Autonomous Spot:
Long-Range Autonomous Exploration of
Extreme Environments with Legged Locomotion

Amanda Bouman*[2], Muhammad Fadhil Ginting*[1], Nikhilesh Alatur*[2], Matteo Palieri[1],
David D. Fan[1], Thomas Touma[2], Torkom Pailevanian[1], Sung-Kyun Kim[1], Kyohei Otsu[1],
Joel Burdick[2], and Ali-akbar Agha-Mohammadi[1]

Abstract— This paper serves as one of the first efforts to enable large-scale and long-duration autonomy using the Boston Dynamics Spot robot. Motivated by exploring extreme environments, particularly those involved in the DARPA Subterranean Challenge, this paper pushes the boundaries of the state-of-practice in enabling legged robotic systems to accomplish real-world complex missions in relevant scenarios. In particular, we discuss the behaviors and capabilities which emerge from the integration of the autonomy architecture NeBula (Networked Belief-aware Perceptual Autonomy) with next-generation mobility systems. We will discuss the hardware and software challenges, and solutions in mobility, perception, autonomy, and very briefly, wireless networking, as well as lessons learned and future directions. We demonstrate the performance of the proposed solutions on physical systems in real-world scenarios.[3] The proposed solution contributed to winning 1st-place in the 2020 DARPA Subterranean Challenge, Urban Circuit.[4]

Fig. 1: Live-mission image of Autonomous Spot robot climbing down four flights of stairs in the Urban Circuit of the DARPA Subterranean Challenge. This platform is one of the elements in team CoSTAR's solution that won the Urban Circuit of this competition. (Image credit: DARPA).

I. INTRODUCTION

Autonomous robot mapping and traversal of extreme environments under time constraints has a wide variety of real-world applications, including search and rescue after natural disasters [1], exploration of extreme planetary terrains [2], [3], [4], and inspection of urban underground environments [5]. As a concrete mission, we focus on the DARPA Subterranean (SubT) Challenge [6]: a robotic competition that targets missions to explore, map, and search extreme underground environments.

Extreme terrains typically involve mobility-stressing elements that can impose conflicting requirements on the development of mobility systems. For example, in the context of the SubT challenge, the systems need to 1) be small enough to move through passages as narrow as 80 cm in diameter while carrying a large-enough payload capable of providing high-levels of sensing, autonomy, computing, and communication capabilities, and 2) remain operational for long-duration missions ($\geq$ 1 hour) while actively exploring large areas (multi-kilometer in length) that require traversal of mobility-stressing features, such as stairs, uneven terrain, and risky, obstacle-laden areas.

Legged robots offer unique mobility capabilities which make them highly suitable for traversing challenging environments that would prove difficult for wheeled robots, as they have the ability to meet locomotion, size, payload, and endurance requirements to operate in extreme environments. For some prominent examples, see: ANYmal [7], [8], Robosimian [9], DRC-HUBO+ [10], Nimbro Momaro [11], MIT Cheetah [12], BigDog [13], Ghost Robotics Vision 60 [14].

The robotics research community is now in the early stages of empowering legged robots with high levels of autonomy to carry out complex missions in challenging, real-life environments [15]. Ramezani et al. [16] equipped the ANYmal quadruped with a LiDAR SLAM framework for autonomous mapping capabilities. The solution in [16] requires manual teleoperation to build an initial map of the environment, upon which the robot can autonomously navigate within the constructed map. The method is demonstrated in an industrial complex.

Bayer et al. [17] demonstrated fully autonomous exploration in rough, single-level, indoor and outdoor terrains. The researchers augmented an experimental hexapod platform with commercial vision sensors which were used for localization and terrain mapping. Miller et al. [14] endowed a Ghost Vision 60 quadruped with higher levels of autonomy to explore a tunnel environment during the 2019 DARPA Subterranean Challenge, Tunnel Circuit. They present one of the first efforts in autonomous legged exploration of an
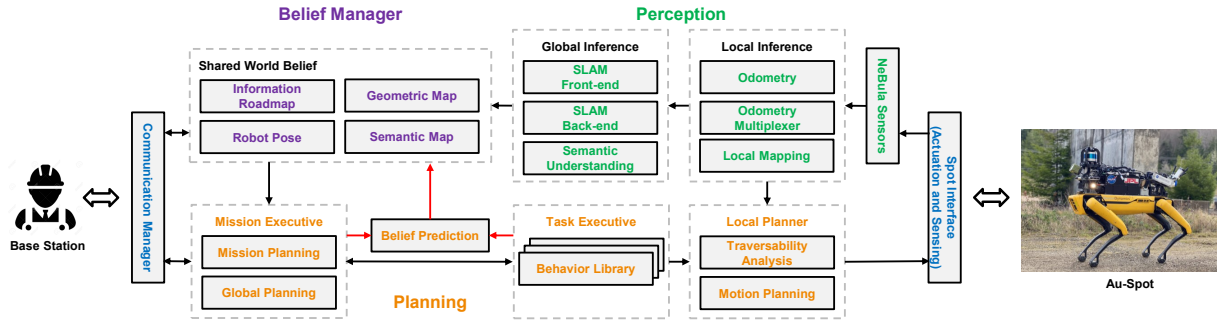
---

* These authors contributed equally to this work.
[1] NASA Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA, USA, {firstname.lastname}@jpl.nasa.gov
[2] Department of Mechanical and Civil Engineering, Division of Engineering and Applied Science, California Institute of Technology, Pasadena, CA, USA, {abouman,jwb}@caltech.edu
[3] Selected mission clips: https://youtu.be/MSKEaPtYOLI
[4] CoSTAR Team Website: https://costar.jpl.nasa.gov

Fig. 2: Overview of the NeBula system architecture enabling high-level autonomy on Spot. Red paths denote NeBula's belief-aware planning where the planner aims to minimize mission risk by generating information-seeking paths that increase the accuracy of the shared world belief.

unknown, GPS-denied subterranean environments, focused on single-level, tunnel-like environments.

Contributions: In this work, we focus on Boston Dynamics' Spot robot as our base mobility platform (Fig. 1). We briefly discuss the NeBula (Networked Belief-aware Perceptual Autonomy) architecture and explain some of the key elements of integrating comprehensive autonomy with the Spot robot. We describe the behaviors and overall performance of the system in a complex, autonomous mission during the Urban Circuit of the DARPA Subterranean Challenge. While the main objective of this paper is to provide a system-level overview of the entire autonomy stack, we will describe in deeper detail some specific aspects of the algorithms that are critical to enabling legged autonomy in complex missions.

Highlights of this paper or areas where we advance the current state-of-practice on Spot and legged robots are:

1) Endowing a legged platform with high-level autonomy so that it may traverse kilometer-scale distances in a multi-level, underground, GPS-denied environment within 60 minutes.
2) Enabling reliable multi-sensor odometry in perceptually-degraded environments.
3) Demonstrating perception- and traversability-aware local planning on legged platforms to negotiate challenging terrains and perceptually-degraded environments.
4) Developing a rugged and lightweight hardware system to equip Spot with the NeBula autonomy package.

The performance of these technologies was successfully field-tested at the Urban Circuit of the DARPA SubT Challenge (and practice runs leading to the competition), as part of team CoSTAR's solution.

Outline: In Section II, we provide an overview of the NeBula architecture and describe its elements. In Section III, we discuss the legged mobility system and the hardware payload. Sections IV, V, and VI focus on selected algorithmic aspects of legged robot odometry, local planning, and high-level mission planning. Experimental results are presented in Section VII, followed by future work discussion and conclusions.

II. NEBULA AUTONOMY

Motivated by autonomous exploration of extreme surfaces and subsurface terrains on the Moon, Mars and other plane-

tary bodies, NASA's Jet Propulsion Laboratory (NASA JPL) is developing an autonomy architecture referred to as NeBula (Networked Belief-aware Perceptual Autonomy). The main focus of NeBula is to provide computationally tractable methods to predict and assess various outcomes and risks in uncertain settings. These methods subsequently enable reliable, coordinated multi-robot exploration of unknown and hard-to-access terrains. To deal with uncertainty in unknown environments, NeBula employs a probabilistic approach. It takes the uncertainty into account to probabilistically fuse various sensing modalities, creates a probabilistic representation of the robot's knowledge of the environment, computes risk, and "proactively" plans to minimize the mission risk.

Architecture: Figure 2 illustrates a high-level overview of the NeBula architecture and how its modules are interconnected. Spot interface module, which includes Spot's internal locomotion system and inbuilt factory sensors, and NeBula's sensors will be discussed further in Section III. The odometry module, responsible for measuring and estimating the state and relative motion of the robot, will be discussed in Section IV. The belief manager block constructs and maintains the robot's model of the environment. The planning blocks include the 1) mission planning module that switches between various behaviors such as exploration, stair-climbing, communication-recovery, etc., 2) global planning which guides the coverage behavior and 3) traversability analysis and local motion planning. We will briefly discuss the planning modules in Sections V and VI. The communication block is responsible for enabling data exchange between multiple robots and a base station (described in [18]).

The belief prediction module is a critical component in the NeBula architecture that enables perception-aware and uncertainty-aware planning. This module allows the planner to take perceptual capability into account and helps reduce the risk by increasing the accuracy of the world representation (red arrows in Fig. 2). We will discuss this feature of NeBula further throughout the paper.

III. AU-SPOT MOBILITY SYSTEM

Locomotion System: Spot is a quadrupedal robot developed by Boston Dynamics to provide mobility on challenging terrain, which may not be negotiated by traditional wheeled robots, including steps and stairs, among others.
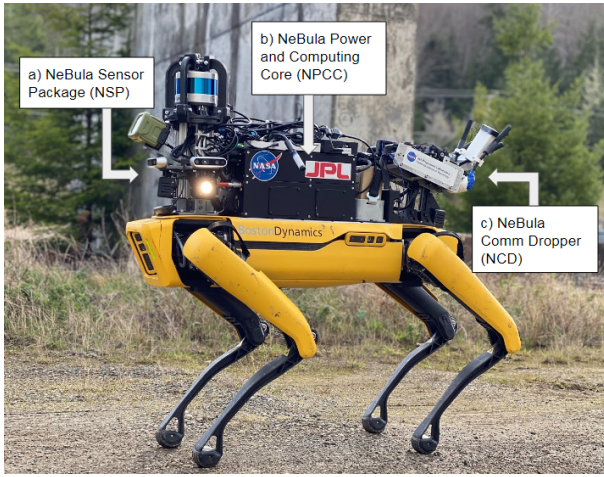
Fig. 3: "Autonomous Spot:" Spot powered by NeBula (Au-Spot).

Fig. 4: Architecture of the NeBula Multi-Sensor Fusion Framework

Sensing system: Spot's factory perception package from Boston Dynamics comprises five custom RealSenses distributed around the robot. To enable higher levels of autonomy required in the SubT challenge, we augment Spot's in-built sensing package with NeBula Sensor Package (NSP). The NSP includes a LiDAR, Intel RealSense cameras, high-intensity LEDs, an IMU, gas and wifi detectors, and a thermal camera. These sensors are integrated into a shock-absorbing, rigid mechanical super structure. The NSP can experience significant forces, moments, and vibrations as Spot negotiates complex terrain at high speeds. A combination of hard resin urethane, semi rigid carbon-infused nylon, and aluminum are used in the manufacturing process for increased structural integrity, sensor data fidelity and lightweight build. Further, the design takes into consideration atypical load paths for shock absorption during falls.

Power and Computing: The NeBula Power and Computing Core (NPCC) is designed to mount onto Spot as an auxiliary payload which provides power to all NeBula sensors and computers used for autonomy. The payload enclosure is designed with aluminum to provide protection to the internal electronics if Spot were to fall. The payload is powered from an external lithium high capacity battery to provide isolation and extended battery life for Spot's internal battery. The NPCC also features a custom power distribution and safety module, which provides fuses, overcurrent protection, overvoltage protection, inrush current limiting and power sequencing of five high efficiency voltage regulators for the sensors, lights, and computers. The payload uses two high-power computers for sensing, autonomy, and semantic scene understanding. For brevity, in the rest of this paper, we refer to the combined NSP, NPCC, and Spot robot as Autonomous-Spot or Au-Spot (Fig. 3).

IV. NeBula Odometry on Legged Systems

To enable autonomous robotic operation in extreme environments, a reliable odometry source is a prerequisite. In such scenarios, darkness, presence of obscurants (e.g. dust, fog, smoke), 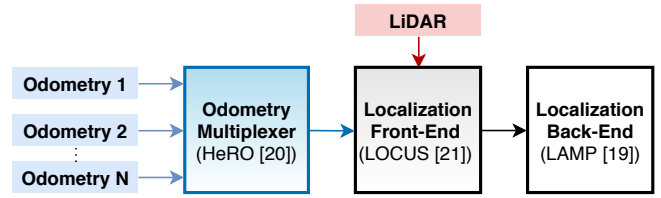self-similar areas, and strong platform vibrations caused by mobility-stressing terrains are common features which pose severe challenges to robotic perception. As a result, accurate odometry estimation on legged platforms is a critical challenge, especially under perceptually-degraded conditions.

Uneven and slippery areas make inertial sensing inaccurate while the material composition of the surface where the legged robot is walking on (e.g soft moquette, hard concrete) has strong impacts on the accuracy of kinematic-based odometry (KO). Darkness, or sudden excessive change in illumination, along with dust and the occasional presence of fog and gas, pose significant challenges to cameras. Potential visual aliasing phenomena in texture-less or texture-repetitive environments make feature-tracking problematic, decreasing the overall reliability of vision-based odometry (VO). Self-similar environments with repetitive geometry and lack of distinctive landmarks make scan-matching based methods ambiguous and prone to drift: moreover, active stereo cameras (including the in-built factory ones on the Spot platform) have a limited field of view, which renders them insufficient for our long-range perception applications.

Solution Architecture: To overcome these challenges, NeBula relies on a LiDAR-centric uncertainty-aware, multi-sensor fusion framework where a *selected* odometry source is fused as a prior with LiDAR information to enable accurate ego-motion estimation under challenging perceptual conditions. The main components of the proposed approach are: (i) an anomaly-aware odometry multiplexer (HeRO), (ii) a multi-sensor LiDAR-centric SLAM front-end (LOCUS) and (iii) a SLAM back-end (LAMP) [19]. Fig. 4 provides a high-level overview of the proposed approach. We discuss each component in the following.

Odometry Multiplexer: To select the best odometry prior to be fused with LiDAR information, we feed multiple and heterogeneous sources of odometry available onboard (e.g., KO, VO, etc.) into an anomaly-aware odometry multiplexer, referred to as HeRO [20]. At every time step, HeRO runs a confidence test on each odometry stream (prior) to detect potential anomalies (e.g., gaps, jumps, divergences) and identify the most reliable input $\mathbf{Y} \in SE(3)$ to be used as a prior in the LiDAR-based front-end.

Localization Front-End: The output of the odometry multiplexer is fed into a multi-sensor LiDAR-centric SLAM front-end module, referred to as LOCUS [21] that performs a cascaded GICP-based scan-to-scan and scan-to-submap matching operation to estimate the relative motion of the robot between consecutive LiDAR acquisitions. Let L_k denote the LiDAR scan acquired the k-th time step. We

indicate with $\mathbf{E}_k^{k-1} = \mathbf{Y}_{k-1}^{-1}\mathbf{Y}_k \in SE(3)$ the rigid body transformation of HeRO's output between two consecutive LiDAR acquisitions.

In the scan-to-scan matching stage, GICP computes the optimal transformation $\hat{\mathbf{T}}_k^{k-1}$ that minimizes the residual error $\mathcal{E}$ between corresponding points in L_{k-1} and L_k.

$$\hat{\mathbf{T}}_k^{k-1} = \arg\min_{\mathbf{T}_k^{k-1}} \mathcal{E}(\mathbf{T}_k^{k-1}L_k, L_{k-1}) \tag{1}$$

To enhance accuracy we initialize the optimization with $\mathbf{E}_k^{k-1}$. In the case where no input is received by HeRO, we rely on the identity transformation as the prior and the system reverts to pure LiDAR odometry.

To enable global consistency across the history of scans, the motion estimated in the scan-to-scan matching stage is further refined by a scan-to-submap matching step. Here, L_k is matched to a local submap S_k which is a robot-centered subset of the global map. The global map is composed of accumulated past point clouds aligned to the robot pose in the world frame.

$$\tilde{\mathbf{T}}_k^{k-1} = \arg\min_{\mathbf{T}_k^{k-1}} \mathcal{E}(\mathbf{T}_k^{k-1}L_k, S_k) \tag{2}$$

The initial guess of this optimization is $\hat{\mathbf{T}}_k^{k-1}$, which results from Eq. (1). After scan-to-scan and scan-to-submap matching, the final estimated motion $\tilde{\mathbf{T}}_k^{k-1}$ between consecutive LiDAR acquisitions is used to update the robot pose in the world.

Localization Back-End: The odometry produced by the front-end is fed into the back-end of our SLAM system, referred to as LAMP [19] which receives pose-to-pose constraints and solves a Pose Graph Optimization (PGO) and Incremental Consistency Measurement (ICM) problem for global localization when loop closures are detected during traversal.

V. Local Planning

This section describes our approach to enable Spot to traverse challenging environments, where assessing the traversability risk and planning safe paths is a prerequisite for autonomous navigation.

A. Traversability Map

We model the traversability of the environment as a grid map $g = (m^1, \cdots, m^n)$ with $n = n_l \times n_w$ cells, arranged in a *locally* 2D grid, where $m^i \in \{safe, lethal\}$ is a Bernoulli random variable. $m^i = safe$ represents the event that the robot can safely navigate through the i-th cell. We infer the probability distribution $p(g)$ over grid g and store it as a traversability map. The steps involved in computing $p(g)$ are detailed in this subsection.

Risk Sources: There is a variety of traversability-stressing elements which increase the probability of failure during traversal. These elements can be purely geometric (e.g., positive obstacles, negative obstacles, steep slopes) or of
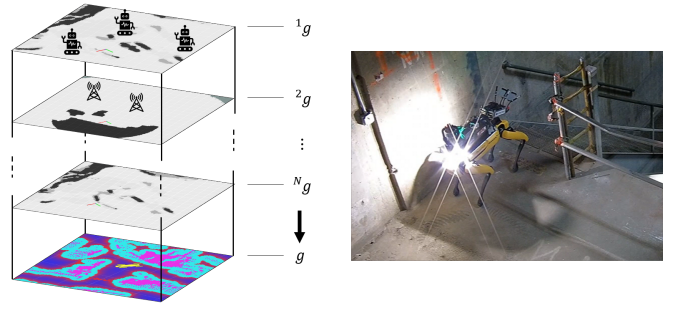

Fig. 5: The multi-layer traversability map $(^1g, \cdots, ^Ng)$, which includes information about various traversability-stressing elements (including other robots in ^1g, communication nodes in ^2g, large negative and positive obstacles in ^Ng, etc.). The bottom map illustrates the aggregated traversability map g.

semantic nature (mission items such as deployed communication nodes or other robots).

Multi-Fidelity Terrain Map: For detecting the aforementioned traversability-stressing elements in unknown environments, we build a local terrain map of the robot's surrounding, via data collected from NSP. Specifically, we build a multi-fidelity map at various ranges. We use depth cameras for short-range sensing, instantaneous LiDAR point clouds for medium-range sensing, and spatially fused point clouds [22] for long-range detection. The combination of these various sensing capacities yields an efficient trade-off among *range*, *density* and *accuracy* in the resulting terrain map.

Multi-Layer Traversability Map (MLT): To capture various traversability-stressing elements, we construct a layered representation (Fig. 5) of the traversability map $g = \{^1g, ..., ^Ng\}$, where each layer $^\ell g$ captures a certain risk source (as explained above). At each layer ℓ, we apply algorithms and perform an analysis relevant to the corresponding risk element. Specifically, positive and negative obstacles as well as steep slopes are detected on the dense short-range by applying a step filter relative to the local ground plane. On the medium- and long-range: *(i)* Positive obstacles are detected by performing ground segmentation [23] and settling-based collision checks [24], *(ii)* Negative obstacles by searching for surface discontinuities (holes) in the LiDAR point cloud, and *(iii)* steep slopes by using settling methods such as [24]. Mission items (e.g., deployed communication nodes and other robots) can be detected semantically and superposed on the traversability map.

Fast Traversability Evaluation: To enable online generation of MLT as the robot moves, we perform traversability evaluation only on a representative and sparse set of sampled cells $Q = \{i_s\}_{s=0}^S$ in the grid map. On each sampled cell $i \in Q$, the traversability risks $p(^\ell m^i = lethal)$ are computed and stored in all layers $\ell \in N$. To compute the traversability risk at every cell in the grid, we approximate the MLT at a higher resolution by interpolating with a Gaussian kernel or by adding inflation with a radial decay to each query cell $i \in Q$ along the spatial dimension. Thus, we compute the traversability $p(^\ell m^n = safe)$ for all cells n in all layers ℓ.

Superposition: For action generation, we create a single traversability map g by fusing the N different layers $\{^1g, ..., ^Ng\}$. The information in these layers are not independent in general. We approximate a conservative risk estimate by element-wise multiplication of traversability probabilities:

$$p(m^i = safe) = \prod_{\ell=1}^{N} \mathbb{P}(^\ell m^i = safe) \ \forall i = 1, \cdots, n \quad (3)$$

The bottom layer in Fig. 5 illustrates an example traversability map (plotting the maximum likelihood estimation of $p(g)$), obtained during a real mission.

B. Uncertainty and Perception-aware Planning

To enable traversability through narrow passages and obstacle-laden environments, one needs to reduce the map uncertainty. We address this challenge by taking into account uncertainty which comes from noisy, asymmetric, and limited sensors in order to find trajectories with minimal path length that also reduce uncertainty in the map $p(g)$ so as to increase the probability that the path taken will be safe. In the NeBula architecture, we formalize this notion with an uncertainty-aware planner. This general framework can be used to create behaviors which intelligently reduce risk coming from uncertainties in sensing and the environment. We outline the general framework here.

Uncertainty-aware Representation: Let μ^i be the mean of the Bernoulli distribution of m^i. Then we can model the distribution (or our belief) of μ^i with its parameters, $p(\mu^i | \hat{\mu}^i, \sigma^i)$ (e.g. a mean $\hat{\mu}^i$ and variance σ^i in the case of a beta distribution). The "confidence" about μ^i is captured in σ^i, where fully unknown and fully known cells have the highest and lowest σ^i values, respectively [25].

Map prediction: The uncertainty-aware representation allows us to incorporate perceptual capabilities into the planning. We define a policy π that returns an ordered sequence of grid locations that the robot visits, along with the orientations of the robot at those locations: $\pi(\cdot) = \{i_k, \theta_k\}_{k=0}^{K}$. Given the sensors available on the robot and their configuration and noise characteristics, along with a given policy π, we update our belief of the traversability values in a recursive manner, which we call τ:

$$(\hat{\mu}_k^i, \sigma_k^i) = \tau(\hat{\mu}_{k-1}^i, \sigma_{k-1}^i, z_k(\pi)) \quad (4)$$

where the measurement $z_k(\pi)$ is predicted from a generative model, at the k-th time step along the trajectory π. This becomes increasingly important when the sensor configuration is highly asymmetric on a robot, which is the case for Spot as it has blind spots and areas where sensory measurement noise is considerably higher than other areas. Maintaining separate probability distributions for individual cells in the map, we predict the map $p(g)$ for k-th time step into the future as follows:

$$g_k \equiv \{(\hat{\mu}_k^1, \sigma_k^1), \cdots, (\hat{\mu}_k^n, \sigma_k^n)\} \quad (5)$$

Risk-Aware/Perception-Aware Planning: Next we define a risk measure that takes perceptual capabilities and uncertainties into account when planning trajectories. We also define an optimal policy π^* as the policy which minimizes the total path risk $\mathcal{R}^\pi$ along the K-step path while moving towards a given goal:

$$\mathcal{R}^\pi = 1 - \prod_{k=0}^{K} p(m^{i_k} = safe | \hat{\mu}_k^i, \sigma_k^i, \pi) \quad (6)$$

$$\pi^* = \arg \min_{\pi \in \Pi} \mathbb{E}[\mathcal{R}^\pi] \quad (7)$$

Efficient methods for computing predicted risk uncertainty over a 2-D grid for a given sensor model have been considered in [26]. When computational complexity is a challenge (with respect to the robot speed), to enable efficient real-time computation of Eq. (7), we rely on a cascaded policy, where one can optimize for position and orientation (along the path) sequentially.

Execution: We execute the planning problem outlined in Eq. 7 in a receding-horizon control (RHC) fashion: We optimize for a K-step policy. Then, when executing the policy, we select a waypoint at a distance d along the path π^*, send it to the robot, and while robot is moving towards the waypoint, we resolve the path planning problem to generate a new path from the new robot position. Selecting d is a trade-off between smoothness and tracking error, where a larger d improves stability and smoothness, while a smaller d keeps tracking error of π^* lower. The combined effect of perception-aware planning and RHC-based execution will cause Au-Spot to prefer moving in directions that lead to richer sensory input, which leads to a larger reduction in uncertainty via collecting more task-relevant information.

VI. AREA COVERAGE AND SEARCH BEHAVIOR

Our mission planning objective is to coordinate a team of autonomous robots to rapidly map and navigate a large (multi-Km), unknown environment characterized by complex topology and terrain under a one-hour time constraint. The global planner realizes this objective by maximizing the area covered by the sensors' footprint along the planned trajectory, within the mission time limit. The planner relies on a representation of the environment with an information-rich graph structure which reduces the policy search space to one that is tractable when exploring large environments over long time horizons.

Global Planner: We employ a sparse bidirectional graph $G = (V, E)$ that captures the connectivity of the free space in the environment (e.g., [27]). A node $v_i \in V$ represents a robot pose, and an edge $e_{ij} \in E$ represents a robot transition between connected nodes v_i and v_j. Each node v_i has attached to it a feature vector containing the probability $p_c(v_i)$ that the robot has seen a sensor-defined neighborhood around the node. Likewise, each edge e_{ij} will induce a local path computed according to Eq. 7. Then, to each edge e_{ij} we attach a feature vector containing the path length $\ell_{e_{ij}}$ and path traversability risk $\mathcal{R}^{e_{ij}}$ computed by Eq. 6.

Graph Construction: We partition graph nodes into frontier nodes $v^f \in V$ and breadcrumb nodes $v^b \in V$. Frontier nodes v^f indicate the areas of the map that have not been fully explored yet, typically, at the boundary of the known

and unknown free spaces. A breadcrumb node v^b indicates the areas of the map that have already been fully explored. As the robot moves in the environment to cover and search the space, the graph is expanded through the addition of new breadcrumb and frontier nodes. In other words, visiting a frontier node v^f is expected to lead to changes in the map belief $p(v^f)$, whereas visiting a breadcrumb node will not significantly impact our knowledge of the map coverage.

Graph Policy: A graph policy λ guides the robot towards the next best node on the graph to maximize the covered area. Specifically, we compute a mapping (feedback policy) $\lambda : V \to V$ on the graph which maximizes a reward function that encodes a trade-off between coverage information and traversal cost. A macro action $\lambda(v_i)$ induces traversal along edge e_{ij} and updates the graph from V to V'. In the following, we discuss steps involved in computing λ.

Coverage Information: The coverage belief of a graph composed of nodes $\{v_i, \cdots, v_N\} \in V$ is defined as:

$$P_c(V) = \{p_c(v_i), \cdots, p_c(v_N)\}$$

where $p_c(v_i)$ is the occupancy Bernoulli distribution over a local map centered at node v_i. We use entropy to measure the posterior uncertainty of the graph coverage. Entropy of a random variable $x \sim p(x)$ is defined as $H_p(x) = E[-\log p(x)]$. Thus, the graph coverage entropy can be represented as:

$$H_{p_c}(V) = -\sum_{v_i \in V} p_c(v_i) \log p_c(v_i) + (1 - p_c(v_i))\log(1 - p_c(v_i))$$

Coverage Information Gain: The coverage information gain (i.e., coverage uncertainty reduction) in belief $p_c(V)$ induced by macro action $\lambda(v_i)$ is defined as:

$$I(V \mid \lambda(v_i)) = \underbrace{H_{p_c}(V)}_{\text{current entropy}} - \underbrace{H_{p_c}(V' \mid \lambda(v_i))}_{\text{future entropy}} \quad (8)$$

where the second term represents the expected future entropy of the graph after the agent has executed macro action $\lambda(v_i)$.

Generalized Reward: To capture energy and distance measures in the coverage planning, we define the one-step reward to be the weighted sum of information gain and distance traveled under the macro action $\lambda(v_i)$:

$$Rew(v_i, \lambda(v_i)) = w_1 I(V \mid \lambda(v_i)) - w_2 \ell_{e_{ij}} \quad (9)$$

where w_1 and w_2 weigh the information gain and traveled distance, respectively.

Graph Policy Optimization: Let $v(k)$ denote the k-th node visited under graph policy λ. Similarly, let $e(k, k+1)$ denote the edge between nodes $v(k)$ and $v(k+1)$. We solve for the mapping λ that maximizes the sum of future expected reward:

$$\lambda^* = \arg\max_{\lambda(\cdot)} \mathbb{E}[\sum_k Rew(v(k), \lambda(v(k)))] \quad (10)$$

This optimization can be solved via value iteration-based methods or forward search methods.
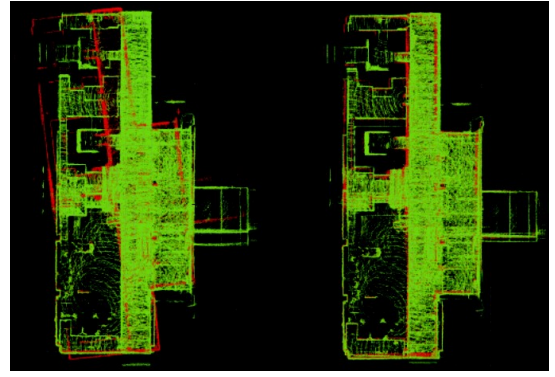

Fig. 6: Map created by Au-Spot exploring Eagle Rock Substation, with different odometry sources: proposed method in green against KO in red (left) and KVO in red (right).
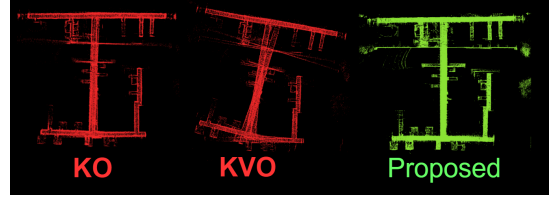

Fig. 7: Map created by Au-Spot while exploring an office building at NASA's JPL with different odometry sources: KO (left), KVO (middle), and the Proposed method (right).

VII. Experimental Results

The NeBula autonomy architecture is implemented on two Boston Dynamics Spot robots and field-tested in subsurface, multi-level, and perceptually-degraded GPS-denied environments, including underground unstructured environments and industrial power plants.

As part of the Urban Circuit of the DARPA Subterranean Challenge, two Au-Spots were deployed into an Industrial Plant in February 2020 for live-mission exploration and 3D-Mapping of the environment (Fig. 9). The missions included detection of artifacts of interest such as backpacks, human survivors, gas-leaks, and cellphones via different sensing modalities including vision, thermal, gas sensors and wifi detection among others. The competition divided into four live-missions in unknown environments. Our Au-Spots explored a combined distance of 4km, including successfully climbing multiple flights of stairs. Points were awarded for accurate artifact detection and for successfully reporting this information back to the base-station. The NeBula framework successfully detected and localized a total of 16 artifacts, giving team CoSTAR a 1st-place win.

Odometry Estimation: To demonstrate the performance of the proposed odometry pipeline, we compare the localization accuracy achievable using individual sensing channels with the proposed uncertainty-aware multi-sensor approach in perceptually-degraded environments. Figure 6 depicts the results of the proposed method on data collected in the Eagle Rock subway station, Los Angeles. The rough and varying terrain causes KO to perform poorly, while a large amount of visual features causes KVO to produce a sufficiently accurate
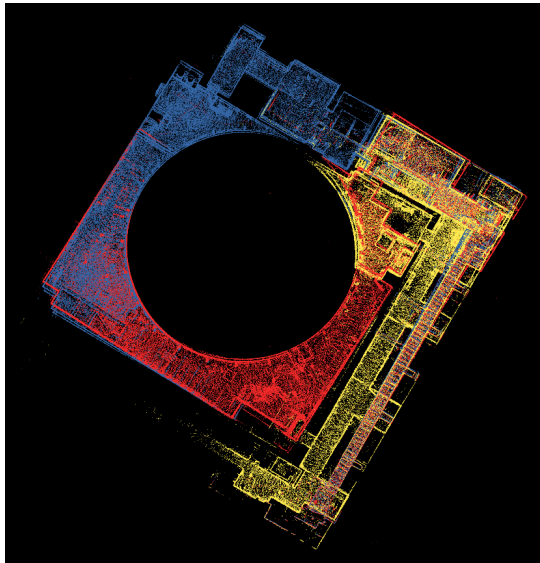
Fig. 8: Top view of area covered in urban environment by robot fleet consisting of one Au-Spot [yellow] and two UGVs [red/blue] during one run of the SubT Challenge, Urban Circuit. The map contains three different floors connected by multi-flight staircases. Note the complex topology (narrow passages, varying-sized interconnected rooms, and the outer circular geometry) requiring an irregular exploratory behavior that is characterized by a large looping path combined with many auxiliary paths necessary for the inspection of smaller structures.
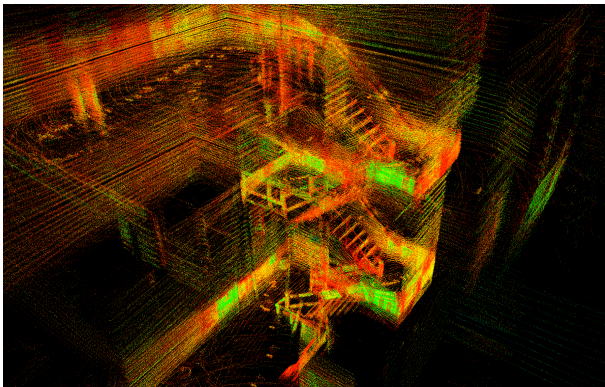

Fig. 9: 3D map generated by NeBula while traversing four flights of stairs. The map was created in real-time during the run (see Fig. 1).

map. Conversely, Fig. 7 depicts results from data collected in NASA JPL's 198/161 offices. In this case, the soft carpet on the floor results in KO providing much more accurate maps than KVO. KVO is instead challenged by feature-less white walls and the repetitive visual textures of the carpet. The different features of various environments make perception and odometry estimation challenging to a single sensing channel alone, hence the need for multi-channel odometry fusion. As seen in both figures, the proposed odometry generation method results in more accurate maps than those obtained by KO or KVO-based odometry.

Traversability: Our perception-aware local planner enabled Au-Spot to safely and efficiently navigate through difficult environments, which contained a variety of unstructured obstacles and terrain, including ramps, slopes, piles of twisted metal, machinery and rubble (Fig. 10). Negative

Fig. 10: Narrow corridors (Top Left), water/oil (Top Right), stair wells (Bottom left), and raised concrete slabs (Bottom Right) are examples of some of the difficult terrain successfully navigated by Au-Spot. For full mission video, see [28].

obstacles such as dropoffs and holes were also successfully avoided.

Coverage Planner: Au-Spot's coverage planner successfully guided a fleet of two Au-Spots and two wheeled UGV robots to collaboratively explore and map a large unstructured environment within the one hour time limit in the SubT Challenge. Fig. 8 depicts the area explored by the robots during one of the four live-mission runs. In this single (one hour) run, the fleet of robots mapped a total volume of approximately 25,000 m^3. One of the most challenging elements of the course was traversing 4 flights of stairs, which induces pitching motions, physical slippage on the stair edges, and poor visual cues due to repetitive patterns of the stairs and railings. Fig. 1 shows Au-Spot successfully climbing down stairs, and Fig. 9 depicts the map produced during the stair climbing operations, which allows the robot to accurately localize artifacts on multiple levels of the industrial power plant.

Complex Mission: The video in [28] depicts Au-Spot navigating and mapping one of the courses in the Urban Circuit of the DARPA SubT Challenge under time, communication, and computation constraints. The video begins as Au-Spot leaves the staging area where the human supervisor sends the "go-command". Thereafter, Au-Spot's behavior at various phases of the mission, including when it is searching for artifacts, descending stairs, and deploying communication nodes, are presented. The video highlights how autonomous traversability and coverage planning enables the robot to thoroughly explore the environment, which includes challenging features such as hallways with narrow openings, large open rooms, raised platforms, and rubble. Over more than 1 km travel distance, the system was able to detect and localize artifacts while maintaining a localization error of less than 5 m – the maximum artifact localization error to score points in the SubT Challenge.

VIII. Conclusions

Motivated by exploring extreme environments and in particular underground environments in DARPA Subterranean Challenge, this system-focused paper discusses our developments toward endowing legged robots with hardware and perception capabilities required for high-levels of autonomy in extreme environments. Specifically, we have presented our NeBula autonomy architecture applied to Boston Dynamics' Spot robot. NeBula is an architecture for risk- and perception-aware autonomy, applicable to a wide range of robots. In this paper, we have discussed a few representative NeBula modules, including odometry, traversability, and coverage planning, pertaining to the DARPA Subterranean Challenge. We believe this work takes an important step in advancing the state-of-the-practice and demonstrates the capabilities of legged robots for accomplishing complex, real-world, live-missions in extreme environments.

References

[1] K. Nagatani, S. Kiribayashi, Y. Okada, K. Otake, K. Yoshida, S. Tadokoro, T. Nishimura, T. Yoshida, E. Koyanagi, M. Fukushima, and S. Kawatsuma, "Emergency response to the nuclear accident at the fukushima daiichi nuclear power plants using mobile rescue robots," *Journal of Field Robotics*, vol. 30, no. 1, pp. 44–63, 2013.

[2] A. Agha, K. L. Mitchell, and P. J. Boston, "Robotic Exploration of Planetary Subsurface Voids in Search for Life," in *AGU Fall Meeting Abstracts*, 2019.

[3] T. Sasaki, K. Otsu, R. Thakker, S. Haesaert, and A. Agha-mohammadi, "Where to map? iterative rover-copter path planning for mars exploration," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2123–2130, 2020.

[4] A. Husain, H. Jones, B. Kannan, U. Wong, T. Pimentel, S. Tang, S. Daftry, S. Huber, and W. L. Whittaker, "Mapping planetary caves with an autonomous, heterogeneous robot team," in *IEEE Aerospace Conference*, Big Sky, MT, 2013.

[5] H. Kolvenbach, D. Wisth, R. Buchanan, G. Valsecchi, R. Grandia, M. Fallon, and M. Hutter, "Towards autonomous inspection of concrete deterioration in sewers with legged robots," *Journal of Field Robotics*, pp. 1–14, 2020.

[6] DARPA Subterranean Challenge. [Online]. Available: https://www.subtchallenge.com

[7] R. Zimroz, M. Hutter, M. Mistry, P. Stefaniak, K. Walas, and J. Wodecki, "Why should inspection robots be used in deep underground mines?" in *International Symposium on Mine Planning and Equipment Selection*, 2019.

[8] C. D. Bellicoso, M. Bjelonic, L. Wellhausen, K. Holtmann, F. Günther, M. Tranzatto, P. Fankhauser, and M. Hutter, "Advances in real-world applications for legged robots," *Journal of Field Robotics*, vol. 35, no. 8, pp. 1311–1326, 2018.

[9] S. Karumanchi, K. Edelberg, I. Baldwin, J. Nash, J. Reid, C. Bergh, J. Leichty, K. Carpenter, M. Shekels, M. Gildner, D. Newill-Smith, J. Carlton, J. Koehler, T. Dobreva, M. Frost, P. Hebert, J. Borders, J. Ma, B. Douillard, P. Backes, B. Kennedy, B. Satzinger, C. Lau, K. Byl, K. Shankar, and J. Burdick, "Team RoboSimian: Semi-autonomous Mobile Manipulation at the 2015 DARPA Robotics Challenge Finals," *Journal of Field Robotics*, vol. 34, no. 2, pp. 305–332, 2017.

[10] T. Jung, J. Lim, H. Bae, K. K. Lee, H.-M. Joe, and J.-H. Oh, "Development of the humanoid disaster response platform DRC-HUBO+," *IEEE Transactions on Robotics*, vol. 34, no. 1, pp. 1–17, 2018.

[11] M. Schwarz, T. Rodehutskors, D. Droeschel, M. Beul, M. Schreiber, N. Araslanov, I. Ivanov, C. Lenz, J. Razlaw, S. Schüller, D. Schwarz, A. Topalidou-Kyniazopoulou, and S. Behnke, "NimbRo Rescue: Solving disaster-response tasks with the mobile manipulation robot Momaro," *Journal of Field Robotics*, vol. 34, no. 2, pp. 400–425, 2017.

[12] S. Seok, A. Wang, M. Y. Chuah, D. J. Hyun, J. Lee, D. M. Otten, J. H. Lang, and S. Kim, "Design principles for energy-efficient legged locomotion and implementation on the MIT Cheetah robot," *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 3, pp. 1117–1129, June 2015.

[13] M. Raibert, K. Blankespoor, G. Nelson, and R. Playter, "Bigdog, the rough-terrain quadruped robot," *IFAC Proceedings Volumes*, vol. 41, no. 2, pp. 10 822–10 825, 2008.

[14] I. D. Miller, F. Cladera, A. Cowley, S. S. Shivakumar, E. S. Lee, L. Jarin-Lipschitz, A. Bhat, N. Rodrigues, A. Zhou, A. Cohen, A. Kulkarni, J. Laney, C. J. Taylor, and V. Kumar, "Mine tunnel exploration using multiple quadrupedal robots," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2840–2847, 2020.

[15] J. Delmerico, S. Mintchev, A. Giusti, B. Gromov, K. Melo, T. Horvat, C. Cadena, M. Hutter, A. Ijspeert, D. Floreano, L. M. Gambardella, R. Siegwart, and D. Scaramuzza, "The current state and future outlook of rescue robotics," *Journal of Field Robotics*, vol. 36, no. 7, pp. 1171–1191, 2019.

[16] M. Ramezani, M. Brandao, B. Casseau, I. Havoutis, and M. Fallon, "Legged robots for autonomous inspection and monitoring of offshore assets," *Offshore Technology Conference*, 2020.

[17] J. Bayer and J. Faigl, "On autonomous spatial exploration with small hexapod walking robot using tracking camera intel realsense t265," in *2019 European Conference on Mobile Robots (ECMR)*, 2019.

[18] K. Otsu, S. Tepsuporn, R. Thakker, T. S. Vaquero, J. A. Edlund, W. Walsh, G. Miles, T. Heywood, M. T. Wolf, and A. Agha-mohammadi, "Supervised autonomy for communication-degraded subterranean exploration by a robot team," in *IEEE Aerospace Conference*, 2020.

[19] K. Ebadi, Y. Chang, M. Palieri, A. Stephens, A. Hatteland, E. Heiden, A. Thakur, B. Morrell, L. Carlone, and A. Agha-mohammadi, "LAMP: Large-scale autonomous mapping and positioning for exploration of perceptually-degraded subterranean environments," in *IEEE International Conference on Robotics and Automation*, 2020.

[20] A. Santamaria-navarro, R. Thakker, D. D. Fan, B. Morrell, and A. Agha-mohammadi, "Towards resilient autonomous navigation of drones," in *International Symposium on Robotics Research*, Hanoi, Vietnam, 2019.

[21] M. Palieri, B. Morrell, A. Thakur, K. Ebadi, J. Nash, L. Carlone, C. Guaragnella, and A. Agha-mohammadi, "LOCUS - A multi-sensor lidar-centric solution for high-precision odometry and 3D mapping in real-time," *(Under Review)*.

[22] H. Oleynikova, Z. Taylor, M. Fehr, R. Siegwart, and J. Nieto, "Voxblox: Incremental 3D euclidean signed distance fields for on-board mav planning," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Vancouver, BC, 2017.

[23] M. Himmelsbach, F. v. Hundelshausen, and H. . Wuensche, "Fast segmentation of 3D point clouds for ground vehicles," in *2010 IEEE Intelligent Vehicles Symposium*, 2010.

[24] P. Krüsi, P. Furgale, M. Bosse, and R. Siegwart, "Driving on point clouds: Motion planning, trajectory optimization, and terrain assessment in generic nonplanar environments," *Journal of Field Robotics*, vol. 34, no. 5, pp. 940–984, 2017.

[25] A. Agha-mohammadi, E. Heiden, K. Hausman, and G. Sukhatme, "Confidence-rich grid mapping," *The International Journal of Robotics Research*, vol. 38, no. 12-13, pp. 1352–1374, 2019.

[26] E. Heiden, K. Hausman, G. S. Sukhatme, and A. Agha-mohammadi, "Planning high-speed safe trajectories in confidence-rich maps," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Argentina.

[27] A. Agha-mohammadi, S. Chakravorty, and N. Amato, "FIRM: Sampling-based feedback motion planning under motion uncertainty and imperfect measurements," *International Journal of Robotics Research (IJRR)*, vol. 33, no. 2, pp. 268–304, 2014.

[28] "Video of Au-Spot exploring and mapping one of the courses in the DARPA SubT Challenge Urban Circuit," https://costar.jpl.nasa.gov and https://youtu.be/MSKEaPtYOLI.