trials were performed on the generated maps to test the collision probability when navigating from a start point to a goal point. Five selection strategies were compared:

1) *optimal:* use all landmarks and all measurements;
2a) *coverage, glc, global:* use maximal coverage to proposed in [19] and [41] to select landmarks and robot poses. Thirty landmarks are selected and robot poses are selected when the robot travels longer than 1.2 m. GLC is used to

[3]The software is available for download at: https://github.com/BeipengMu/focused_slam.git.

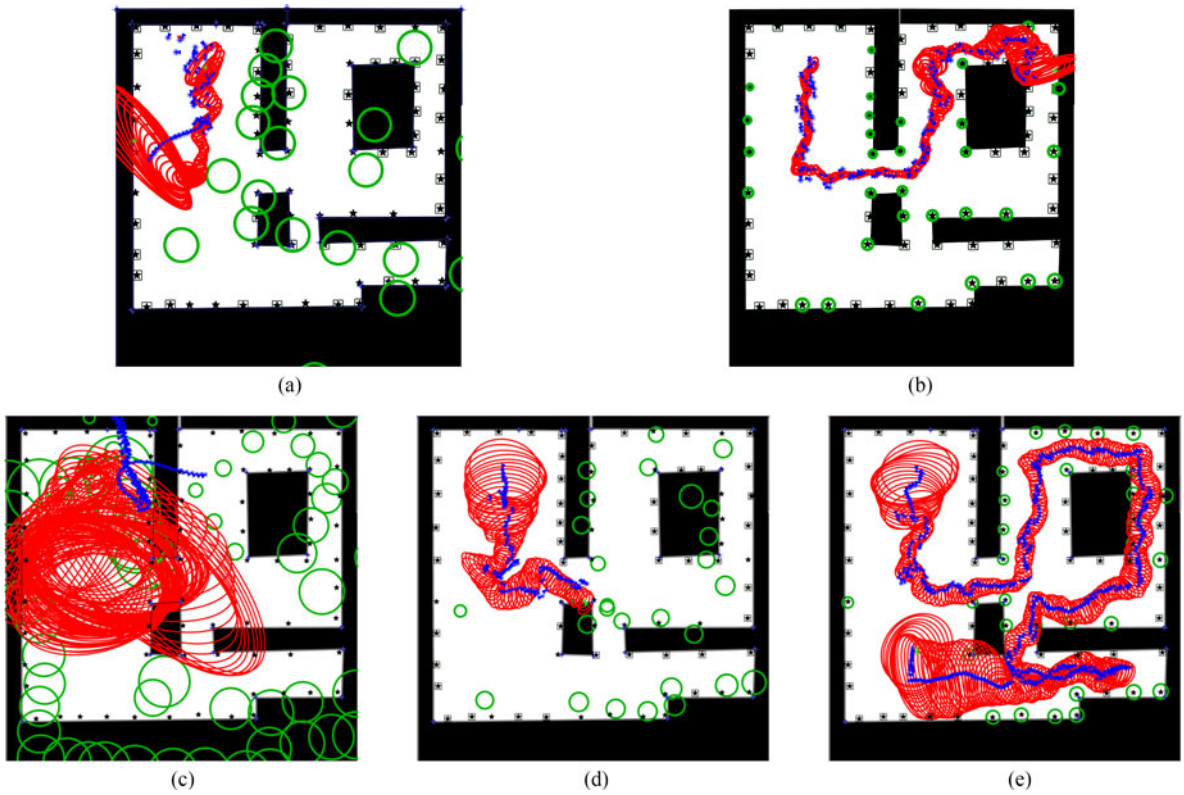


Fig. 7. Navigation with focused landmark selection and map building. Green circles represent selected landmarks with their size representing uncertainty. Blue lines are the nominal trajectories each robot wants to follow with red circles representing pose uncertainty. The focused approach can successfully navigate through narrow passages. (a) case 2 maximal coverage and GLC, without optimal initialization, (b) case 2 maximal coverage and glc with optimal initialization, (c) case 3 only select measurements, (d) case 4 only select landmarks, (e) case 5 two-stage selection.

TABLE I
SIMULATED DATASET

environment size	120 m × 120 m
distance traveled	1066.9 m
robot field of view	35 m, 180^o
landmarks (black stars)	74
odometry measurements	1992
landmark measurements	12 760
focused landmarks α	30
landmark measurement budget K^R	80
Odometry noise Q	0.1
Measurements noise R	0.1

marginalize out unselected variables [7]. Global optimal point is used to initialize GLC.

2b) *coverage, glc, direct:* same as (2a), but use odometry as initialization of GLC.

3) *all, info:* use all landmarks, and apply measurement selection based on information gain proposed in [5] until the number of measurements exceeds a threshold;

4) *focus, down-sampling:* select focused landmarks and uniformly down-sample measurements;

5) *focus, info:* select focused landmarks for minimal collision, and select measurements based on information gain on focused variables (our proposed method);

Case (1) is a bound on other approaches. Case (2) is a map reduction approach that starts with full variables and measurements. Typically, as implemented with (2a), the full graph is optimized first, and then, reduced with the global optimal point. However, in our case, we try to avoid large scale global optimization and do not have access to the optimal values, as implemented in (2b). Case (3) is a measurement sparsification approach but does not sparsify variables. Case (4) uses focused landmarks to minimize collisions, but uses a naive way to sparsify measurements. Case (5) is our proposed two-stage selection algorithm.

Fig. 7(a)–(e) shows the sample trials for cases (2), (3), (4), and (5), respectively, and Fig. 8 shows the overall probabilities of collision obtained from all trials. The trials are stopped whenever there is an actual collision with an obstacle. The landmarks that are not selected are distinguished by small black boxes around them. The landmark uncertainty are shown in green. The blue line represents the robot trajectory. The red ellipses represent the uncertainties of the robot along the trajectory. In the focused landmark selection cases (5), the proposed procedure picks the landmarks that contribute more in reducing the robot's uncertainty in desired regions (narrow passages) and spend the computational budget to reduce the uncertainty of these focused landmarks. The key point to note from Fig. 7(e) is that the robot uncertainty is preferentially reduced in the areas of the environment where the corridors are tight and there is a higher chance of collision. In case (3), the measurements are spread across landmarks. As a result, each landmark gets very little resource thus the method fails to recover a meaningful

TABLE II
COMPARISON OF SIMULATED MAPPING RESULTS

case	Variable Selection	Measurement Selection	Number of Landmarks	Number of Poses	Number of Factors	Error on Landmarks (m)	Time(s)
1	all	all	74	1993	14 752	0.014	1.24
2a	max coverage	GLC	30	707	2932	0.012	960.2
2b	max coverage	GLC	30	707	2939	22.28	697.3
3	all	info gain	74	61	141	36.43	52.6
4	focus	down-sample	30	79	160	10.75	1.10
5	focus	info gain	**30**	**35**	**109**	0.21	69.43

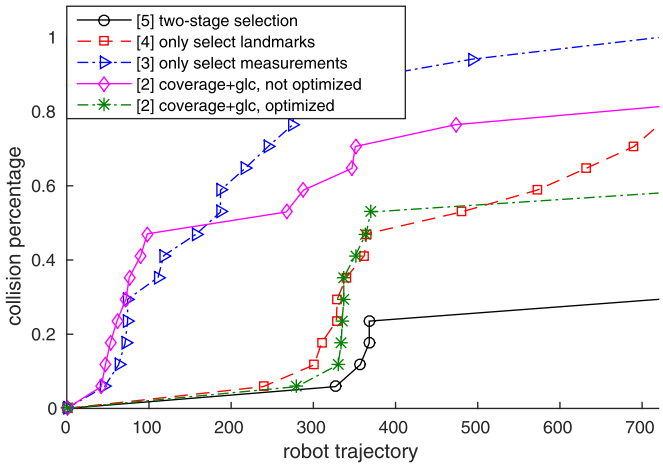


Fig. 8. Collision probability of Monte Carlo simulations. Focused two-stage algorithm (black) has lower collision probability compared to unfocused (other colors).

Fig. 9. Environment used for hardware experiments.

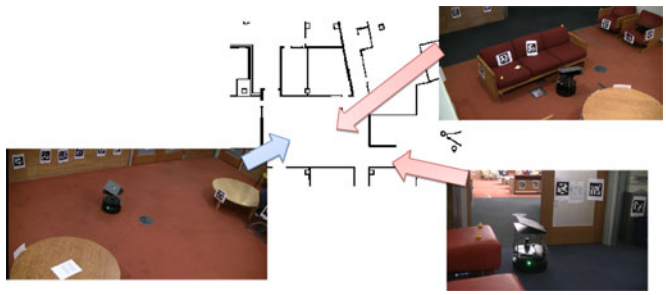


Fig. 10. Floor plan of the hardware experiment environment. Narrow passages include a door way and a sofa cluster.

TABLE III
OFFICE DATASET

length	10 min 24 s
distance traveled	115.5 m
# odometry measurements	5211
# landmark measurements	8252
# landmarks	114
focused landmarks α	30
landmark measurement budget K^R	90
Odometry noise Q	0.1
Measurements noise R	0.1

map for navigation, thus showing that, in this case, measurement selection alone would not produce an acceptable result. Case (4) only selects landmarks, more resources are spent on the focused landmarks, thus the map is more accurate, but measurement selection is not based on how much they contribute to uncertainty reduction, thus the landmark positions are much less accurate than selecting both (case 5). In case (2), landmarks and robot poses are reduced based on maximal coverage, and variables are removed via sparse GLC. First notice map reduction is more time-consuming than map selection, because the variables and measurements need to be removed are in the majority. And computing CLT for landmarks with big Markov blankets is especially time consuming. Without a prior optimization over the full graph, sparse GLC loses information through marginalization, and the remaining graph fails to converge. A prior optimization over the full graph gives optimal landmarks locations and GLC maintains the right locations through marginalization. However, full-graph optimization is not desirable, especially on resource constrained systems. Furthermore, maximal coverage does not maintain enough features in narrow passages, the robot has major collisions there even with accurate landmark estimates.

We further compare the cases from a mapping perspective in Table II based on three metrics: the sparsity of graph, represented by number of landmarks, robot poses and factors in the graph; the accuracy of SLAM, represented by the error on landmark locations. The proposed focused two-stage landmark and measurement selection approach achieves a sparser graph than others, maintains accurate estimates of landmarks, which results in sparser systems with good task performance.

The percentage of collision in Monte Carlo simulation is given in Fig. 8. Of all the trials, the proposed two-stage selection achieved significant less collisions. The collision probability of other approaches quickly go up in the very beginning due to poor

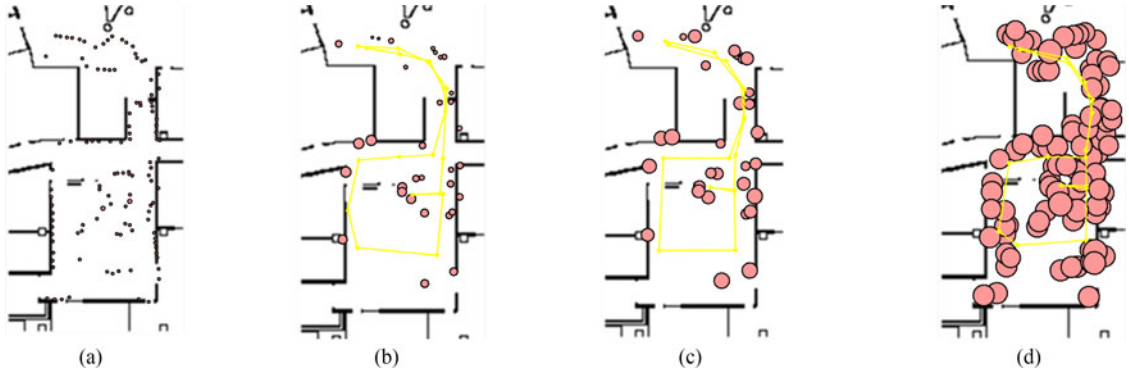


Fig. 11. Mapping results. Magenta circles represent landmarks with the size representing its uncertainty. The proposed two-stage approach (case 3) outperforms either selecting measurement only (case 5) or selecting landmark only (case 4) isolated, (a) case 1 optimal, (b) case 5 two-stage selection, and (c) case 4 only landmarks.

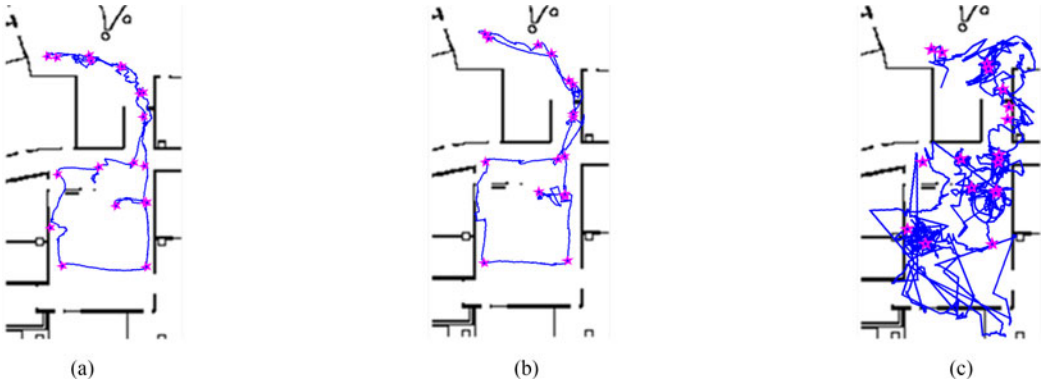


Fig. 12. Navigation results. Magenta stars represent designed waypoints for robot to follow. Blue lines represent the estimated robot trajectory by Kalman filter. The proposed two-stage approach (case 5, focus, info) has least collisions, while only measurement reduction (case 3, all, info) has bad estimates and only landmark reduction (case 4, focus, down-sampling) has more collisions. (a) case 5 two-stage selection, (b) case 4 only select landmarks, and (c) case 3 only select measurements.

map or around 300 steps into navigation, which corresponds to narrow passage at upper-right corner.

B. Hardware Experiment

In the hardware experiments, the turtlebot is equipped with an ASUS Xtion Pro RGB-D camera (we only use the RGB camera in this study) and a Hokuyo URG-04LX-UG01 laser-range finder. Fig. 10 shows the floor plan of the environment. AprilTags [42] were installed to create an initial pool of landmarks, as shown in Fig. 9. The area consists of an office with desks, a doorway, and an open atrium with couches and chairs scattered. A summary of the dataset is provided in Table III.

The odometry measurements are obtained from the turtlebot's wheel encoders. The landmark measurements are obtained by running the AprilTag detector with the RGB images, which gives the relative orientation and range of the tags in the robot's frame [42]. The selected measurements and odometry information are then fed into iSAM [15] to optimize the graph. Note that we did not use the laser data for SLAM, only for detecting the closest distance to obstacles.

Fig. 11 compares mapping results of case (1) (*optimal*), case (3) only select measurements, case (4) only select landmarks, and case (5) two-stage selection. The rebuilt robot trajectory is shown with a color map, where the red color on the trajectory indicates the risky (close to obstacles) regions and blue indicates the safer regions. Magenta circles represent landmarks with the size representing its uncertainty. The focused approach [see Fig. 11(b)] can concentrate the measurements on the narrow passage and door way, resulting in less uncertainty there. The other approaches scatter the measurements across different landmarks, and thus, have much higher landmark uncertainty in narrow passages.

TABLE IV
NAVIGATION PERFORMANCE

Method	length	# collisions
(5) two-stage selection	10 m 28 s	2
(4) only select landmarks	10 m 52 s	13
(3) only select measurements	28 m 40 s	20

Fig. 12 and Table IV compare the navigation results. Magenta stars represent designed waypoints for the robot to follow. Blue lines represent the estimated robot trajectory from a Kalman filter. The proposed two-stage approach (5) has least collisions, while only select landmarks (4) has more collisions and only select measurements (3) has bad estimates.

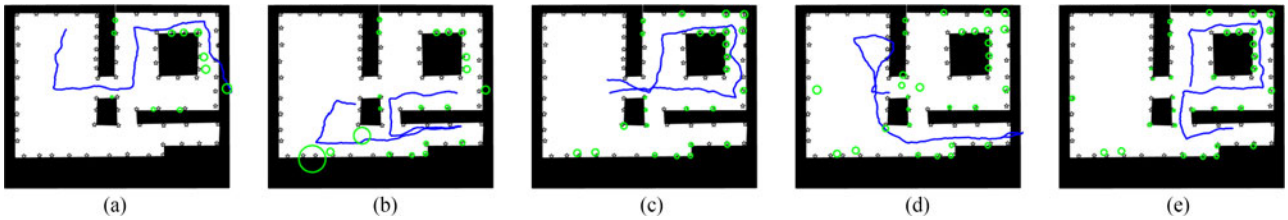


Fig. 13. Incremental selection. Green circles represent selected landmarks after processing each new batch. Blue lines represent the robot's trajectory in each batch. The robot is able to use existing focused landmarks as priors and augment focused landmarks when passing through narrow passages. (a) Batch 1, (b) Batch 2, (c) Batch 3, (d) Batch 4, and (e) Batch 5.

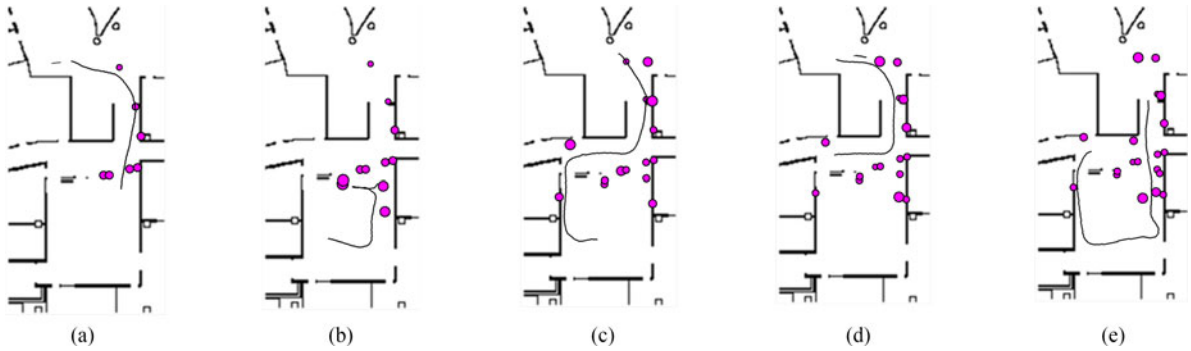


Fig. 14. Incremental selection. This dataset contains five batches. Magenta circles represent selected landmarks after processing each new batch. Black lines represent the robot's trajectory in each batch. The robot is able to use existing focused landmarks as priors and augment focused landmarks with new observed data. (a) Batch 1, (b) Batch 2, (c) Batch 3, (d) Batch 4, and (e) Batch 5.

TABLE V
COMPARISON OF BATCH AND INCREMENTAL SELECTION

	batch	incremental
Number of Landmarks	30	33
Number of Poses	30	43
Number of Edges	120	150
Mean Error on Landmarks	0.1345	0.8001
Min Mahalanobis Distance	14.53	27.05

TABLE VI
STATISTICS OF DATA BATCHES

Batch No.	Time(s)	Distance	No. measurements
1	120	16.87	1711
2	120	13.67	1049
3	120	25.12	1902
4	120	25.18	1240
5	120	29.10	1390

C. Incremental Selection

In the incremental setting, the robot iterates between landmark and measurement selection on streaming batches of data. When a new batch is processed, it selects focused variables and measurements using results from the previous batches as a prior. Both simulated and real-world data are used to show the incremental capability developed in Section VI.

In the simulated environment, the dataset in Table I is divided into five batches, and Algorithm 3 is applied. The Mahalanobis distance for selecting landmarks is set to be 100 and the information gain for selecting measurements is set to be 0.5. Fig. 13 shows how the algorithm progresses. In the first two batches of data, the robot selects landmarks in narrow passages as focused variables, however, the uncertainty over them is high as the robot has not closed a loop yet. After batch three, the robot closes loops, and significantly reduces the uncertainty on the focused landmarks. Table V compares the performance with incremental selection with batch selection.

The incremental algorithm is also tested on the office dataset in Table III. The dataset is divided into five batches. Their statistics are listed in Table VI.

Fig. 14 shows the map after processing each batch. Magenta circles represent selected landmarks after processing each new batch. Black lines represents the robot's trajectory in each batch. The robot is able to use existing focused landmarks as priors and augment focused landmarks with new observed data.

VIII. CONCLUSION

We have presented a two-stage landmark and measurement selection procedure designed for a real robot operating in unknown or uncertain environments. With exploring time and space growing, physical robotic system will have to discard data at some point. In this paper, we provide one principled choice of which data to retain. Namely, the data which is the most useful to achieve the task at hand. Simulations and hardware results demonstrate that the approach can identify a relevant subset of

landmarks and accurately localize them to reduce the probability of colliding with obstacles. As a result, the robot is able to navigate the environment, without the memory or computational requirements growing beyond the constraints. We have focused on the task of navigation as an example here, but there are many inference and planning tasks that require a similar prioritization of variables and measurements.

REFERENCES

[1] M. Wainwright and M. Jordan, "Graphical models, exponential families, and variational inference," *Found. Trends Mach. Learning*, vol. 1, nos. 1/2, pp. 1–305, Jan. 2008.

[2] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques*. Cambridge, MA, USA: MIT Press, 2009.

[3] C. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*, 1st ed. Berlin, Germany: Springer, 2007.

[4] T. Steiner, G. Huang, and J. Leonard, "Location utility-based map reduction," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2015, pp. 479–486.

[5] V. Ila, J. Porta, and J. Andrade-Cetto, "Information-based compact pose SLAM," *IEEE Trans. Robot.*, vol. 26, no. 1, pp. 78–93, Feb. 2010.

[6] H. Kretzschmar and C. Stachniss, "Information-theoretic compression of pose graphs for laser-based SLAM," *Int. J. Robot. Res.*, vol. 31, no. 11, pp. 1219–1230, 2012.

[7] N. Carlevaris-Bianco, M. Kaess, and R. M. Eustice, "Generic node removal for factor-graph SLAM," *IEEE Trans. Robot.*, vol. 30, no. 6, pp. 1371–1385, Dec. 2014.

[8] G. Huang, M. Kaess, and J. Leonard, "Consistent sparsification for graph optimization," in *Proc. Eur. Conf. Mobile Robots.*, Sep. 2013, pp. 150–157.

[9] M. Mazuran, W. Burgard, and G. Tipaldi, "Nonlinear factor recovery for long-term SLAM," *Int. J. Robot. Res.*, vol. 35, nos. 1–3, pp. 50–72, 2016.

[10] B. Mu, A. Agha-mohammadi, L. Paull, M. Graham, J. How, and J. Leonard, "Two-stage focused inference for resource-constrained collision-free navigation," in *Proc. Robot. Sci. Syst.*, Jul. 2015.

[11] S. Thrun, Y. Liu, D. Koller, A. Ng, Z. Ghahramani, and H. Durrant-Whyte, "Simultaneous localization and mapping with sparse extended information filters," *Int. J. Robot. Res.*, vol. 23, nos. 7/8, pp. 693–716, 2004.

[12] M. Walter, R. Eustice, and J. Leonard, "Exactly sparse extended information filters for feature-based SLAM," *Int. J. Robot. Res.*, vol. 26, no. 4, pp. 335–359, 2007.

[13] J. Vial, H. Durrant-Whyte, and T. Bailey, "Conservative sparsification for efficient and consistent approximate estimation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2011, pp. 886–893.

[14] F. Dellaert and M. Kaess, "Square root SAM: Simultaneous location and mapping via square root information smoothing," *Int. J. Robot. Res.*, vol. 25, no. 12, pp. 1181–1203, 2006.

[15] M. Kaess, A. Ranganathan, and F. Dellaert, "iSAM: Incremental smoothing and mapping," *IEEE Trans. Robot.*, vol. 24, no. 6, pp. 1365–1378, Dec. 2008.

[16] L. Paull, G. Huang, and J. J. Leonard, "A unified resource-constrained framework for graph SLAM," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2016, pp. 1346–1353.

[17] A. Krause, A. Singh, and C. Guestrin, "Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies," *J. Mach. Learning Res.*, vol. 9, pp. 235–284, Jun. 2008.

[18] M. Beinhofer, J. Mller, and W. Burgard, "Effective landmark placement for accurate and reliable mobile robot navigation," *Robot. Autonomous Syst.*, vol. 61, no. 10, pp. 1060–1069, 2013.

[19] M. Dymczyk, S. Lynen, M. Bosse, and R. Siegwart, "Keep it brief: Scalable creation of compressed localization maps," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2015, pp. 2536–2542.

[20] S. Frintrop and P. Jensfelt, "Attentional landmarks and active gaze control for visual SLAM," *IEEE Trans. Robot.*, vol. 24, no. 5, pp. 1054–1065, Oct. 2008.

[21] A. Kim and R. Eustice, "Active visual SLAM for robotic area coverage: Theory and experiment," *Int. J. Robot. Res.*, vol. 34, no. 45, pp. 457–475, Apr. 2015. [Online]. Available: http://dx.doi.org/10.1177/0278364914547893

[22] S. Hochdorfer and C. Schlegel, "Landmark rating and selection according to localization coverage: Addressing the challenge of lifelong operation of SLAM in service robots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2009, pp. 382–387.

[23] G. Dissanayake, H. Durrant-Whyte, and T. Bailey, "A computationally efficient solution to the simultaneous localisation and map building (SLAM) problem," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2000, vol. 2, pp. 1009–1014.

[24] S. Zhang, L. Xie, and M. Adams, "Entropy based feature selection scheme for real time simultaneous localization and map building," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Aug. 2005, pp. 1175–1180.

[25] H. Strasdat, C. Stachniss, and W. Burgard, "Which landmark is useful? Learning selection policies for navigation in unknown environments," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2009, pp. 1410–1415.

[26] R. Lerner, E. Rivlin, and I. Shimshoni, "Landmark selection for task-oriented navigation," *IEEE Trans. Robot.*, vol. 23, no. 3, pp. 494–505, Jun. 2007.

[27] P. Sala, R. Sim, A. Shokoufandeh, and S. Dickinson, "Landmark selection for vision-based navigation," *IEEE Trans. Robot.*, vol. 22, no. 2, pp. 334–349, Apr. 2006.

[28] Y. Huang and K. Gupta, "Collision-probability constrained PRM for a manipulator with base pose uncertainty," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2009, pp. 1426–1432.

[29] B. Luders, "Robust sampling-based motion planning for autonomous vehicles in uncertain environments," Ph.D. dissertation, Dept. of Aeronautics Astronautics, Massachusetts Inst. of Technol., Cambridge, MA, USA, May 2014.

[30] L. Blackmore, M. Ono, and B. Williams, "Chance-constrained optimal path planning with obstacles," *IEEE Trans. Robot.*, vol. 27, no. 6, pp. 1080–1094, Dec. 2011.

[31] A. Agha-mohammadi, S. Chakravorty, and N. Amato, "FIRM: Sampling-based feedback motion planning under motion uncertainty and imperfect measurements," *Int. J. Robot. Res.*, vol. 33, no. 2, pp. 268–304, 2014.

[32] S. Prentice and N. Roy, "The belief roadmap: Efficient planning in belief space by factoring the covariance," *Int. J. Robot. Res.*, vol. 28, nos. 11/12, pp. 1448–1465, 2009.

[33] H. Kurniawati, T. Bandyopadhyay, and N. Patrikalakis, "Global motion planning under uncertain motion, sensing, and environment map," *Auton. Robots*, vol. 33, no. 3, pp. 255–272, 2012.

[34] A. Lambert and N. L. Fort-Piat, "Safe task planning integrating uncertainties and local maps federations," *Int. J. Robot. Res.*, vol. 19, no. 6, pp. 597–611, 2000.

[35] D. Rosen, G. Huang, and J. Leonard, "Inference over heterogeneous finite-/infinite-dimensional systems using factor graphs and Gaussian processes," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2014, pp. 1261–1268.

[36] D. J. C. MacKay, *Information Theory, Inference & Learning Algorithms*. New York, NY, USA: Cambridge Univ. Press, 2002.

[37] N. D. Toit and J. Burdick, "Robot motion planning in dynamic, uncertain environments," *IEEE Trans. Robot.*, vol. 28, no. 1, pp. 101–115, Feb. 2012.

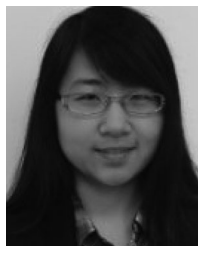
[38] J. V. D. Berg, S. Patil, and R. Alterovitz, "Motion planning under uncertainty using iterative local optimization in belief space," *Int. J. Robot. Res.*, vol. 31, no. 11, pp. 1263–1278, 2012.

[39] D. Bertsekas, *Dynamic Programming and Optimal Control*, vol. III. Belmont, MA, USA: Athena Sci., 2007.

[40] D. M. Rosen, M. Kaess, and J. J. Leonard, "RISE: An incremental trust-region method for robust online sparse least-squares estimation," *IEEE Trans. Robotics*, vol. 30, no. 5, pp. 1091–1108, Oct. 2014.

[41] H. Johannsson, M. Kaess, M. Fallon, and J. Leonard, "Temporally scalable visual SLAM using a reduced pose graph," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2013, pp. 54–61.

[42] E. Olson, "AprilTag: A robust and flexible visual fiducial system," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2011, pp. 3400–3407



Beipeng Mu (M'16) received the B.Eng. degree in automation from Tsinghua University, Beijing, China in 2011, M.S. and Ph.D. degrees in aeronautics and astronautics from Massachusetts Institute of Technology, Cambridge, MA, USA, in 2013 and 2016, respectively.

She is a Research Scientist with Oculus Research, Pittsburgh, PA, USA. Her research interest includes autonomous navigation and mapping for mobile robots, information extraction from complex data, and statistical inference.

Liam Paull (M'14) received the B.Sc. degree in computer engineering from McGill University, Montréal, QC, Canada, in 2004 and the Ph.D. degree in electrical and computer engineering from University of New Brunswick, Fredericton, NB, USA, in 2013.

He is a Research Scientist in the Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA, USA. His research interests include robust inference and control for resource-constrained and safety-critical robotic systems including both marine and autonomous vehicle applications.

Ali-Akbar Agha-Mohammadi received the B.S. in electrical and computer engineering from Tabriz University, East Azerbaijan Province, Iran, and the M.S. in electrical engineering (control systems) from Khaje Nasir Toosi University of Technology, Tehran Province, Iran, in 2005 and 2008, repectively.

He is a Robotics Research Technologist with NASA JPL/California Institute of Technology, Pasadena, CA, USA. Previously, he was a Research Engineer with Qualcomm Research and a Postdoctoral Researcher with Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA, USA. His research interests include robotics, stochastic systems, control systems, estimation, and filtering theory.

John J. Leonard (F'14) received the B.S.E.E. degree in electrical engineering and science from University of Pennsylvania, Philadelphia, PA, USA, in 1987 and the D.Phil. degree in engineering science from University of Oxford, Oxford, U.K., in 1994.

He is a Samuel C. Collins Professor of Mechanical and Ocean Engineering. His research addresses the problems of navigation and mapping for autonomous mobile robots. He joined the Massachusetts Institute of Technology (MIT), Cambridge, MA, USA, faculty in 1996, after five years as a Postdoctoral Fellow and Research Scientist in the MIT Sea Grant Autonomous Underwater Vehicle Laboratory. He was a Team Leader for MIT's DARPA Urban Challenge team, which was one of 11 teams to qualify for the Urban Challenge final event and one of six teams to complete the race. He served as a Co-Director of the Ford–MIT Alliance from 2009 to 2013.

Prof. Leonard received a National Science Foundation Career Award in 1998 and the King-Sun Fu Memorial Best Transactions on Robotics Paper Award in 2006.

Jonathan P. How (SM'05) received the B.A.Sc. degree from University of Toronto, Toronto, ON, Canada, in 1987, and the S.M. and Ph.D. degrees in aeronautics and astronautics from Massachusetts Institute of Technology (MIT), Cambridge, MA, USA, in 1990 and 1993.

He is the Richard Maclaurin Professor of Aeronautics and Astronautics with MIT. Prior to joining MIT in 2000, he was an Assistant Professor with Stanford University, Stanford, CA, USA. His research interests include robust coordination and control of autonomous vehicles. He is a Fellow of the AIAA.