FIRM: Feedback Controller-Based Information-State Roadmap
– A Framework for Motion Planning Under Uncertainty –

Ali-akbar Agha-mohammadi, Suman Chakravorty, Nancy M. Amato

Abstract— Direct transformation of sampling-based motion planning methods to the Information-state (belief) space is a challenge. The main bottleneck for roadmap-based techniques in belief space is that the incurred costs on different edges of the graph are not independent of each other. In this paper, we generalize the Probabilistic RoadMap (PRM) framework to obtain a Feedback controller-based Information-state RoadMap (FIRM) that takes into account motion and sensing uncertainty in planning. The FIRM nodes and edges lie in belief space and the crucial feature of FIRM is that the costs associated with different edges of FIRM are independent of each other. Therefore, this construct essentially breaks the "curse of history" in the original Partially Observable Markov Decision Process (POMDP), which models the planning problem. Further, we show how obstacles can be rigorously incorporated into planning on FIRM. All these properties stem from utilizing feedback controllers in the construction of FIRM.

I. INTRODUCTION

Sampling-based path planning algorithms such as Probabilistic Roadmap (PRM) [1] methods, Rapidly exploring Randomized Trees (RRT) [2], and their variants have shown great success in solving robot motion planning problems in the absence of uncertainty. However, direct transformation of these methods to planning under uncertainty is a challenge. The first issue is ensuring that the roadmap nodes are reachable. The second challenge is that the incurred costs on different edges of the roadmap depend on each other; this violates an assumption in roadmap based methods that each roadmap edge represents an independent planning problem.

In this paper, we generalize the PRM framework to the Feedback controller-based Information-state RoadMap (FIRM) that takes into account both motion and sensing uncertainties. The probability density function (pdf) over the state is called *belief* or *information-state*. The FIRM is constructed as a roadmap in belief space, where its nodes are small subsets of belief space and the edges of FIRM are Markov chains in belief space. It is the first method that generalizes the PRM to belief space in such a way that the incurred costs on different edges of roadmap are independent of each other, while still providing a straightforward approach to sample reachable nodes in belief space. These properties are a direct consequence of utilizing feedback
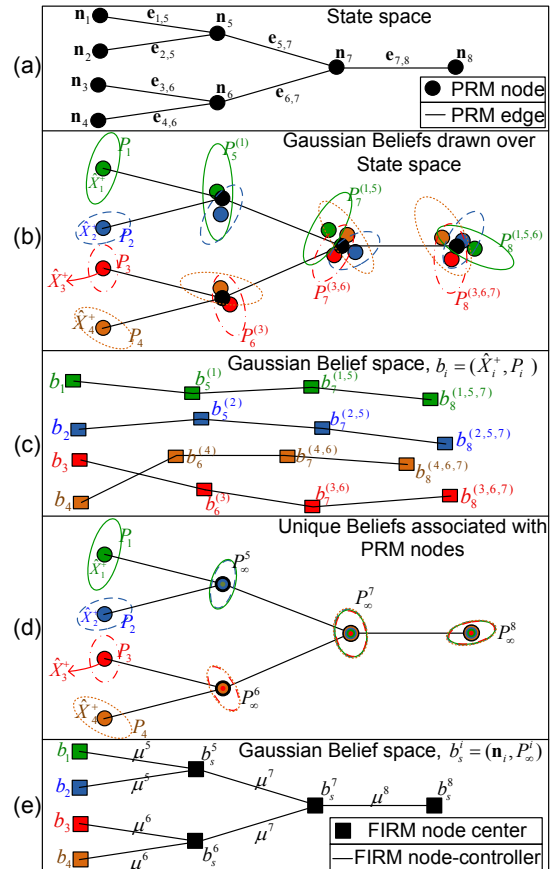
Fig. 1. (a) A simple PRM in state-space. (b) The belief evolution along different paths that lead to the $\mathbf{n}_8$. As it is seen the belief depends on the traveled path by robot. For example $P_8^{(3,6,7)}$ denotes the estimation covariance at node $\mathbf{n}_8$, when the robot has traversed a path through nodes $(3, 6, 7)$ to reach node 8. (c) Corresponding belief paths in the belief space. Belief at each node depends on the history of belief evolution. (d) Unique beliefs assigned to each PRM node. Regardless of the belief history, the belief at each node stops at these predefined beliefs. (e) The FIRM corresponding to the given PRM. Note that μ^j's are not FIRM edges. Indeed, (i, j)-th FIRM edge is generated by the action of μ^j on b_s^i.

controllers in the construction of FIRM. Planning under motion and sensing uncertainty is essentially a Partially Observable Markov Decision Process (POMDP). An important contribution of FIRM is that it breaks the curse of history in such POMDPs and provides the optimal policy over the roadmap instead of only a nominal path.

Figure 1 illustrates the curse of history in POMDPs and the approach of FIRM in breaking it. Although there exists a single edge $e_{(7,8)}$ between nodes $\mathbf{n}_7$ and $\mathbf{n}_8$ in PRM (cf. Fig. 1(a)), the belief evolution along $e_{(7,8)}$ is not unique (cf. Fig. 1(c)) and it depends on the path, which has led to $\mathbf{n}_7$.

Moreover, even if we assume there is only one path that leads to $\mathbf{n}_7$, every time the robot traverses this path, due to the randomness in measurements, the belief will end up at a different value on $\mathbf{n}_7$, and therefore the belief-dependent cost on edge $e_{(7,8)}$ is not predictable without having the full knowledge of the belief at $\mathbf{n}_7$ or equivalently the full knowledge of history of observations before reaching $\mathbf{n}_7$.

In FIRM, using node-controllers and appropriate stopping conditions, we stop the belief evolution at predefined unique beliefs associated with PRM nodes (cf. Fig. 1(d)). Doing so, we break the curse of history by making the after-node belief evolution independent of before-node belief evolution. Thus, we can construct the PRM-like roadmap in belief space, i.e., FIRM, with independent edge costs.

Without edge independence, the incorporation of obstacles in planning on roadmaps is a challenge because it either entails costly repeated computations of collision probabilities, or some collision measure has to be designed that in general cannot capture the true collision probabilities and may lead to overly conservative plans. However, in FIRM, owing to the independence of edges, we can compute the collision probabilities offline and incorporate obstacles in planning over FIRM that leads to more reliable and less conservative plans.

In the next section, we review the most relevant related work. Section III provides an overview of the method and its contributions. In Section IV, we derive FIRM MDP as a computationally tractable approximation of POMDP. In Section V, we detail how the assumptions inherent in FIRM can be satisfied and construct a FIRM. Experimental results are presented in Section VI.

II. RELATED WORK

In recent years, there has been a concerted effort to incorporate uncertainty into sampling-based motion planning methods. A class of these methods deal with map uncertainty, such as [3]–[5], while the methods in [6]–[8] deal with motion uncertainty. Another class of methods that are most related to FIRM consider both motion and sensing uncertainties in planning, such as [9]–[15]. In the following, we briefly discuss the planning approaches in these references and place them in context with FIRM.

Censi *et al.* [9] propose a planning algorithm based on graph search and constraint propagation on a grid-based representation of the space. In the LQG-MP method of Van den Berg *et al.* [10], the best path is found among the finite number of RRT paths by simulating the performance of LQG on all of them. Platt *et al.* in [11] plan in continuous space by finding the best nominal path through nonlinear optimization methods. Prentice *et al.* [12] and Huynh *et al.* [13] propose PRM-based approaches, where the best path is found through breadth-first search on the Belief roadmap.

In all these methods, a best nominal path is computed offline. This nominal path is fixed regardless of the process and sensing noises in the execution phase. [11] performs replanning when large deviations happen, which is a computationally expensive procedure since all the costs along the path have to be reproduced for the new initial belief. In FIRM, however, the best feedback policy, i.e., a mapping from belief space to actions, is computed offline, which is a main goal of planning under motion and sensing uncertainty (POMDPs). In the method proposed by Toit *et al.* [14], the nominal path is updated dynamically in a receding horizon control approach, which entails repeatedly solving open loop optimal control problems at every time step. Kurniawati *et al.* [15] compute the value function at sampled milestones and thus compute the optimal policy rather than the optimal nominal controls.

In the methods that account for sensing uncertainty, the state has to be estimated based on measurements. To handle unknown future measurements in the planning stage, methods [9], [11]–[14] consider only the maximum likelihood (ML) observation sequence to predict the estimation performance. In contrast, FIRM takes all possible future observations into account in planning.

In the presence of obstacles, due to the dependency of collision events in different time steps, it is a burdensome task to include the collision probabilities in planning. That is why the methods such as [9], [10], [14] design some safety measures to account for obstacles in planning. However, in FIRM, collision probabilities can be computed and seamlessly incorporated in planning stage.

III. METHOD OVERVIEW AND CONTRIBUTIONS

The FIRM graph is a generalization of the PRM graph, whose nodes are small subsets of belief space and whose edges are Markov chains induced by feedback controllers. As a result, planning on FIRM is a Markov Decision Process (MDP), referred to as FIRM MDP here. FIRM MDP is defined on FIRM nodes, and thus it can be solved using standard Dynamic Programming (DP) techniques [16].

Inducing reachable belief nodes: FIRM samples nodes in the robot state space and then utilizes feedback controllers to automatically induce unique beliefs associated with each of these state space nodes (cf. Fig. 1). The controller can drive the belief into the neighborhood of these belief states in finite time and thus ensures reachability. This way FIRM addresses the hard task of sampling in reachable belief space that is usually required in belief space planning [15], [17], [18].

Breaking the curse of history: A fundamental contribution of FIRM is that the optimal action, at a given node, does not depend on the traversed nodes, actions, and observations prior to this node, i.e., it is independent of the history of the information process (cf. Fig. 1). This is a direct consequence of inducing reachable belief nodes using the feedback controllers, which essentially breaks the curse of history in POMDPs. In addition, the sampling-based nature of the method borrowed from PRM allows us to ameliorate the curse of dimensionality.

Efficient planning: The construction of FIRM is offline and thus online planning (and replanning) is feasible. Moreover, in FIRM, the optimal feedback policy, instead of the nominal path, is computed offline. This is done by solving the

dynamic programming problem associated with the FIRM MDP on belief nodes induced by the feedback controllers.

Incorporating obstacles in planning: In the FIRM framework, the collision probabilities can be computed, which leads to more accurate plans, as opposed to a simplified collision measure, that may lead to conservative plans. The obstacles induce a *failure node* in the FIRM MDP, into which the robot can be absorbed. Further, due to the offline construction of FIRM, the heavy computational burden of estimating collision probabilities can be done offline.

The generic algorithm for offline construction of FIRM is presented in Algorithm 1.

Algorithm 1: Generic Construction of FIRM (Offline)

---

1 Construct a PRM with nodes $\{\mathbf{n}_j\}$ and edges $\{e_{ij}\}$;
2 For each PRM node $\mathbf{n}_j$, design a controller μ^j and reachable FIRM node B_j;
3 For each FIRM node B_i, characterize the allowed set of controllers $A(i)$;
4 For each B_i and $\mu^j \in A(i)$, compute the cost, collision probabilities and transition probabilities associated with going from B_i to B_j;
5 Solve the FIRM MDP to compute feedback π over FIRM nodes.

The generic algorithm for online planning on FIRM is presented in Algorithm 2.

Algorithm 2: Generic planning on FIRM (Online)

---

1 Given an initial belief, invoke some controller $\mu^i(\cdot)$ in FIRM to absorb the robot into FIRM node B_i;
2 Given the system is in set B_i, invoke the higher level feedback policy π to choose the lower level feedback controller $\mu^j(\cdot)$ where $j = \pi(B_i)$;
3 Let the node-controller $\mu^j(\cdot)$ execute untill absorption into the B_j or failure;
4 Repeat steps 2-3 untill absorption into the goal node B_{goal} or failure.

The concrete instantiations of these generic algorithms are given in section V.

IV. THE POMDP TO FIRM MDP TRANSFORMATION

In this section, we detail how to transform a POMDP into a FIRM MDP. In the first subsection, the POMDP problem is briefly outlined. In subsection B, we develop the transformation for the obstacle-free case, and in subsection C, we show how to incorporate obstacles into the planning.

A. Preliminaries

Consider a controlled hidden Markov model with hidden state $X \in \mathbb{X}$, control $u \in \mathbb{U}$, and observation $Z \in \mathbb{Z}$, with transition probability model $p(X'|X, u)$ and observation model $p(Z|X)$. Let $Z_{0:k}$ denote the set of observations untill time k. Then, the information-state (belief) b_k of the system at time k, is defined as the probability distribution of the underlying system state X given $Z_{0:k}$, i.e., $b_k(X) = p(X|Z_{0:k})$. Let the space of all such beliefs be denoted

by the belief space $\mathbb{B}$. It is well known that the infinite horizon POMDP problem can be cast as an MDP problem in belief space, called belief MDP problem here, whose solution is obtained by solving the following stationary Dynamic Programming (DP) equation on the belief space $\mathbb{B}$ [16], [19]:

$$J(b) = \min_u \{c(b, u) + \int_{\mathbb{B}} p(b'|b, u) J(b') db'\}, \forall b \in \mathbb{B}, \quad (1)$$

where $c(b, u)$ is the incremental cost of taking action u at belief state b, $J(b)$ is the optimal cost-to-go from belief state b, and $p(b'|b, u)$ represents the transition probability density over belief states, given that control u is taken at belief state b. This transition probability can be derived using Bayes rule and the law of total probability [16], [19]. However, as is well known, the above DP equation is exceedingly difficult to solve since it is defined over whole belief space, and suffers from the curse of history. In the following, we show how the POMDP can be reduced to a computationally tractable MDP over FIRM nodes.

B. Obstacle-free FIRM

Let us consider the DP in Eq.(1), through which the cost-to-go function can be computed for the belief MDP problem. We further restrict the problem's scope by the following assumption. Consider a set of sampled nodes in the state space of the robot $\{\mathbf{n}_i\}_{i=1}^N$ that includes the goal node $\mathbf{n}_{goal}$, into whose vicinity we want to transfer the robot.

Assumption 1: We assume that corresponding to every state node $\mathbf{n}_i$, there exists a unique stopping belief b_s^i and an associated feedback controller $u = \mu^i(b)$ such that the controller can drive the belief state into B_i in finite time with probability one, where B_i is a small neighborhood of b_s^i. B_i is referred to as the i-th stopping region or i-th belief node.

The feedback controller μ^i is a stationary controller, called the i-th node-controller. Assumption 1 is satisfied if μ^i is a *proper policy* [16]. Under the node-controller $\mu^i(b)$, the belief evolves according to a Markov chain whose transition density function is denoted by $p^{\mu^i}(b'|b) := p(b'|b, \mu^i(b))$. Thus, the node-controller essentially induces the Markov chain $p^{\mu^i}(b'|b)$ over the belief space $\mathbb{B}$. Controller μ^i is proper iff in the Markov chain $p^{\mu^i}(b'|b)$, each belief b is connected to the B_i with a path of positive probability transitions [16]. Therefore, irreducibility of the chain $p^{\mu^i}(b'|b)$ is the sufficient condition for μ^i to be proper. Irreducibility essentially implies that the Markov chain can go from any point in the belief space to any non-zero measure set in the belief space in finite time with probability one. In the next section, we discuss how such a node-controller can be constructed.

Planning goal: The planning goal is to transfer the robot into some pre-specified region B_{goal} corresponding to the goal node $\mathbf{n}_{goal}$, with probability one, following which the system can remain there without incurring any further cost. This is also known as a stochastic shortest path problem [16].

Assumption 2: It is assumed that the belief process induced by μ^i can stop iff it enters the node B_i. Further, once the system is in the node B_i, it is allowed to invoke one of

the controllers $\mu^j(\cdot)$ among $j \in A(i)$, *the k-nearest neighbor set of i, that in turn will draw the system into the region B_j.*

Based on these assumptions, the original MDP in belief space, formulated using DP in Eq.(1), is now turned into a Semi-Markov Decision Process (SMDP) [20] on the belief space or equivalently an MDP on the continuous regions B_i. We call this restricted form of the original POMDP, the FIRM MDP, whose DP formulation is:

$$J(b) = \min_{j \in A(i)} C^{\mu^j}(b) + \int_{B_j} p^{\mu^j}(b'|b)J(b')db', \forall b \in B_i, \forall i. \quad (2)$$

In the equation above, $C^{\mu^j}(b)$ represents the expected cost of invoking node-controller $\mu^j(\cdot)$ starting at belief state b till the node-controller stops executing. Mathematically:

$$C^{\mu^j}(b) = \sum_{t=0}^{\mathcal{T}} c(b_t, \mu^j(b_t)|b_0 = b), \quad (3)$$

$$\mathcal{T}^{\mu^j}(b) = \min_t \{t|b_t \in B_j, b_0 = b\}. \quad (4)$$

where $\mathcal{T}^{\mu^j}$, which is a function of initial belief, is a random stopping time denoting the time at which the belief state enters the node B_j under the controller μ^j. The pdf $p^{\mu^j}(b'|b)$ represents the belief transition pdf given that μ^j is invoked at b.

The FIRM MDP, though computationally more tractable than the original POMDP, is defined on the continuous neighborhoods B_i and thus, still formidable to solve. Instead, let us consider the following piecewise constant approximation:

$$J(b) \approx J(b_s^i), \quad C^{\mu^j}(b) \approx C^{\mu^j}(b_s^i), \quad \forall b \in B_i, \forall i. \quad (5)$$

Given the above approximation, FIRM MDP in Eq.(2) can be approximated as follows:

$$J(b_s^i) = \min_{j \in A(i)} C^{\mu^j}(b_s^i) + P^{\mu^j}(B_j|b_s^i)J(b_s^j), \forall i, \quad (6)$$

where, $P^{\mu^j}(B_j|b_s^i)$ represents the probability that the controller μ^j invoked at b_s^i takes b_s^i into the B_j. Note that in the absence of obstacles assumption 1 implies that $P^{\mu^j}(B_j|b_s^i) = 1$.

Equation Eq.(6) is an arbitrarily accurate approximation to the original FIRM MDP in Eq.(2) given that the functions $C^{\mu^j}(\cdot)$ and $P^{\mu^j}(\cdot|\cdot)$ are smooth with respect to their arguments (i.e., at least continuous), and given that the belief nodes B_i are sufficiently small. The approximation essentially states that any belief in the region B_i is represented by b_s^i for the purpose of decision making. Abusing the notation and defining $J(B_i) := J(b_s^i)$, $C^{\mu^j}(B_i) := C^{\mu^j}(b_s^i)$, and $P^{\mu^j}(\cdot|B_i) := P^{\mu^j}(\cdot|b_s^i)$ leads to the equation:

$$J(B_i) = \min_{j \in A(i)} C^{\mu^j}(B_i) + P^{\mu^j}(B_j|B_i)J(B_j)$$
$$= \min_{j \in A(i)} C^{\mu^j}(B_i) + J(B_j), \forall i \quad (7a)$$

$$j^* = \pi(B_i) = \arg \min_{j \in A(i)} C^{\mu^j}(B_i) + J(B_j), \forall i. \quad (7b)$$

Thus, the original POMDP becomes a finite N-state MDP in Eq.(7) defined on the abstract "belief nodes" $\{B_i\}_{i=1}^N$.

Given $C^{\mu^j}(\cdot)$ and $P^{\mu^j}(\cdot|\cdot)$, this problem can easily be solved using standard DP techniques such as value/policy iteration to yield a feedback policy π on the higher level embedded MDP defined on the belief nodes B_i. Given that the system stops in node B_i, this policy determines which node-controller μ^{j^*} has to be invoked, where $j^* = \pi(B_i)$. In order to solve Eq.(7), the generalized costs $C^{\mu^j}(\cdot)$ and transition probabilities $P^{\mu^j}(\cdot|\cdot)$ need to be evaluated. We discuss how to compute these in Section IV on FIRM construction.

C. Incorporating Obstacles into FIRM

In the presence of obstacles, we can never assure that the node-controller $\mu^j(\cdot)$ can drive any $b \in B_i$ into B_j with probability one. Instead, we have to specify the failure probabilities that the robot collides with an obstacle. Let us denote the failure set on $\mathbb{X}$ by F (i.e., $F = \mathbb{X} - \mathbb{X}_{free}$). Now, let $P^{\mu^j}(F|b)$ denote the probability that under node-controller μ^j the system enters the failure set F before it enters the region B_j, given that the initial belief is b. Again, for smooth transition pdf's and given that the sets B_j are suitably small, and abusing the notation to $P^{\mu^j}(\cdot|B_i) := P^{\mu^j}(\cdot|b_s^i)$, we can modify Eq.(7) to incorporate obstacles in the state space:

$$J(B_i) = \min_j C^{\mu^j}(B_i) + J(F)P^{\mu^j}(F|B_i)$$
$$+ J(B_j)P^{\mu^j}(B_j, \overline{F}|B_i), \quad (8a)$$
$$j^* = \pi(B_i) = \arg \min_j C^{\mu^j}(B_i) + J(F)P^{\mu^j}(F|B_i)$$
$$+ J(B_j)P^{\mu^j}(B_j, \overline{F}|B_i), \quad (8b)$$

where $P^{\mu^j}(B_j, \overline{F}|B_i)$ denotes the probability of reaching B_j, under controller μ^j invoked at B_i, before hitting an obstacle. $J(F)$ is a user-defined suitably high cost-to-go value for failure. It is assumed that the system can enter the goal region or the failure set and remain there subsequently without incurring any additional cost. Thus, all that is required to solve the above DP equation are the values of the costs $C^{\mu^j}(B_i)$ and transition probability functions $P^{\mu^j}(B_j, \overline{F}|B_i)$ and $P^{\mu^j}(F|B_i)$. Thus, the main difference from the obstacle free case is the addition of a "failure" state to the FIRM MDP along with the associated probabilities of failure from the various nodes B_i.

We would also like to quantify the quality of the solution that is obtained by the FIRM. To this end, we require the probability of success of a policy π at the higher level Markov chain on B_i's given by Eq.(8b). The FIRM MDP now has $N + 1$ states $\{S_1, S_2, \cdots, S_{N+1}\}$ that can be decomposed into three disjoint classes: the goal class $S_1 = B_{goal}$, the failure class $S_2 = F$, and the transient class $\{S_3, S_4, \cdots, S_{N+1}\} = \{B_1, B_2, \cdots, B_N\} \setminus B_{goal}$. The goal and failure classes are recurrent classes of this Markov chain. As a result, the transition probability matrix of this higher level $N+1$ state Markov can be decomposed as follows [21]:

$$\mathcal{P} = \begin{bmatrix} \mathcal{P}_g & 0 & 0 \\ 0 & \mathcal{P}_f & 0 \\ \mathcal{R}_g & \mathcal{R}_f & \mathcal{Q} \end{bmatrix}. \quad (9)$$

The (i, j)-th component of $\mathcal{P}$ represents the transition probability from S_j to S_i. Moreover $\mathcal{P}_g = 1$ and $\mathcal{P}_f = 1$, since goal and failure classes are recurrent classes, i.e., the system stops once it reaches the goal or it fails. $\mathcal{Q}$ is a matrix that represents the transition probabilities between belief nodes B_i in transient class. $\mathcal{R}_g$ and $\mathcal{R}_f$ are $(N-1) \times 1$ vectors that represent the probability of the transient nodes $\{B_i\} \setminus B_g$ getting absorbed into the goal node and the failure set, respectively. Then, it can be shown that the success probability from any desired node B_i is given by the i-th component of the vector $\mathcal{P}^s$, denoted by $\mathcal{P}_i^s$ [21]:

$$\Pr(\text{success}|B_i) = \mathcal{P}_i^s, \quad \mathcal{P}^s = (I - \mathcal{Q})^{-1} \mathcal{R}_g. \quad (10)$$

Thus, given that we can suitably construct the node-controllers $\mu^i(\cdot)$, the sets B_i, evaluate the transition costs $C^{\mu^i}(\cdot)$ and the transition probabilities $P^{\mu^i}(\cdot|\cdot)$, we can transform the POMDP into a FIRM MDP.

V. FIRM CONSTRUCTION

In this section, we address how the four elements in the FIRM, i.e., nodes B_i, node-controllers μ^i, transition probabilities $P^{\mu^j}(\cdot|B_i)$, and costs $C^{\mu^j}(B_i)$ can be constructed such that the necessary assumptions in section IV are satisfied.

A. FIRM Nodes B_i and Control Policies μ^j

PRM samples its nodes $\{\mathbf{n}_j\}_{j=1}^N$ from $\mathbb{X}_{free}$ based on some appropriate probabilistic measure [1]. Similarly, in planning in belief space it is desired to sample the belief space, where the main problem is whether the sampled belief is reachable or not. In general, characterizing the whole reachable region of $\mathbb{B}$ is computationally infeasible. A main contribution of FIRM is that instead of sampling in belief space and characterizing if the sampled belief is reachable or not, FIRM exploits node-controllers to induce reachable regions in belief space $\mathbb{B}$ as is explained in the following.

The initial sampling in FIRM is done in the state space using PRM techniques. After sampling PRM nodes $\{\mathbf{n}_j\}_{j=1}^N$ in $\mathbb{X}_{free}$, for each PRM node $\mathbf{n}_j$, we associate a node-controller μ^j and FIRM node $B_j \subset \mathbb{B}$ that satisfy the assumption 1. In the following we restrict our approach to the linear models and nonlinear models that are locally well approximated by the linearization. We also assume that both process and measurement noises are drawn from zero-mean Gaussian distributions. Suppose the system (linearized at $\mathbf{n}_j$) has the state-space form:

$$X_{k+1} = \mathbf{A}^j X_k + \mathbf{B}^j u_k + \mathbf{G}^j W_k, \quad W_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}^j) \quad (11)$$
$$Z_k = \mathbf{H}^j X_k + V_k, \quad\quad\quad V_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}^j). \quad (12)$$

where, W_k and V_k are motion and measurement noises, respectively, drawn from zero-mean Gaussian distributions with covariances $\mathbf{Q}^j$ and $\mathbf{R}^j$.

Node controller: We choose the node-controller μ^j as the stationary Linear Quadratic Gaussian (LQG) controller, designed for the linearized system at $\mathbf{n}_j$ [16]. Under the Gaussian assumption, the belief is characterized by a pair consisting of estimation mean and covariance $b_k = (\widehat{X}_k^+, P_k)$. We

denote the governing dynamics of belief under LQG by $\mathbf{f}_b$, which indeed encapsulates Kalman filtering equations:

$$b_{k+1} = \mathbf{f}_b(b_k, u_k, Z_k), \quad u_k = \mu^j(b_k). \quad (13)$$

From control theory it can be shown that if the pair $(\mathbf{A}^j, \mathbf{B}^j)$ is controllable and the pair $(\mathbf{A}^j, \mathbf{H}^j)$ is observable, then the belief chain in Eq.(13) under μ^j is ergodic, i.e., $\lim_{k \to \infty} b_k = b_\infty = (\widehat{X}_\infty^{+j}, P_\infty^j)$, (see [22] for details). Actually, the dynamics of the estimation covariance P_k is deterministic and it converges to the deterministic covariance P_∞^j [16]. Covariance P_∞^j is computed as $P_\infty^j = (I - \mathbf{L}^j \mathbf{H}^j) P_\infty^{j-}$, where P_∞^{j-} is the solution following Discrete Algebraic Riccati Equation (DARE) within the class of positive semidefinite symmetric matrices.

$$P_\infty^{j-} = \mathbf{G}^j \mathbf{Q}^j \mathbf{G}^{j^T} \quad (14)$$
$$+ \mathbf{A}^j (P_\infty^{j-} - P_\infty^{j-} \mathbf{H}^{j^T} (\mathbf{H}^j P_\infty^{j-} \mathbf{H}^{j^T} + \mathbf{R}^j)^{-1} \mathbf{H}^j P_\infty^{j-}) \mathbf{A}^{j^T},$$
$$\mathbf{L}^j = P_\infty^{j-} \mathbf{H}^{j^T} (\mathbf{H}^j P_\infty^{j-} \mathbf{H}^{j^T} + \mathbf{R}^j)^{-1}. \quad (15)$$

The dynamics of estimation mean $\widehat{X}_k^+$ is random and it can be shown that it converges to a stationary random vector $\widehat{X}_\infty^{+j}$, whose mean is $\mathbf{n}_j = \mathbb{E}[\widehat{X}_\infty^{+j}]$, (see [22] for details).

FIRM node: We define the unique stopping belief b_s^j associated with $\mathbf{n}_j$ as the mean of the stationary belief b_∞^j, i.e., $b_s^j = \mathbb{E}[b_\infty^j]$, which is equal to the following pair:

$$b_s^j = (\mathbf{n}_j, P_\infty^j). \quad (16)$$

According to the irreducible belief chain induced by μ^j, the probability of absorbtion into any nonzero-measure set in belief space centered at b_s^j is one. Therefore, Assumption 1 is satisfied by defining the j-th FIRM node $B_j \subset \mathbb{B}$ as a region centered at b_s^j.

$$B_j = \{b = (X, P) \mid \|X - \mathbf{n}_j\| < \epsilon, \|P - P_\infty^j\| < \delta\}, \quad (17)$$

where, ϵ and δ are suitably small thresholds that determine the FIRM node size B_j.

Transition probabilities and edge costs: Computing transition probabilities $P^{\mu^j}(\cdot|B_i)$, and costs $C^{\mu^j}(B_i)$ associated with invoking node controller μ^j at node B_i, in general can be a computationally expensive tasks. Here, we utilize the Monte Carlo-based (MC-based) methods to approximate the collision probabilities. The dependency of collision events in different time steps, which is ignored in most collision probability computing methods in the POMDP literature, can be taken into account rigorously in MC-based methods. An MC-based approximation can reach any desired accuracy by increasing the number of particles M. However, the main problem of MC-based methods is their high computational cost, which might preclude their use in online scenarios. Nevertheless, owing to the offline construction of FIRM, the high computational burden of MC-based approaches can be tolerated. The method is detailed in [22].

Depending on the application, one can define a variety of cost functions for taking node-controller μ^j at B_i. Here, we first consider estimation accuracy to find the paths, on which the estimator and accordingly controller can perform better.

A measure of estimation error is the trace of estimation covariance. Thus, we use $\Phi = \mathbb{E}[\sum_{k=1}^{\mathcal{T}} \text{tr}(P_k)]$. In stationary LQG, the covariance matrix evolves deterministically and thus the expectation operator can be omitted. However, if the filter of choice is the Extended Kalman Filter (EKF), the covariance matrix evolution is stochastic and this measure can take into account its stochasticity. Moreover, as we are also interested in faster paths, we take into account the corresponding mean stopping time, i.e., $\widehat{\mathcal{T}} = \mathbb{E}[\mathcal{T}]$, and the total cost of invoking μ^j at B_i is considered as a linear combination of estimation accuracy and expected stopping time, with suitable coefficients α_1 and α_2.

$$C^{\mu^j}(B_i) = \alpha_1 \Phi + \alpha_2 \widehat{\mathcal{T}}. \tag{18}$$

B. Offline Construction of FIRM

The crucial feature of FIRM is that it can be constructed offline and stored, independent of future queries. Moreover, owing to the reduction from the original POMDP to an N-state MDP on belief nodes, the FIRM MDP can be solved using standard DP techniques such as value/policy iteration to yield the optimal policy $j^* = \pi(B_i)$ on the FIRM MDP defined on the belief nodes. Indeed, at each belief $b \in B_i$, the policy $j^* = \pi(B_i)$ decides which node-controller μ^{j^*} has to be invoked among $j \in A(i)$. Algorithm 3 details the construction of FIRM.

Algorithm 3: Offline Construction of FIRM Graph

1 input : Free space map, $\mathbb{X}_{free}$
2 output : FIRM graph $\mathcal{G}$
3 Sample PRM nodes $\mathcal{V} = \{\mathbf{n}_j\}_{j=1}^N$;
4 forall the $\mathbf{n}_i \in \mathcal{V}$ **do**
5   Design the stationary LQG μ^i about the node $\mathbf{n}_i$;
6   Compute associated b_s^i using Eq.(16);
7   Construct FIRM node B_i using Eq.(17);
8 forall the i **do**
9   **forall the** $j \in A(i)$ **do**
10   Set $b_0 = b_s^i$;
11   Generate sample belief path $b_{0:\mathcal{T}}$ (using Eq.(13)) and ground truth path $X_{0:\mathcal{T}}$ induced by controller μ^j invoked at B_i;
12   Compute the transition probabilities and costs associated with these sample paths using MC-based approaches.
13 Compute cost-to-go's $\{J(B_i)\}$ and feedback π over the FIRM by solving the DP in Eq.(8);
14 $\mathcal{G} = \big(\{B_i\}, \{J(B_i)\}, \{\mu^i\}, \pi\big)$;
15 return $\mathcal{G}$;

C. Planning with FIRM

Given that the FIRM graph is computed offline, the online phase of planning (and replanning) on the roadmap becomes very efficient and thus, feasible in real time. If the given initial belief b_0 does not belong to any B_i, we create a singleton set $B_0 = b_0$ and connect it to FIRM through its

k-nearest neighbors $A(0)$. Afterwards, due to the designed stopping condition, if no collision occurs, the belief is guaranteed to be in one of the nodes B_i at the decision stages. Thus, given the current node, we use policy π defined in Eq.(8b) over FIRM nodes to find j^*, and pick μ^{j^*} to move the robot into B_{j^*}. Algorithm 4 illustrates this procedure.

VI. EXPERIMENTAL RESULTS

In this section we construct FIRM on a sample environment. A 3-wheel omnidirectional mobile robot is used in experiments with the nonlinear kinematic model given in [23]. The state vector is composed of a 2D location and heading angle $X = [x, y, \theta]^T$. In experiments, the robot is equipped with exteroceptive sensors that provide range and bearing measurements from existing landmarks (radio beacons) in the environment. The 2D location of the j-th landmark is denoted by L_j. Measuring L_j can be modeled as follows:

$$^jZ = [\|^j\mathbf{d}\|, \text{atan2}(^jd_2, ^jd_1) - \theta]^T + {}^j\mathbf{v}, \quad {}^j\mathbf{v} \sim \mathcal{N}(\mathbf{0}, {}^j\mathbf{R}),$$

where, $^j\mathbf{d} = [^jd_1, {}^jd_2]^T := [x, y]^T - L_j$. $^j\mathbf{v}$ is a state-dependent observation noise, with covariance

$$^j\mathbf{R} = \text{diag}((\eta_r \|^j\mathbf{d}\| + \sigma_b^r)^2, (\eta_\theta \|^j\mathbf{d}\| + \sigma_b^\theta)^2). \tag{19}$$

In other words, the uncertainty (standard deviation) of sensor reading increases as the robot gets farther from the landmarks. $\eta_r = \eta_\theta = 0.3$ determines this dependence, and $\sigma_b^r = 0.01$ meter and $\sigma_b^\theta = 0.5$ degrees are the bias standard deviations. Similar model for range sensing is used in [12]. We assume the robot observes all N_L landmarks at all times and their observation noises are independent.

Figure 2(a) shows a sample environment, including obstacles, landmarks, and enumerated nodes in (x, y, θ) space.

Algorithm 4: Online Phase Algorithm

1 input : Initial belief b_0, FIRM graph $\mathcal{G}$
2 if $\exists B_m$ *such that* $b_0 \in B_m$ **then**
3   Set $i = m$ and compute $j^* = \pi(B_m)$;
4 else
5   Define the singleton set $B_0 = b_0$;
6   **forall the** $j \in A(0)$ **do**
7   Generate sample belief path $b_{0:\mathcal{T}}$ (using Eq.(13)) and ground truth path $X_{0:\mathcal{T}}$ induced by controller μ^j invoked at B_i;
8   Compute the transition probabilities and costs associated with these sample paths.
9   Set $i = 0$ and compute $j^* = \pi(B_0)$ using Eq.(8b);
10 while $B_i \neq B_{goal}$ **do**
11   **while** $b_k \notin B_j$ *and "no collision"* **do**
12   Apply the control $u_k = \mu^{j^*}(b_k)$ to the system and get the measurement Z_{k+1};
13   Update belief as $b_{k+1} = \mathbf{f}_b(b_k, \mu^{j^*}(b_k), Z_{k+1})$;
14   **if** *Collision happens* **then return** Collision;
15   Set $i = j$ and compute $j^* = \pi(B_i)$;

Nodes are shown by blue triangles, that encode the position (x, y) and heading angle θ of the robot. Landmarks are shown by black stars. The corresponding FIRM nodes are computed and shown in Fig. 2(b). All elements in Fig. 2(b) are defined in (x, y, θ) space but only the (x, y) portion of them is shown here. Each $b_s^j = (\mathbf{n}_j, P_\infty^j)$ is illustrated by a red dot representing $\mathbf{n}_j$ and a green ellipse, representing 3σ ellipse of covariance P_∞^j. Each FIRM node B_j is a neighborhood around b_s^j. In the experiments, we define the node region using the component-wise version of Eq.(17), to handle the error scale difference in position and orientation variables:

$$B_j = \{b = (X, P) | \ |X - \mathbf{n}_j| \dot{<} \epsilon, |P - P_\infty^j| \dot{<} \Delta\}, \ (20)$$

where, $|\cdot|$ and $\dot{<}$ stand for the absolute value and component-wise comparison operators, respectively. We set $\epsilon = [0.07(meter), 0.07(meter), 1(degree)]^T$ and $\Delta = \epsilon \epsilon^T$ to quantify B_j's. Part of this neighborhood that is defined for estimation mean $\widehat{X}^+$ is shown by a cyan rectangle centered at $\mathbf{n}_j$. The other part of this neighborhood is illustrated by two dashed green ellipses that represent 3σ covariances of $P_\infty^j - \Delta_d$ and $P_\infty^j + \Delta_d$, where Δ_d is the matrix Δ, whose off-diagonal elements are set to zero. For illustration purposes, both these neighborhoods are five times magnified in Fig. 2(b).

Figure 3(a) depicts the sample paths of ground truth state and estimation mean in green and dark red, respectively, for $M = 100$ particles. As seen in Fig. 3(a), the behavior of ground truth on the edges that have access to accurate observations is remarkably close to the planned behavior. In contrast, on the edges that get less informative observations, the controller cannot effectively compensate for the deviations of the ground truth from the nominal path, which can lead to collision with obstacles.

To avoid clutter, Fig. 3(b) depicts sample estimation covariance evolution only for a single particle. In this figure, we set the process and observation noises to zero, to keep the center of ellipses (i.e., estimation mean) on the planned points. However, note that in general estimation mean is affected by the noise (as it is seen in Fig. 3(a)). Indeed, Fig. 3(b) can be seen as the maximum-likelihood estimation uncertainty tube over the roadmap.

To complete the construction of FIRM, we compute the properties associated with invoking the j-th controller at the i-th node, such as collision probability, filtering performance, and stopping time. Table I shows these quantities for several (B_i, μ^j) pairs in FIRM. The pair (B_i, μ^j) represents the (i, j)-th FIRM edge. Finally, we perform planning on FIRM to find the optimal policy based on the defined costs in Eq.(18). We show the most likely path under the best policy of FIRM, i.e., Eq.(8b), in red in Fig. 3(b). The shortest path is also illustrated in Fig. 3(b) in yellow. It can be seen that the "most likely path under the best policy" detours from the shortest path to a path along which the filtering uncertainty is smaller and it is easier for the controller to avoid the collisions.
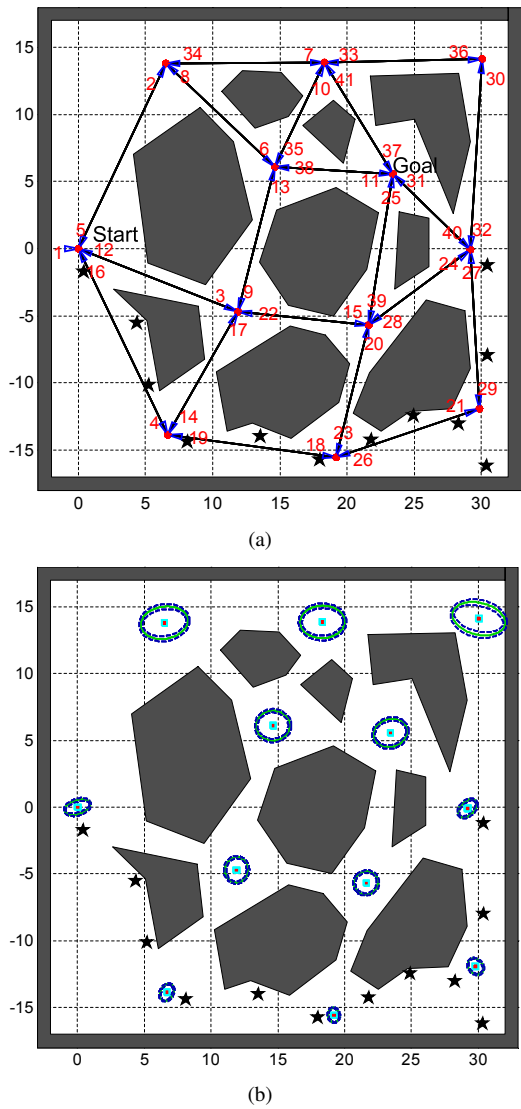

(a)

(b)

Fig. 2. (a) Figure depicts the underlying PRM graph. Gray polygons are the obstacles and black stars represent the landmarks' locations. (b) Selected FIRM nodes $\{B_1, B_2, B_3, B_4, B_6, B_7, B_{15}, B_{18}, B_{21}, B_{27}, B_{31}, B_{36}\}$.

VII. CONCLUSION

In this paper, we have proposed the Feedback controller-based Information-state road map (FIRM) for solving the motion planning problem under motion and sensing uncertainties. This problem originally is a POMDP, whose solution is intractable. Exploiting feedback controllers, we reduce it to a tractable FIRM MDP that can be solved by standard DP techniques. FIRM utilizes feedback controllers to create the reachable node regions in belief space, and construct a graph, on which a higher level policy is defined to provide the optimal plans. An important consequence is that FIRM overcomes the curse of history and curse of dimensionality in the original POMDP problem. Finally, by computing the collision probabilities, obstacles are also appropriately taken into account in planning on FIRM. We believe that FIRM provides an important step toward solving POMDPs and utilizing them as a practical tool for robot motion planning under uncertainty.

TABLE I
COMPUTED COSTS FOR SEVERAL NODE-CONTROLLER PAIRS IN FIRM USING 100 PARTICLES

B_i, μ^j pair	B_1,μ^4	B_4,μ^{18}	B_{18},μ^{21}	B_{21},μ^{27}	B_{27},μ^{31}	B_1,μ^3	B_3,μ^{13}	B_{13},μ^{11}
$1-P^{\mu^j}(F|B_i)$	%97	%95	%99	%77	%79	%87	%55	%79
Φ	18.5967	11.2393	6.8229	15.1148	26.2942	23.6183	48.8189	43.6207
$\mathbb{E}[\mathcal{T}], \sigma[\mathcal{T}]$	238.2, 21.8	193.0, 28.7	150.0, 15.1	209.6, 24.5	170.8, 22.6	200.3, 22.7	242.4, 30.1	219.2, 26.7

REFERENCES

[1] L. Kavraki, P. Svestka, J. Latombe, and M. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 4, pp. 566–580, 1996.

[2] S. Lavalle and J. Kuffner, "Randomized kinodynamic planning," *International Journal of Robotics Research*, vol. 20, no. 378-400, 2001.

[3] A. Nakhaei and F. Lamiraux, "A framework for planning motions in stochastic maps," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2008.

[4] L. Guibas, D. Hsu, H. Kurniawati, and E. Rehman, "Bounded uncertainty roadmaps for path planning," in *International Workshop on Algorithmic Foundations of Robotics*, 2008.

[5] P. Missiuro and N. Roy, "Adapting probabilistic roadmaps to handle uncertain maps," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2006.

[6] R. Alterovitz, T. Siméon, and K. Goldberg, "The stochastic motion roadmap: A sampling framework for planning with markov motion uncertainty," in *Proceedings of Robotics: Science and Systems (RSS)*, June 2007.

[7] S. Chakravorty and S. Kumar, "Generalized sampling based motion planners with application to nonholonomic systems," in *Proceedings of the IEEE Internation Conference on Systems, Man, and Cybernetics*, San Antonio, TX, October 2009.

[8] N. Melchior and R. Simmons, "Particle RRT for path planning with uncertainty," in *International Conference on Intelligent Robots and Systems (IROS)*, 2007.

[9] A. Censi, D. Calisi, A. D. Luca, and G. Oriolo, "A Bayesian framework for optimal motion planning with uncertainty," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Pasadena, CA, May 2008.

[10] J. van den Berg, P. Abbeel, and K. Goldberg, "LQG-MP: Optimized path planning for robots with motion uncertainty and imperfect state information," in *Proceedings of Robotics: Science and Systems (RSS)*, June 2010.

[11] R. Platt, R. Tedrake, L. Kaelbling, and T. Lozano-Perez, "Belief space planning assuming maximum likelihood observatoins," in *Proceedings of Robotics: Science and Systems (RSS)*, June 2010.

[12] S. Prentice and N. Roy, "The belief roadmap: Efficient planning in belief space by factoring the covariance," *International Journal of Robotics Research*, vol. 28, no. 11-12, October 2009.

[13] V. Huynh and N. Roy, "icLQG: combining local and global optimization for control in information space," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2009.

[14] N. D. Toit and J. W. Burdick, "Robotic motion planning in dynamic, cluttered, uncertain environments," in *ICRA*, May 2010.

[15] H. Kurniawati, Y. Du, D. Hsu, and W. S. Lee, "Motion planning under uncertainty for robotic tasks with long time horizons," *International Journal of Robotics Research*, vol. 30, pp. 308–323, 2010.

[16] D. Bertsekas, *Dynamic Programming and Optimal Control: 3rd Edition.* Athena Scientific, 2007.

[17] J. Pineau, G. Gordon, and S. Thrun, "Anytime point based approximations for large POMDPs," *Journal of Artificial Intelligence Research*, vol. 27, pp. 335–380, 2006.

[18] M. Spaan and N. Vlassis, "Perseus: Randomized point-based value iteration for POMDPs," *Journal of Artificial Intelligence Research*, vol. 24, pp. 195–220, 2005.

[19] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics.* MIT Press, 2005.

[20] R. Sutton, D. Precup, and S. Singh, "Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning," *Artificial Intelligence*, vol. 112, pp. 181–211, 1999.

[21] J. R. Norris, *Markov Chains.* Cambridge Univeristy Press, 1997.

[22] A. Agha-mohammadi, S. Chakravorty, and N. Amato, "FIRM: feedback controller-based information-state roadmap, a framework for motion planning under uncertainty," *Technical Report: TR11-001, Parasol Lab., CSE Dept., Texas A&M University*.

[23] T. Kalmár-Nagy, R. D'Andrea, and P. Ganguly, "Near-optimal dynamics trajectory generation and control of an omnidirectional vehicle," *Robotics and Autonomous Systems*, vol. 46, no. 1, pp. 47–64, January 2004.
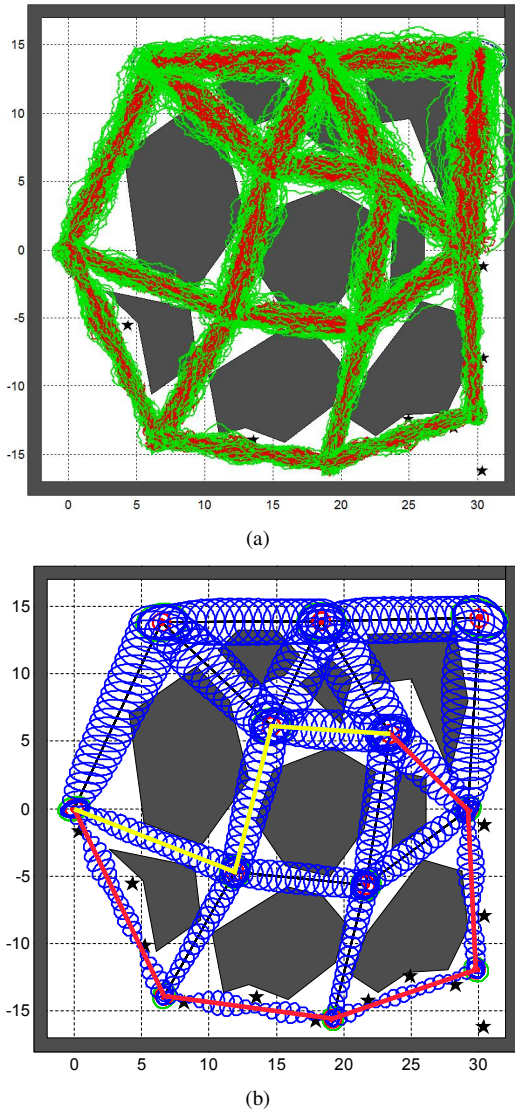
(a)

(b)

Fig. 3. Sample paths induced by the controllers invoked in different nodes. (a) For $M = 100$ particles, the sample ground truth paths and the sample estimation mean paths are shown in green and dark red, respectively. (b) The most likely path under the optimal policy and shortest path are shown in red and yellow respectively. The 3σ ML estimation uncertainty tube is drawn in blue.