

PLGRIM: Hierarchical Value Learning for Large-scale Autonomous Exploration in Unknown Environments

Sung-Kyun Kim,^{*1} Amanda Bouman,^{*2} Gautam Salhotra,³ David D. Fan,¹
Kyohei Otsu,¹ Joel Burdick,² Ali-akbar Agha-mohammadi¹

¹ NASA Jet Propulsion Laboratory California Institute of Technology,

² Department of Mechanical and Civil Engineering, California Institute of Technology,

³ Department of Computer Science, University of Southern California

{sung.kim, david.d.fan, kyohei.otsu, aliagha}@jpl.nasa.gov,

abouman@caltech.edu, jwb@robotics.caltech.edu, salhotra@usc.edu

Abstract

In order for an autonomous robot to efficiently explore an unknown environment, it must account for uncertainty in sensor measurements, hazard assessment, localization, and motion execution. Making decisions for maximal reward in a stochastic setting requires value learning and policy construction over a belief space, i.e., probability distribution over all possible robot-world states. However, belief space planning in a large spatial environment over long temporal horizons suffers from severe computational challenges. Moreover, constructed policies must safely adapt to unexpected changes in the belief at runtime. This work proposes a scalable value learning framework, PLGRIM (Probabilistic Local and Global Reasoning on Information roadMaps), that bridges the gap between (i) local, risk-aware resiliency and (ii) global, reward-seeking mission objectives. Leveraging hierarchical belief space planners with information-rich graph structures, PLGRIM addresses large-scale exploration problems while providing locally near-optimal coverage plans. We validate our proposed framework with high-fidelity dynamic simulations in diverse environments and on physical robots in Martian-analog lava tubes.

1 Introduction

Consider a large-scale coverage mission in an unknown environment, in which a robot is tasked with exploring and searching a GPS-denied unknown area, under given time constraints. This problem has a wide range of applications, such as inter-planetary exploration and search-and-rescue operations (Blank 2020; Nagatani et al. 2013). Essential elements of an autonomy architecture needed to realize such a mission include creating a map of the environment, accurately predicting risks, and planning motions that can meet the coverage and time requirements while minimizing risks. In such an architecture, quantifying and planning over uncertainty is essential for creating robust, intelligent, and optimal behaviors.

From a value learning perspective, a coverage planning problem in an unknown space can be considered an active

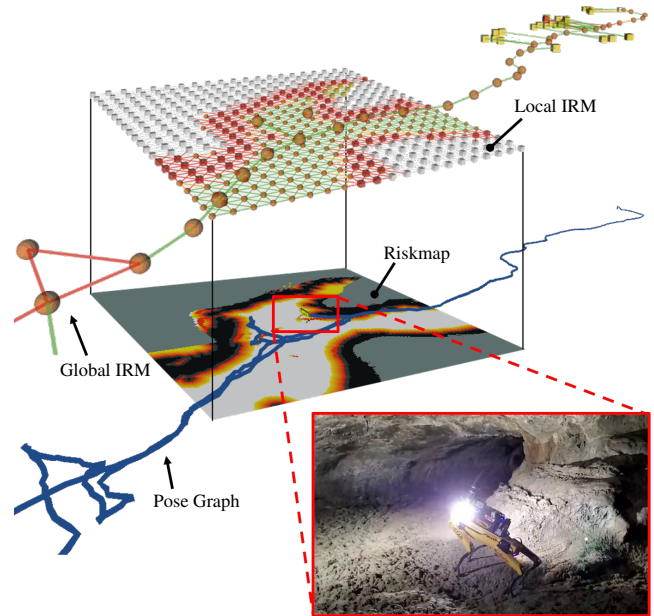


Figure 1: Hierarchical Information RoadMaps (IRMs) generated during Au-Spot's autonomous exploration of Martian-analog caves at Lava Beds National Monument, Tulelake, CA.

learning problem over the robot's belief, where belief is defined as the probability distributions over all possible joint robot-world states. The objective is to find the best action sequence that maximizes the accumulated reward over time. The agent must accumulate data to incrementally build a model of its environment, and need to understand the effects of its actions on the quality and quantity of data it collects.

Since the agent's future actions affect its belief of the world and robot state, this coverage problem is fundamentally a Partially Observable Markov Decision Process (POMDP) problem (Monahan 1982). The agent employs the underlying intrinsic model of the sequential action-observation process under uncertainty, so that it expands its search structure over the belief space and learns the value

^{*}These authors contributed equally to this work.

in a more sample-efficient manner than model-free learning approaches. In addition, its non-myopic reasoning can provide better performance than frontier-based exploration approaches with one-step look-ahead.

Belief value learning in the POMDP setting intrinsically suffers from the curse of dimensionality (Kaelbling, Littman, and Cassandra 1998) and curse of history (Pineau, Gordon, and Thrun 2003). Many powerful methods are proposed to extend the spatial and temporal horizons of POMDPs with varying degrees of efficiency and accuracy, such as (Silver and Veness 2010; Somani et al. 2013; Bonet and Geffner 1998; Kim, Salzman, and Likhachev 2019). In this paper, we focus on challenging exploration problems with very large spatial extents (>1 km), long temporal horizons (>1 hour), and high dimensional belief states (including beliefs on the state of the environment) that exacerbate the curses of dimensionality and history for POMDPs.

The main contribution of this work is in three-fold:

- 1) Scalable belief representation of local traversability and global coverage states of large environments.
- 2) Hierarchical value learning for efficient coverage policy search over a long horizon under uncertainty.
- 3) Policy reconciliation between planning episodes for adaptive and resilient execution in the real world.

More precisely, we introduce spatial and temporal approximations of the coverage policy space to enable computational tractability for real-time online solvers. Spatially, we decompose the belief space into task-relevant partitions of the space, enriched with environment map estimates. The partitioning structure is called an Information Roadmap (IRM) as shown in Fig. 1 (Agha-mohammadi, Chakravorty, and Amato 2014). Temporally, we decompose the problem into local and global hierarchical levels and solve for belief space policies that provide locally near-optimal coverage plans with global completeness. We then propose a Receding Horizon Planning (RHP)-based technique to address real-world stochasticity in state estimation and control at runtime.

The remainder of this paper is as follows: following the related work discussion, Section 3 formalizes the unknown environment coverage problem. In Section 4, we propose a hierarchical belief representation and value learning framework. Experimental results in simulation and on a physical robot are presented in Section 5, and Section 6 concludes this paper.

2 Related Work

Frontier-based exploration is a widely used approach for autonomous exploration (e.g., (Yamauchi 1997; Tao et al. 2007; Keidar and Kaminka 2012; Heng et al. 2015; González-Banos and Latombe 2002; Grabowski, Khosla, and Choset 2003)). By continuing exploration until exhausting all remaining frontiers, frontier-based approaches can guarantee completeness of the coverage of reachable spaces. These methods typically rely on myopic (e.g., one-step) look-ahead greedy policies, selecting the best frontier up-front. Hence, they can be subject to local minima and provide suboptimal solutions in time.

Model-free reinforcement learning (RL) has been applied to coverage and exploration problems (e.g., (Pathak et al. 2017; Burda et al. 2018a,b; Savinov et al. 2018)). In this setting, the typical approach is to find a policy which maps sensor data to actions, with the objective of maximizing the reward. When it comes to long-range, large-scale, and safety-critical missions on physical robots, collecting necessary data can be a significant challenge for this class of methods.

POMDP-based approaches generate a non-myopic policy by considering long-horizon action sequences (e.g., (Kurniawati et al. 2011), (Bai et al. 2015)), interactively learning the value function, and returning the best action sequence that maximizes the accumulated rewards. Different methods have reduced the complexity of the POMDP problem in coverage and exploration problems. Indelman, Carlone, and Dellaert (2015) and Martinez-Cantin et al. (2009) employed a direct policy search scheme with a Gaussian belief assumption. Lauri and Ritala (2016) extended this to non-Gaussian beliefs using the POMCP (Partially Observable Monte-Carlo Planning) solver. However, when it comes to the large-scale coverage missions, the current approaches do not scale well due to the curse of history and dimensionality (Pineau, Gordon, and Thrun 2003).

Hierarchical planning structures (Kaelbling and Lozano-Pérez 2011) aim to tackle larger problems by employing multiple solvers running at different resolutions, and are found to be effective. In the coverage and exploration context, Umari and Mukhopadhyay (2017) applied hierarchical planning to frontier-based exploration, while (Dang et al. 2019) extended the lower-level module to a more sophisticated frontier selection algorithm which considers the information gain along each path. Lauri and Ritala (2016) replaced the lower-level module with a POMDP-based planner to improve local coverage performance with non-myopic planning. Kim, Thakker, and Agha-Mohammadi (2019) proposed a hierarchical online-offline solver for risk-aware navigation. Vien and Toussaint (2015) suggested a hierarchical POMCP framework which outperformed Bayesian model-based hierarchical RL approaches in some benchmarks.

3 Problem Formulation

Autonomous exploration in unknown environments under motion and sensing uncertainty can be formulated as a Partially Observable Markov Decision Process (POMDP), which is one of the most general models for sequential decision making. In this section, we present a POMDP formulation for coverage problems and address its intrinsic challenges.

3.1 Preliminaries

A POMDP is described as a tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{Z}, T, O, R \rangle$, where \mathcal{S} is the set of joint robot-and-world states, \mathcal{A} and \mathcal{Z} are the set of robot actions and observations. At every time step, the agent performs an action $a \in \mathcal{A}$ and receives an observation $z \in \mathcal{Z}$ resulting from the robot’s perceptual interaction with the environment. The motion model $T(s, a, s') = p(s' | s, a)$ defines the probability of being at state s' after taking action

a at state s . The observation model $O(s, a, z) = p(z | s, a)$ is the probability of receiving observation z after taking action a at state s . The reward function $R(s, a)$ returns the expected utility for executing action a at state s . Belief state $b_t \in \mathbb{B}$ at time t denotes a posterior distribution over states conditioned on the initial belief b_0 and past action-observation sequence, i.e., $b_t = p(s | b_0, a_{0:t-1}, z_{1:t})$.

The optimal policy of a POMDP for all time $t \in [0, \infty)$, $\pi_{0:\infty}^* : \mathbb{B} \rightarrow \mathbb{A}$, is defined as:

$$\pi_{0:\infty}^*(b) = \operatorname{argmax}_{\pi \in \Pi_{0:\infty}} \mathbb{E} \sum_{t=0}^{\infty} \gamma^t r(b_t, \pi_t(b_t)), \quad (1)$$

where $\gamma \in (0, 1]$ is a discount factor for the future rewards, $\Pi_{0:\infty}$ is the space of possible policies, and $r(b, a) = \int_s R(s, a) b(s) ds$ denotes a belief reward which is the expected reward of taking action a at belief b .

3.2 Unknown Environment Coverage Problems

For our coverage planning problem, we define the state as $s = (q, W)$, where q is the robot state and W is the world state. We maintain two representations of the world, i.e., $W = (W_r, W_c)$, where W_r denotes the world traversal risk state and W_c is the world coverage state.

W_r encodes the traversability risk of the world with respect to a robot's dynamic constraints. This state is critical in capturing traversability-stressing elements of the environment (slopes, rough terrain, and narrow passages, etc.) and is typically constructed by aggregating long-range sensor measurements. The cost function $C(W_r, q, a)$ returns the actuation effort and risk associated with executing action a at robot state q on W_r .

W_c provides an estimation of what parts of the world have been observed, or *covered*, by a particular sensor. The coverage state is generated by specific sensor measurements, which may not necessarily be useful as navigational feedback, but instead are based on a task at hand. For instance, the coverage sensor may be a thermal camera for detecting thermal signatures, or a vision-based camera for identifying visual clues in the environment. As a robot moves, the sensor footprint sweeps the environment, expanding the covered area, or more generally, the task-relevant information about the world.

The coverage planning objective is to determine a trajectory through an environment that maximizes information gain I while simultaneously minimizing action cost C . As such, the traversal risk and coverage states form the basis of the coverage reward function:

$$R(s, a) = f(I(W_c, a), C(W_r, q, a)), \quad (2)$$

where $I(W_c, a) = H(W_c) - H(W_c | a)$ is quantified as reduction of the entropy H in W_c after taking action a . Note that in unknown space coverage domains, we do not have strong priors about the parts of the world that have not yet been observed. Hence, knowledge about W_c and W_r in Eq. (2) at runtime is incomplete and often inaccurate. Thus, in such domains, a Receding Horizon Planning (RHP) scheme has been widely adopted as the state-of-the-art (Bircher et al. 2016).

In POMDP formulation with RHP, the objective function in Eq. (1) is modified:

$$\pi_{t:t+T}^*(b) = \operatorname{argmax}_{\pi \in \Pi_{t:t+T}} \mathbb{E} \sum_{t'=t}^{t+T} \gamma^{t'-t} r(b_{t'}, \pi_{t'}(b_{t'})), \quad (3)$$

where T is a finite planning horizon for a planning episode at time t . Given the policy from the last planning episode, only a part of the optimal policy, $\pi_{t:t+\Delta t}^*$ for $\Delta t \in (0, T]$, will be executed at runtime. A new planning episode will start at time $t + \Delta t$ with updated belief about q , W_c , and W_r .

3.3 Challenges

We broadly identify the challenges associated with solving the unknown coverage planning problem, Eq. (3), as computational complexity—in both time and space—and conflicting policy objectives over consecutive planning episodes, arising from unexpected updates in the belief at runtime.

Time Complexity

POMDP planning suffers from *the curse of dimensionality* (Kaelbling, Littman, and Cassandra 1998) and *the curse of history* (Pineau, Gordon, and Thrun 2003). The former difficulty refers to fact that size of the belief grows exponentially with the size of the underlying state space. The latter difficulty refers to the fact that the number of action-observation sequences grows exponentially with the planning depth d , i.e., $\mathcal{O}(|\mathbb{A}|^d |\mathbb{Z}|^d)$. As an example, for large-scale exploration of a 1 km-long environment with an action resolution of 1 m, the planning depth d must be at least 10^3 in order to reason about the coverage plan across the environment.

Space Complexity

In addition to the classic time complexity of POMDPs, space complexity also poses a considerable challenge when handling the unknown environment coverage problem. Since $s = (q, W)$, the space complexity is dominated by the world state W . For example, in a grid world, the memory complexity is $\mathcal{O}(|n|^k)$, with n and k denoting the number of discretization levels and the grid dimension, respectively. For a 1 km² environment at a 0.1 m resolution, with floating-point risk and coverage values stored in every cell, required memory is 800 MB. This amount of memory should be allocated for every search node, and thus the full space complexity of planning is $\mathcal{O}(|\mathbb{A}|^d |\mathbb{Z}|^d |n|^k)$ during each planning episode.

Unexpected Belief Updates

As the robot explores its environment, it receives new sensory information, updates its belief, and constructs a new coverage policy in a receding horizon fashion. Policies generated during consecutive planning episodes must respect the kinodynamic constraints of the robot, while simultaneously adapting to unexpected hazards in the environment. We refer to these two distinct, and often opposed, objectives as *consistency* and *resiliency* of the receding-horizon policy, respectively. Path consistency ensures smooth trajectories and continuous velocities during transitions from one policy to the next, while path resiliency ensures the path adapts to unexpected changes in the world risk state. Thus, it is imperative to find a balance between policy consistency and resiliency, particularly for safety-critical systems.

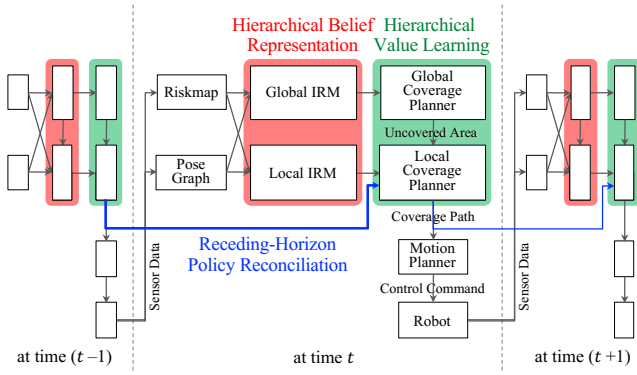


Figure 2: Illustration of PLGRIM framework for large-scale exploration in unknown environments. It *i*) maintains hierarchical beliefs about the traversal risks and coverage states, *ii*) performs hierarchical value learning to construct an exploration policy, and *iii*) reconciles policies over receding-horizon planning episodes.

4 PLGRIM: Hierarchical Coverage Planning on Information Roadmaps

In this section, we present a novel and field-hardened coverage planning autonomy framework, *PLGRIM* (*Probabilistic Local and Global Reasoning on Information roadMaps*), for exploration of large-scale unknown environments with complex terrain. Our proposed methods to tackle the challenges described in Section 3.3 are:

- 1) **Space Complexity:** We introduce a hierarchical belief space representation that is compact, versatile, and scalable. We refer to this representation as an Information RoadMap (IRM). Hierarchical IRMs can effectively encode a large-scale world state, while simultaneously capturing high-fidelity information locally.
- 2) **Time Complexity:** We propose hierarchical POMDP solvers that reason over long horizons within a suitable replanning time with locally near-optimal performance. Higher-level policies guide lower-level policies, resulting in a cascaded decision process.
- 3) **Unexpected Belief Updates:** We introduce a receding-horizon policy reconciliation method that respects the robot’s dynamic constraints while ensuring resiliency to unexpected observations.

In the following subsections, we provide the technical details about the proposed framework, illustrated in Fig. 2.

4.1 Overview

To enable efficient and reactive robot behaviors on very large scales, we decompose the problem into tractable subproblems by introducing spatial and temporal abstractions. Spatially, the belief space is approximated by a task-dependent structure, enriched with environment map estimates. Temporally, the belief space is approximated by the aggregation of multiple structures, each spanning a different spatial range. Finally, we introduce a cascaded optimization problem that returns a policy over the stratified belief space in real time.

Belief Decomposition

Let us denote the global world state as W^g and the local world state as W^ℓ , which is a subset of the global state, i.e., $W^\ell \subset W^g$, around the robot. We define local and global belief states as $b^\ell = p(q, W^\ell)$ and $b^g = p(q, W^g)$, respectively, where $p(W^\ell)$ is a local, robot-centric, rolling-window world belief representation with high-fidelity information, and $p(W^g)$ is a global, unbounded world belief representation with approximate information.

Policy Decomposition

We decompose the policy into local and global policies: π^ℓ and π^g , respectively. The overall policy $\pi \in \Pi$ is constructed by combining the local and global policies:

$$\pi(b) = \pi^\ell(b^\ell; \pi^g(b^g)). \quad (4)$$

We approximate the original RHP optimization problem in Eq. (3) as cascaded hierarchical optimization problems as follows:

$$\begin{aligned} \pi_{t:t+T}(b) &= \operatorname{argmax}_{\pi \in \Pi_{t:t+T}} \mathbb{E} \sum_{t'=t}^{t+T} \gamma^{t'-t} r(b_{t'}, \pi(b_{t'})) \\ &\approx \operatorname{argmax}_{\pi^\ell \in \Pi_{t:t+T}^\ell} \mathbb{E} \sum_{t'=t}^{t+T} \gamma^{t'-t} r^\ell(b_{t'}^\ell, \pi^\ell(b_{t'}^\ell; \pi_{t:t+T}^g(b_t^g))), \end{aligned} \quad (5)$$

$$\text{where } \pi_{t:t+T}^g(b^g) = \operatorname{argmax}_{\pi^g \in \Pi_{t:t+T}^g} \mathbb{E} \sum_{t'=t}^{t+T} \gamma^{t'-t} r^g(b_{t'}^g, \pi^g(b_{t'}^g)). \quad (6)$$

where $r^\ell(b^\ell, \pi^\ell(b^\ell))$ and $r^g(b^g, \pi^g(b^g))$ are approximate belief reward functions for the local and global belief spaces, respectively. Note that the codomain of the global policy $\pi^g(b^g)$ is a parameter space Θ^ℓ of the local policy $\pi^\ell(b^\ell; \theta^\ell)$, $\theta^\ell \in \Theta^\ell$.

According to this formulation, we maintain the hierarchical belief representations (Section 4.2) and solve for hierarchical POMDP policies (Section 4.3). For local planning consistency and resiliency, we extend Eq. (5) to a joint optimization problem given the previous planning episode policy (Section 4.4).

4.2 Hierarchical Belief Representation

We introduce a hierarchical approximation of the belief space by decomposing the environment representation into multiple information-rich structures, each referred to as an Information Roadmap (IRM). We construct and maintain IRMs at two hierarchical levels: the *Local IRM* and *Global IRM*, as illustrated in Fig. 1.

World Belief Information Sources

During its exploration of an unknown environment, at any given time, the robot’s understanding of the world is limited to noisy estimates of an observed subset of the world. IRMs are constructed from these estimates—namely, the *Riskmap* and *Pose Graph*. A Riskmap, constructed through the aggregation of point cloud sensor measurements, is a local rolling-window map that provides risk assessment, effectively encoding the risk belief over the local world state W^ℓ (Fan

Algorithm 1 Hierarchical IRM Construction

input: Riskmap, Pose Graph
Local IRM
 Local IRM $G^\ell = (N^\ell, E^\ell) \leftarrow (\emptyset, \emptyset)$
 Add uniformly sampled nodes $\{n_i^\ell\}_i$ around the robot to N^ℓ
for each $n_i^\ell \in N^\ell$ **do**
 Compute risk probability $p(n_{i,r}^\ell)$ and coverage probability $p(n_{i,c}^\ell)$ from Riskmap and Pose Graph for n_i^ℓ
 Add $p(n_{i,r}^\ell)$ and $p(n_{i,c}^\ell)$ to the properties of n_i^ℓ
end for
 Add edges for 8-connected neighbors, $\{e_{ij}\}_{i,j}$, to E^ℓ
for each $e_{ij}^\ell \in E^\ell$ **do**
 Compute traversal risk ρ_{ij} and distance d_{ij} for e_{ij}^ℓ
 Add ρ_{ij} and d_{ij} to the properties of e_{ij}^ℓ
end for
Global IRM
if not initialized **then**
 Global IRM $G^g = (N_b^g \cup N_f^g, E^g) \leftarrow (\emptyset, \emptyset)$
end if
 Get the current robot pose q from Pose Graph
if q is farther from any breadcrumb node $\forall n_i^g \in N_b^g$ than \bar{d}_b **then**
 Add a new breadcrumb node $n^g = q$ to N_b^g
end if
 Run FRONTIERMANAGER to add new frontiers $\{n_{f+}^g\}$ with coverage probabilities $\{p(n_{f+,c}^g)\}$, and prune invalidated frontiers, $\{n_{f-}^g\}$, based on the current Riskmap and Pose Graph
 \triangleright (Keidar and Kaminka 2012)
for each node $n_i^g \in \mathcal{N}_{G^g}(q)$ **do**
 for each nearby node $n_j^g \in \mathcal{N}_{G^g}(n_i^g)$ **do**
 Compute the traversal distance d_{ij} and risk ρ_{ij}
 if $d_{ij} < \bar{d}_e$ and $\rho_{ij} < \bar{\rho}_e$ **then**
 Add an edge e_{ij}^g to E^g with properties d_{ij} and ρ_{ij}
 else
 Remove the edge e_{ij}^g from E^g
 end if
 end for
end for
return G^ℓ and G^g

et al. 2021). A Pose Graph estimates the past trajectory of the robot from relative pose measurements and informs the coverage belief over the global world state W^g (Ebadi et al. 2020).

World Belief Construction

For compact and versatile representation of the world, we choose a generic graph structure, $G = (N, E)$ with nodes N and edges E , as the data structure to represent the belief about the world state. Using this framework, nodes represent discrete areas in space, and edges represent actions. More precisely, we define an action as a motion control from the current node $n_i \in N$ to a neighboring node $n_j \in N$, connected by an edge $e_{ij} \in E$.

For a detailed description of the Local and Global IRM construction processes, see Algorithm 1. We now describe the distinguishing features of each IRM:

- 1) *Local IRM*: As an instantiation of the local world belief $p(W^\ell)$, we employ a rolling, fixed-sized grid struc-

ture $G^\ell = (N^\ell, E^\ell)$, which is centered at the robot's current position. We uniformly sample nodes $n_i^\ell \in N^\ell$ from W^ℓ , and compute the risk and coverage probability distribution over a discrete patch centered at each node, i.e., $p(n_{i,r}^\ell)$ and $p(n_{i,c}^\ell)$, which are stored as node properties. For an edge e_{ij}^ℓ , we compute and store the traversal distance d_{ij} and risk ρ_{ij} , which effectively encodes $p(W_r^\ell)$ between two connected nodes. In summary, the Local IRM contains relatively high-fidelity information at a high resolution, but locally.

- 2) *Global IRM*: As an instantiation of the global world belief $p(W^g)$, we employ a sparse bidirectional graph structure $G^g = (N^g, E^g)$, which is fixed in the global reference frame. Due to the space complexity concerns detailed in Section 3.3, a densely-sampled grid structure, like G^ℓ , is not a viable option for G^g , as it should span up to several kilometers. Instead, we sparsely and non-uniformly sample nodes $n_i^g \in N^g$ from W^g based on certain node-classifying conditions. Specifically, N^g contains two mutually exclusive subsets of nodes: *breadcrumbs* and *frontiers*. Breadcrumb nodes are sampled directly from the Pose Graph, and thus capture the *covered traversable* space of W^g . Alternatively, frontier nodes are sampled from the border between covered and uncovered areas, and thus capture the *uncovered traversable* space of W^g . Finally, in order for such a candidate node n_i^g to be added to G^g , there must exist a traversable path to at least one nearby node $n_j^g \in N^g$. If such a path exists, an edge e_{ij}^g , storing traversal distance d_{ij} and risk ρ_{ij} , is added to G^g . See Fig. 3 for identification of breadcrumb and frontier nodes in G^g . In summary, the Global IRM captures the free-space connectivity of W^g with a notion of coverage, and does not explicitly encode highly-likely untraversable or uncertain areas in W^g in order to achieve compact representation of the large-scale environment.

4.3 Hierarchical Value Learning

Given Local and Global IRMs as the hierarchical belief representation, we solve the cascaded hierarchical POMDP problems, Eq. (5) and Eq. (6), for coverage in an unknown environment.

Solver Formulation

We start by introducing some notations. We define *value function* $V(b; \pi)$ as the expected reward of following policy π , starting from belief b :

$$V(b; \pi) = \mathbb{E} \left[\sum_t \gamma^t r(b_t, \pi(b_t)) \right]. \quad (7)$$

From a recursive form of the value function, we can define the value of taking action a in belief b under a policy π by the *action-value function*:

$$Q(b, a; \pi) = r(b, a) + \sum_{b' \in \mathbb{B}} \gamma \mathcal{T}(b, a, b') V(b'; \pi), \quad (8)$$

where $\mathcal{T}(b, a, b')$ is the transition probability from b to b' by

action a , as follows:

$$\mathcal{T}(b, a, b') = \sum_{z \in \mathbb{Z}} p(b'|b, a, z) p(z|b, a). \quad (9)$$

A POMDP solver tries to learn $Q(b, a)$ and $V(b) = \max_{a \in \mathbb{A}} Q(b, a)$, and returns the policy π that specifies the best action for a given belief b , i.e., $\pi(b) = \arg\max_{a \in \mathbb{A}} Q(b, a)$.

Generalized Coverage Reward

Entropy provides a measure of uncertainty of a random variable's belief. Given an IRM $G = (N, E)$ containing a $p(n_{i,c})$ value for each node $n_i \in N$, the entropy of the world coverage state is:

$$H(p(W_c)) = - \sum_i^{|N|} \left[p(n_{i,c}) \log p(n_{i,c}) + p(n_{i,-c}) \log p(n_{i,-c}) \right]. \quad (10)$$

If $a \in \mathbb{A}$ is a motion from node $n_i \in N$ to node $n_j \in N$ along edge $e_{ij} \in E$, then the coverage information gain (i.e., coverage uncertainty reduction) in *coverage belief* $p(W_c)$ induced by a is defined as:

$$I(W_c | a) = \underbrace{H(p(W_c))}_{\text{current entropy}} - \underbrace{H(p(W_c | a))}_{\text{future entropy}}, \quad (11)$$

where the second term represents the expected future entropy of the world coverage state after execution of action a .

Although the action cost function at each hierarchical level is dependent upon the IRM's particular action set E (i.e., Local and Global IRMs have different action sets), it can be generically formulated as:

$$C(W_r, q, a) = k_d d_{ij} + k_\rho \rho_{ij} + k_\mu \mu_{ij}(q), \quad (12)$$

where d_{ij} and ρ_{ij} are the traversal distance and risk along edge e_{ij} , respectively. The cost $\mu_{ij}(q)$ is associated with the current motion primitive, and is a consequence of the robot's non-holonomic constraints, such as the heading direction. Constants k_d , k_ρ , and k_μ weigh the importance of traversal distance, risk, and motion primitive history on the total action cost.

Then, finally the coverage reward function is defined as a weighted sum of the information gain and action cost:

$$R(s, a) = k_I I(W_c, z) - k_C C(W_r, q, a), \quad (13)$$

where k_I and k_C are constant weights.

Local-Global Coverage Planner Coordination

In our cascaded hierarchical optimization framework, we first solve for the global policy in Eq. (6). The global policy solution then serves as an input parameter to the local policy in Eq. (5). This means that Global Coverage Planner (GCP) provides *global guidance* to the Local Coverage Planner (LCP).

The role of GCP is to construct a low-fidelity policy that provides global guidance to uncovered areas, at which point,

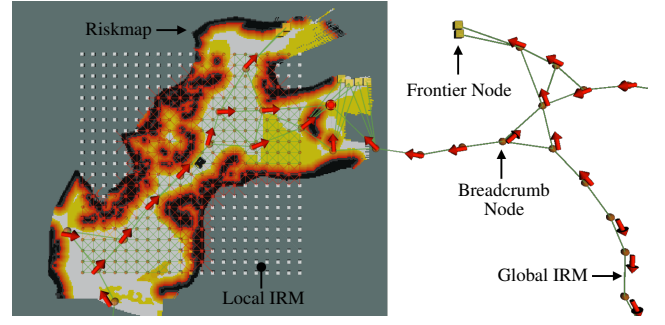


Figure 3: QMDP policy (red arrows displayed above *breadcrumb* nodes) for Global Coverage Planning (GCP). A red sphere indicates the QMDP frontier goal.

LCP instructs a local coverage behavior. More concretely, a target frontier node in the Global IRM, $n_f^g \in N_f^g$, can be extracted from the global-level control $a^g \in \mathbb{A}^g$ provided by GCP. Since the environment can be very large (>1 km), GCP must be capable of reasoning over hundreds of nodes on the Global IRM. To alleviate this scalability challenge, we assume that GCP's policy terminates at frontier nodes. By classifying frontier nodes as terminal in the belief space, we can assume no changes occur to the world coverage state before termination. Therefore, we omit W from the state space for GCP.

The role of LCP is to construct a high-fidelity policy that provides local guidance based on information gathering, traversal risk (e.g., proximity to obstacles, terrain roughness, and slopes), and the robot's mobility constraints (e.g., acceleration limits and non-holonomic constraints of wheeled robots). LCP has two phases: i) reach the target area based on GCP's guidance, and ii) construct a local coverage path after reaching the target area. If the target frontier is outside the Local IRM range, i.e., $n_f^g \notin W^\ell$, LCP simply instantiates high-fidelity control based on the global-level control a^g in order to reach the target frontier. If the target frontier n_f^g is within the Local IRM range, i.e., $n_f^g \in W^\ell$, then LCP performs the nominal information-gathering coverage optimization, as described in Eq. (5).

Global Coverage Planner (GCP) Algorithm

In this work, we adopt the QMDP approach for the global coverage planning problem (Littman, Cassandra, and Kaelbling 1995). The key idea of QMDP is to assume the state becomes fully observable after one action, so that the *value function* for further actions can be evaluated efficiently in an MDP (Markov Decision Process) setting. In our global coverage planning domain, we define the first action to be the robot's relocation to a nearby node on the Global IRM. At this point, the robot pose is assumed to be fully observable, while the world risk and coverage states remain unchanged.

More formally, we solve for $Q_{\text{MDP}}^g(q^g, a^g)$ by ignoring uncertainty in the robot pose q^g and changes in the world coverage state W_c^g . In this MDP setting, $Q_{\text{MDP}}^g(q^g, a^g)$ can be learned by Value Iteration very efficiently, even for long discount horizons. Then, we evaluate the *action-value function* in Eq. (8) in a POMDP setting for the current belief and

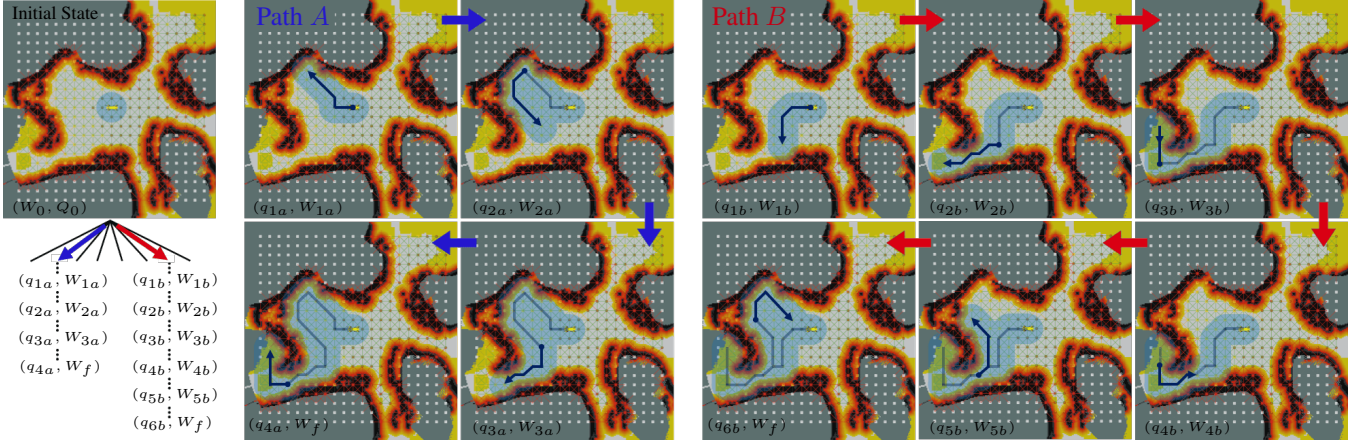


Figure 4: Illustrative example of coverage path planning on the Local IRM with Monte-Carlo Tree Search. The field-of-view of the robot’s coverage sensor is represented by a blue circle. Macro actions (6 steps on Local IRM in this example) associated with the two tree branches, paths A and B, are shown. Note that the final world coverage states in both branches are identical. Path A is evaluated to be more rewarding than B since fewer actions were required to cover the same area.

the feasible one-step actions:

$$Q(b^g, a^g) = \int_{q^g} b(q^g) Q_{\text{MDP}}^g(q^g, a^g) dq^g. \quad (14)$$

Finally, a POMDP policy can be obtained as follows:

$$\pi^g(b^g) = \operatorname{argmax}_{a^g \in \mathbb{A}^g} Q(b^g, a^g). \quad (15)$$

An example of the GCP policy is depicted in Fig. 3.

Local Coverage Planner (LCP) Algorithm

In order to solve Eq. (5), we employ POMCP (Partially Observable Monte Carlo Planning) algorithm (Silver and Veness 2010). POMCP is a widely-adopted POMDP solver that leverages the Monte Carlo sampling technique to alleviate both of the *curse of dimensionality* and *history*. Given a generative model (or a black box simulator) for discrete action and observation spaces, POMCP can learn the *value function* of the reachable belief subspace with an adequate exploration-exploitation trade-off.

More concretely, POMCP evaluates $Q^\ell(b^\ell, a^\ell)$ in Eq. (8) by unrolling recursive value backpropagation through sampled action-observation sequences. UCT algorithm for action selection helps to balance between exploration and exploitation in order to learn the *action-value function*. Initially, it explores the search space (possible action-observation sequences) with a random or a heuristically guided rollout policy. While incrementally building the belief tree, it gradually exploits the learned values for more focused exploration. See illustration of local coverage planning in Fig. 4.

4.4 Receding-Horizon Policy Reconciliation

We extend the receding-horizon *local* coverage planning problem to address the trade-off between policy consistency and resiliency, as described in Section 3.3.

We define a policy reconciliation optimization problem by introducing the previous planning episode policy into Eq. (3)

for the current planning episode. For notational brevity, let us denote the time when the previous policy was generated as t_0 and the current time as $t_1 = t_0 + \Delta t$. In order to reconcile consecutive policies over receding horizons, we extend Eq. (3) as follows, given the previous policy $\pi_{t_0:t_0+T}^-$ constructed at time t_0 for a finite horizon of T :

$$\begin{aligned} & \pi_{t_1:t_1+T}^*(b; \pi_{t_0:t_0+T}^-) \\ &= \operatorname{argmax}_{\pi \in \Pi_{t_1:t_1+T}} \left[\mathbb{E} \sum_{t'=t_1}^{t_1+T} \gamma^{t'-t_1} r(b_{t'}, \pi(b_{t'})) \right. \\ & \quad \left. - \lambda \mathcal{R}(\pi_{t_0:t_0+T}^-, \pi_{t_1:t_1+T}) \right], \quad (16) \end{aligned}$$

where $\mathcal{R}(\pi_{t_0:t_0+T}^-, \pi_{t_1:t_1+T})$ is a regularizing cost function that penalizes inconsistency between the previous and current policies in terms of kinodynamic constraints, and λ is a regularization weight parameter. The first term in Eq. (16) pursues policy resiliency based on the up-to-date world belief, which may encode unexpected hazards, while the second term promotes policy consistency.

Since the conflict between policy consistency and resiliency is most severe at the junction between two consecutive policies, we decompose Eq. (16) into two time frames, $(t_1 : t_1 + \tau)$ and $(t_1 + \tau : t_1 + T)$ for $\tau \in [0, T - \Delta t]$, and formulate it as a simplified joint optimization problem for τ^* and $\pi_{t_1+\tau^*:t_1+T}^*$:

$$\tau^* = \operatorname{argmax}_{\tau \in [0, T - \Delta t]} \mathbb{E} \sum_{t'=t_1}^{t_1+\tau} \gamma^{t'-t_1} r(b_{t'}, \pi_{t_0:t_0+T}^-(b_{t'})), \quad (17)$$

$$\begin{aligned} & \pi_{t_1+\tau^*:t_1+T}^* \\ &= \operatorname{argmax}_{\pi \in \Pi_{t_1+\tau^*:t_1+T}} \mathbb{E} \sum_{t'=t_1+\tau^*}^{t_1+T} \gamma^{t'-t_1} r(b_{t'}, \pi(b_{t'})), \quad (18) \end{aligned}$$

$$\pi_{t_1:t_1+T}^* = [\pi_{t_1:t_1+\tau^*}^-; \pi_{t_1+\tau^*:t_1+T}^*]. \quad (19)$$

Policy reconciliation is performed in Eq. (17) over a single optimization variable τ . By re-evaluating the previous pol-

icy $\pi_{t_0:t_0+T}^-$ with updated robot-world belief $b_{t'}$, τ dictates how much of the new $\pi_{t_1:t_1+T}^*$ should be in agreement with the previous policy. Effectively, a larger τ promotes policy consistency, while a smaller τ promotes policy resiliency.

Given τ^* from Eq. (17), the optimization problem in Eq. (18) becomes identical to Eq. (3), except the change of start time, and can be solved by LCP, as described in Section 4.3. The final receding-horizon policy $\pi_{t_1:t_1+T}^*$ is then constructed by concatenating the previous policy and a new partial policy, as in Eq. (18).

5 Experimental Results

In order to evaluate our proposed framework, we perform high-fidelity simulation studies with a four-wheeled vehicle (Husky robot) and real-world experiments with a quadruped (Boston Dynamics Spot robot). Both robots are equipped with custom sensing and computing systems, enabling high levels of autonomy and communication capabilities (Otsu et al. 2020; Agha-mohammadi and et al. 2021). The entire autonomy stack runs in real-time on an Intel Core i7 processor with 32 GB of RAM. The stack relies on a multi-sensor fusion framework. The core of this framework is 3D point cloud data provided by LiDAR range sensors mounted on the robots (Ebadi et al. 2020). We refer to our autonomy stack-integrated Spot as Au-Spot (Bouman* et al. 2020).

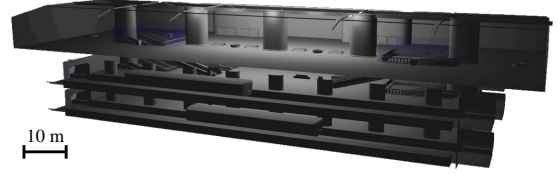
5.1 Baseline Algorithms

We compare our PLGRIM framework against a local coverage planner baseline (next-best-view method) and a global coverage planner baseline (frontier-based method).

- 1) *Next-Best-View (NBV)*: NBV first samples viewpoints in a neighborhood of the robot, and then plans a deterministic path over a high-fidelity local world representation to each viewpoint (Bircher et al. 2016). The set of viewpoint paths serves as the policy search space. Each policy in the space is evaluated, and NBV selects the policy with the maximum reward, computed using action cost and information gain from the world representation. While NBV is able to leverage local high-fidelity information, it suffers due to its spatially limited world belief and sparse policy space.
- 2) *Hierarchical Frontier-based Exploration (HFE)*: Frontier based exploration methods construct a global, but low-fidelity, representation of the world, where frontiers encode approximate local information gain. The set of frontiers serves as the policy search space. Exploration interleaves a one-step look-ahead frontier selection and the creation of new frontiers, until all frontiers have been explored. Hierarchical approaches can enhance the performance of frontier-based methods by modulating the spatial scope of frontier selection (Umari and Mukhopadhyay 2017). However, while HFE is able to reason across the global world belief, it suffers from downsampling artifacts and a sparse policy space composed of large action steps.

Note that in order to achieve reasonable performance in the complex simulated environments, we allow each baseline to leverage our Local and Global IRM structures as the underlying search space.

(a) Simulated Subway 1x



(b) Simulated Maze



(c) Simulated Cave

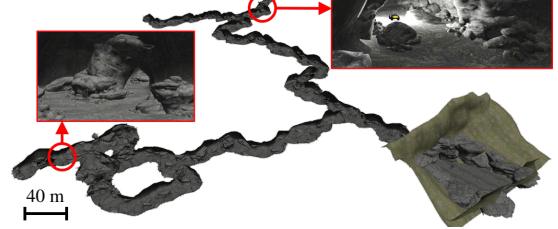


Figure 5: PLGRIM’s performance was evaluated in various simulated environments: (a) subway station, (b) maze (top-down view), and (c) cave.

5.2 Simulation Evaluation

We demonstrate PLGRIM’s performance, as well as that of the baseline algorithms, in a simulated subway, maze, and cave environment. Fig. 5 visualizes these environments.

Simulated Subway Station

The subway station consists of large interconnected, polygonal rooms with smooth floors, devoid of obstacles. There are three varying sized subway environments, whose scales are denoted by 1x, 2x, and 3x. Fig. 9(a)-(c) shows the scalable performance of PLGRIM against the baselines. In a relatively small environment without complex features (Subway 1x), NBV performance is competitive as it evaluates high-resolution paths based on information gain. However, as the environment scale grows, its myopic planning easily gets *stuck* and the robot’s coverage rate drops significantly. HFE shows inconsistent performance in the subway environments. The accumulation of locally suboptimal decisions, due to its sparse environment representation, leads to the construction of a globally inefficient IRM structure. As a result, the robot must perform time-consuming detours in order to *pick up* leftover frontiers.

Simulated Maze and Cave

The maze and cave are both unstructured environments with complex terrain (rocks, steep slopes, etc.) and topology (narrow passages, sharp bends, dead-ends, open-spaces, etc.). The coverage rates for each algorithm are displayed in Fig. 9(d)-(e). PLGRIM outperforms the baseline methods in these environments. By constructing long-horizon cover-

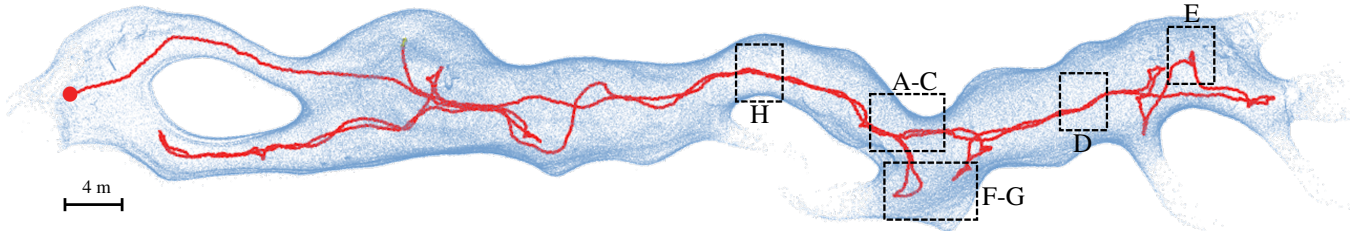


Figure 6: PLGRIM's exploration trajectory in Valentine Cave, Lava Beds National Monument, Tulelake, CA. Exploration started at the mouth of the cave (red circle), reached the end of the cave on the right, and returned back to visit uncovered areas. Boxes indicate the portions of the trajectory associated with the alphabetized snapshots in Fig. 7 and Fig. 8.

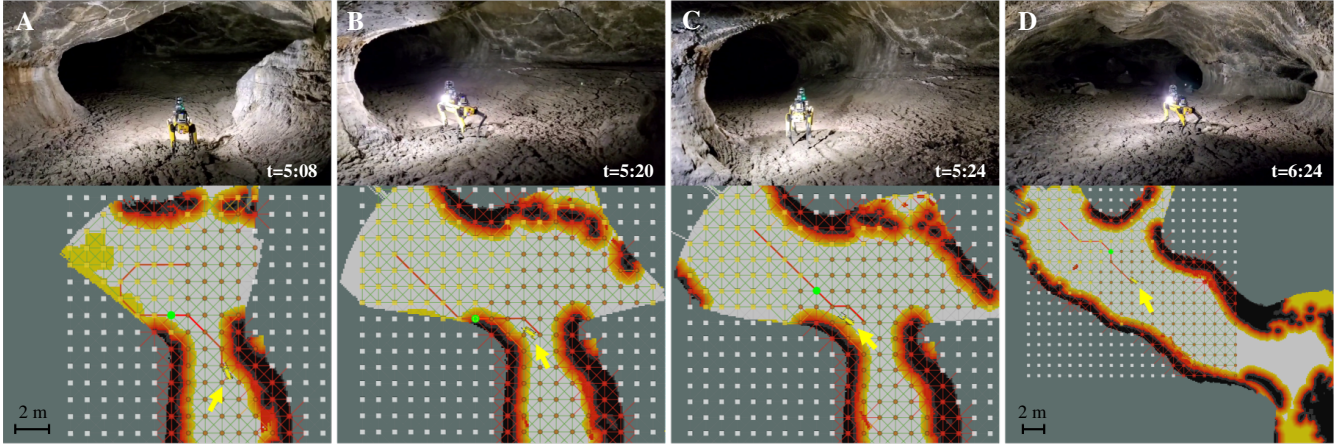


Figure 7: The Local IRM (yellow, brown, and white nodes represent uncovered, covered and unknown areas, respectively) is shown overlaid on the Riskmap. A yellow arrow indicates the robot's location. LCP plans a path (red) that fully covers the local area (snapshot A). When $p(W^\ell)$ updates, the path is adjusted to extend towards the large uncovered swath while maintaining smoothness with the previous path. Another $p(W^\ell)$ update reveals that the path has entered a hazardous area—wall of lava tube (snapshot B). As a demonstration of LCP's resiliency, the path shifts away from the hazardous area, and the robot is re-directed towards the center of the tube (snapshot C). One minute later, the robot encounters a fork in the cave. The LCP path curves slightly toward fork apex (for maximal information gain) before entering the wider, less-risky channel (snapshot D).

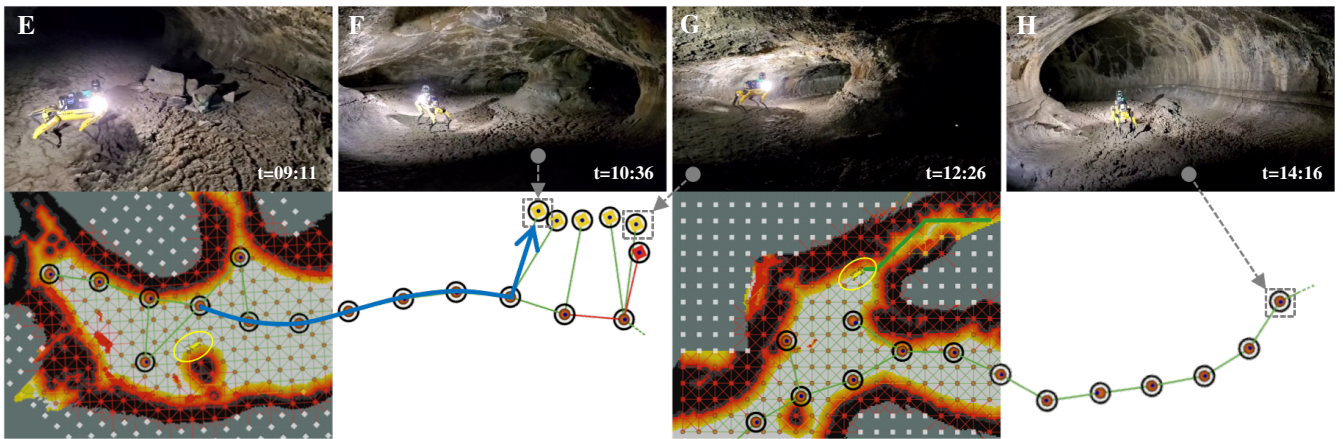


Figure 8: Portions of the Global IRM constructed in the lava tube are visualized—yellow nodes represent frontiers, brown nodes represent breadcrumbs. Gray arrows associate a frontier with a snapshot of the robot exploring that frontier. GCP plans a path (blue) along the Global IRM to a target frontier after the local area is fully covered (snapshot E). The robot explores the area around the frontier (snapshot F), and then explores a neighboring frontier at the opening of a narrow channel to its right. LCP plans a path (green) into the channel (snapshot G). Later, after all local areas have been explored, the robot is guided back towards the mouth of cave along the breadcrumb nodes (snapshot H).

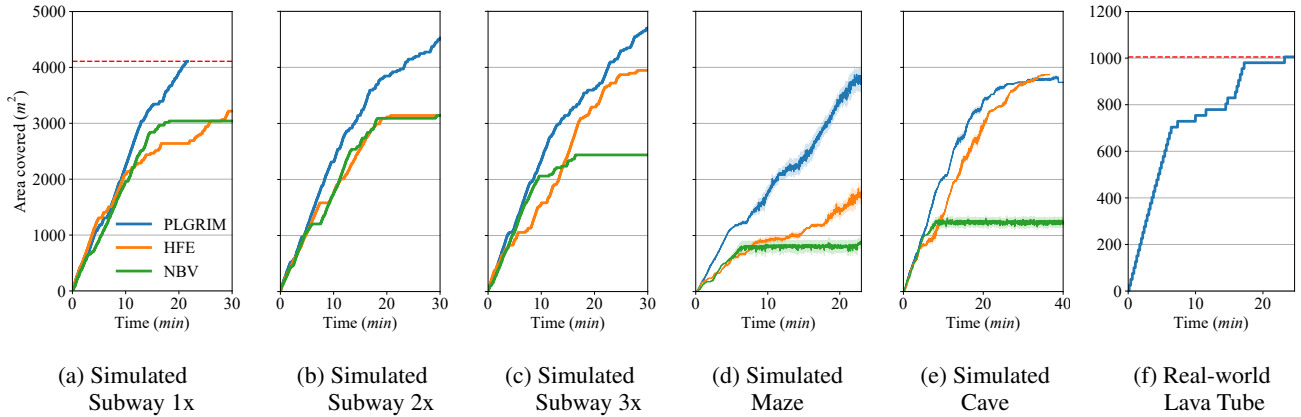


Figure 9: Exploration by PLGRIM and baseline methods in simulated subway environments of increasing size (a)-(c), and in simulated and real-world cave environments (d)-(f). For (d) and (e), the covered area is the average of two runs. Red dashed lines indicate 100% coverage of the environments, where applicable.

age paths over a high-resolution world belief representation, PLGRIM enables the robot to safely explore through hazardous terrain. Simultaneously, it maintains an understanding of the global world, which is leveraged when deciding where to explore next after exhausting all local information. In the cave, NBV’s reliance on a deterministic path, without consideration of probabilistic risk, causes the robot to drive into a pile of rocks and become inoperable. NBV exhibits similarly poor performance in the maze. However, in this case, NBV’s myopic planning is particularly ineffectual when faced with navigating a topologically-complex space, and the robot ultimately gets *stuck*. As was the case in the subway, HFE suffers in the topologically-complex maze due to an accumulation of suboptimal local decisions. In particular, frontiers are sometimes not detected in the sharp bends of the maze, leaving the robot with an empty local policy space. As a result, the robot cannot progress and spends considerable time backtracking along the IRM to distant frontiers.

5.3 Real-World Evaluation

We extensively validated PLGRIM on physical robots in real-world environments. In particular, PLGRIM was run on Au-Spot in a lava tube, located in Lava Beds National Monument, Tulelake, CA. The cave consists of a main tube, which branches into smaller, auxiliary tubes. The floor is characterized by ropy masses of cooled lava. Large boulders, from ceiling breakdown, are scattered throughout the tube. Fig. 6 shows the robot’s trajectory overlaid on the aggregated LiDAR point cloud. Fig. 7 and 8 discuss how PLGRIM is able to overcome the challenges posed by large-scale environments with complex terrain and efficiently guide the robot’s exploration. Fig. 9(f) shows the area covered over time.

6 Conclusion

In this work, we develop a hierarchical framework for exploring large-scale, unknown environments with complex terrain in a POMDP setting. To obtain a tractable solution,

we introduce a hierarchical belief space representation that effectively encodes a large-scale world state, while simultaneously capturing high-fidelity information local to the robot. Then we propose cascaded POMDP solvers that reason over long horizons within a suitable replanning time. We demonstrate our framework in high-fidelity dynamic simulation environments and in real-world environments, namely a natural cave. Future work includes incorporating semantic information gain, such as a science target signature, into the IRMs, as well as extending the PLGRIM framework to multi-robot coverage problems.

Acknowledgments

The work is partially supported by the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration (80NM0018D0004), Defense Advanced Research Projects Agency (DARPA), and the BRAILLE Mars Analog project funded by the NASA PSTAR program (NNH16ZDA001N). We acknowledge our team members in Team CoSTAR for the DARPA Subterranean Challenge, Dr. Jennifer Blank at NASA Ames, Shushman Choudhury, and the resource staff at Lava Beds National Monument for their support.

References

- Agha-mohammadi, A.; Chakravorty, S.; and Amato, N. 2014. FIRM: Sampling-based Feedback Motion Planning Under Motion Uncertainty and Imperfect Measurements. *IJRR* 33(2): 268–304.
- Agha-mohammadi, A.; and et al. 2021. NeBula: Quest for Robotic Autonomy in Challenging Environments; TEAM CoSTAR at the DARPA Subterranean Challenge. In *arXiv preprint arXiv:2103.11470*.
- Bai, H.; Cai, S.; Ye, N.; Hsu, D.; and Lee, W. S. 2015. Intention-aware online POMDP planning for autonomous driving in a crowd. In *IEEE ICRA*, 454–460.
- Bircher, A.; Kamel, M.; Alexis, K.; Oleynikova, H.; and Siegwart, R. 2016. Receding horizon “next-best-view” planner for 3D exploration. In *IEEE ICRA*, 1462–1468.

- Blank, J. G. 2020. Robotic Mapping and Exploration of a Terrestrial Lava Tube: A Structured Planetary Cave Mission Simulation with a Remote Astrobiology Science Team. In *3rd International Planetary Caves Conference*.
- Bonet, B.; and Geffner, H. 1998. Learning Sorting and Decision Trees with POMDPs. In *ICML*, 73–81.
- Bouman*, A.; Ginting*, M.; Alatur*, N.; Palieri, M.; Fan, D.; Touma, T.; Pailevanian, T.; Kim, S.; Otsu, K.; Burdick, J.; and Agha-Mohammadi, A. 2020. Autonomous Spot: Long-Range Autonomous Exploration of Extreme Environments with Legged Locomotion. In *IEEE/RSJ IROS*.
- Burda, Y.; Edwards, H.; Pathak, D.; Storkey, A.; Darrell, T.; and Efros, A. 2018a. Large-Scale Study of Curiosity-Driven Learning. In *arXiv preprint arXiv:1808.04355*.
- Burda, Y.; Edwards, H. A.; Storkey, A. J.; and Klimov, O. 2018b. Exploration by Random Network Distillation. In *arXiv preprint arXiv:1810.12894*.
- Dang, T.; Khattak, S.; Mascari, F.; and Alexis, K. 2019. Explore Locally, Plan Globally: A Path Planning Framework for Autonomous Robotic Exploration in Subterranean Environments. In *ICAR*, 9–16.
- Ebadi, K.; Chang, Y.; Palieri, M.; Stephens, A.; Hatteland, A.; Heiden, E.; Thakur, A.; Morrell, B.; Carlone, L.; and Agha, A. 2020. LAMP: Large-scale Autonomous Mapping and Positioning for Exploration of Perceptually-degraded Subterranean Environments. In *ICRA*.
- Fan, D. D.; Otsu, K.; Kubo, Y.; Dixit, A.; Burdick, J.; and Agha-mohammadi, A.-a. 2021. STEP: Stochastic traversability evaluation and planning for safe off-road navigation. In *arXiv preprint arXiv:2103.02828*.
- González-Banos, H. H.; and Latombe, J.-C. 2002. Navigation strategies for exploring indoor environments. *IJRR* 21(10-11): 829–848.
- Grabowski, R.; Khosla, P.; and Choset, H. 2003. Autonomous exploration via regions of interest. In *IEEE/RSJ IROS*, volume 2, 1691–1696.
- Heng, L.; Gotovos, A.; Krause, A.; and Pollefeys, M. 2015. Efficient visual exploration and coverage with a micro aerial vehicle in unknown environments. In *ICRA*, 1071–1078.
- Indelman, V.; Carlone, L.; and Dellaert, F. 2015. Planning in the continuous domain: A generalized belief space approach for autonomous navigation in unknown environments. *IJRR* 34(7): 849–882.
- Kaelbling, L.; Littman, M.; and Cassandra, A. 1998. Planning and acting in Partially Observable stochastic domains. *Artificial Intelligence* 101: 99–134.
- Kaelbling, L. P.; and Lozano-Pérez, T. 2011. Planning in the know: Hierarchical belief-space task and motion planning. In *Workshop on Mobile Manipulation, IEEE ICRA*.
- Keidar, M.; and Kaminka, G. A. 2012. Robot exploration with fast frontier detection: theory and experiments. In *AAMAS*, 113–120.
- Kim, S.-K.; Salzman, O.; and Likhachev, M. 2019. POMHDP: Search-based belief space planning using multiple heuristics. In *ICAPS*, volume 29, 734–744.
- Kim, S.-K.; Thakker, R.; and Agha-Mohammadi, A.-A. 2019. Bi-directional value learning for risk-aware planning under uncertainty. *IEEE RA-L* 4(3): 2493–2500.
- Kurniawati, H.; Du, Y.; Hsu, D.; and Lee, W. S. 2011. Motion planning under uncertainty for robotic tasks with long time horizons. *IJRR* 30(3): 308–323.
- Lauri, M.; and Ritala, R. 2016. Planning for robotic exploration based on forward simulation. *Robotics and Autonomous Systems* 83: 15–31.
- Littman, M.; Cassandra, A.; and Kaelbling, L. 1995. Learning policies for partially observable environments: Scaling up. In *Machine Learning Proceedings*, 362–370.
- Martinez-Cantin, R.; De Freitas, N.; Brochu, E.; Castellanos, J.; and Doucet, A. 2009. A Bayesian exploration-exploitation approach for optimal online sensing and planning with a visually guided mobile robot. *Autonomous Robots* 27(2): 93–103.
- Monahan, G. E. 1982. State of the art—a survey of partially observable Markov decision processes: theory, models, and algorithms. *Management science* 28(1): 1–16.
- Nagatani, K.; Kiribayashi, S.; Okada, Y.; Otake, K.; Yoshida, K.; Tadokoro, S.; Nishimura, T.; Yoshida, T.; Koyanagi, E.; Fukushima, M.; et al. 2013. Emergency response to the nuclear accident at the Fukushima Daiichi Nuclear Power Plants using mobile rescue robots. *JFR* 30(1): 44–63.
- Otsu, K.; Tepsuporn, S.; Thakker, R.; Vaquero, T. S.; Edlund, J. A.; Walsh, W.; Miles, G.; Heywood, T.; Wolf, M. T.; and Agha-mohammadi, A. 2020. Supervised autonomy for communication-degraded subterranean exploration by a robot team. In *IEEE Aerospace Conference*.
- Pathak, D.; Agrawal, P.; Efros, A. A.; and Darrell, T. 2017. Curiosity-driven Exploration by Self-supervised Prediction. In *arXiv preprint arXiv:1705.05363*.
- Pineau, J.; Gordon, G.; and Thrun, S. 2003. Point-based value iteration: An anytime algorithm for POMDPs. In *IJCAI*, 1025–1032.
- Savinov, N.; Raichuk, A.; Marinier, R.; Vincent, D.; Pollefeys, M.; Lillicrap, T.; and Gelly, S. 2018. Episodic Curiosity through Reachability. In *arXiv preprint arXiv:1810.02274*.
- Silver, D.; and Veness, J. 2010. Monte-Carlo planning in large POMDPs. In *NeurIPS*, 2164–2172.
- Somani, A.; Ye, N.; Hsu, D.; and Lee, W. S. 2013. DESPOT: On-line POMDP Planning with Regularization. In *NeurIPS*, 1772–1780.
- Tao, T.; Huang, Y.; Sun, F.; and Wang, T. 2007. Motion planning for slam based on frontier exploration. In *International ICMA*, 2120–2125.
- Umari, H.; and Mukhopadhyay, S. 2017. Autonomous robotic exploration based on multiple rapidly-exploring randomized trees. In *IEEE/RSJ IROS*, 1396–1402.
- Vien, N. A.; and Toussaint, M. 2015. Hierarchical Monte-Carlo Planning. In *AAAI*.
- Yamauchi, B. 1997. A frontier-based approach for autonomous exploration. In *IEEE International Symposium on CIRA*, 146–151.