

Autonomous Teamed Exploration of Subterranean Environments using Legged and Aerial Robots

Mihir Kulkarni^{1,3,*}, Mihir Dharmadhikari^{1,3,*}, Marco Tranzatto², Samuel Zimmermann², Victor Reijgwart², Paolo De Petris³, Huan Nguyen³, Nikhil Khedekar³, Christos Papachristos¹, Lionel Ott², Roland Siegwart², Marco Hutter², and Kostas Alexis³

Abstract—This paper presents a novel strategy for autonomous teamed exploration of subterranean environments using legged and aerial robots. Tailored to the fact that subterranean settings, such as cave networks and underground mines, often involve complex, large-scale and multi-branched topologies, while wireless communication within them can be particularly challenging, this work is structured around the synergy of an onboard exploration path planner that allows for resilient long-term autonomy, and a multi-robot coordination framework. The onboard path planner is unified across legged and flying robots and enables navigation in environments with steep slopes, and diverse geometries. When a communication link is available, each robot of the team shares submaps to a centralized location where a multi-robot coordination framework identifies global frontiers of the exploration space to inform each system about where it should re-position to best continue its mission. The strategy is verified through a field deployment inside an underground mine in Switzerland using a legged and a flying robot collectively exploring for 45 min, as well as a longer simulation study with three systems.

I. INTRODUCTION

The collective progress in robotic systems has enabled their utilization in a multitude of autonomous exploration and mapping missions. Aerial and ground robots are now employed in diverse search and rescue [1–4], industrial inspection [5–12], surveillance [13], planetary exploration [14–19] and other mission scenarios. Despite the progress, a set of environments remain particularly demanding for robots to autonomously explore and challenge the state-of-the-art in robot navigation and autonomy. Among them, subterranean settings such as underground mines, subway infrastructure and cave networks are especially strenuous. Motivated by the goals and aspirations of the DARPA Subterranean Challenge [20], this work culminates on a sequence of developments in autonomous exploration path planning for both legged and aerial robots that have enabled the robots of Team CERBERUS to explore underground mines, multi-level power plants, tunnels, sewers, caverns, lava tubes and more.

Specifically, we present two synergistic contributions, namely a) “GBPlanner2”, a revised and enhanced version

This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Agreement No. HR00111820045. The presented content and ideas are solely those of the authors.

¹University of Nevada, Reno, 1664 N. Virginia, 89557, Reno, NV, USA
mkulkarni@nevada.unr.edu

²ETH Zurich, Leonhardstrasse 21, 8092, Zurich, Switzerland

³NTNU, O. S. Bragstads Plass 2D, 7034, Trondheim, Norway

* The authors have contributed equally.

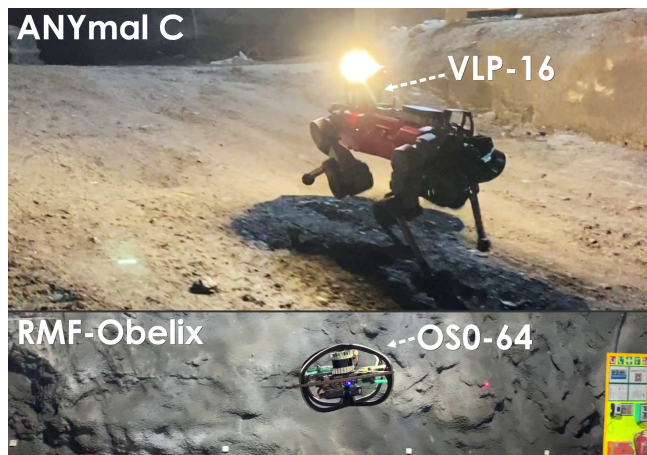


Fig. 1. Instances of autonomous exploration of a subterranean environment.

of our Graph-Based subterranean exploration path Planner [21] to uniformly guide both legged and aerial robots (open-sourced at https://github.com/ntnu-arl/gbplanner_ros), as well as b) the newly developed CoOperation for Heterogeneous Robot Teams (COHORT) framework that facilitates the synergistic exploration of large-scale, multi-branching, and diverse subterranean settings having both narrow and wide sections. The two methods working in harmony enable the autonomous cooperative exploration and mapping of complex underground domains using legged and flying systems that coordinate their exploration, while each retains complete individual exploratory capacity.

To verify the proposed solution for teamed exploration path planning, a set of evaluations are presented including a deployment in the Hagerbach underground mine in Switzerland (Figure 1) using legged and aerial robots exploring for 37 min and 8 min respectively, as well as a large-scale simulation study. We demonstrate that due to the individual properties of GBPlanner2 and COHORT, as well as their synergy, the robots present efficient autonomous exploration behaviors that are resilient against the complexities introduced by the vast scale of the environments, the challenging terrain and complicated geometries involved, as well as the difficulty of establishing reliable communications. A page of results is maintained at <https://s.ntnu.no/exploration>.

In the remaining paper, Section II presents related work, followed by the problem statement in III. The proposed approach is detailed in Section IV, with evaluation studies in Section V and conclusions in Section VI.

II. RELATED WORK

A rich body of work has focused on the problems of autonomous single and multi-robot exploration [21–35]. Early work in single-robot exploration included the sampling of “next-best-views” [25], and the detection of frontiers [26], while recent efforts have focused on powerful planning techniques such as random trees and graphs possibly combined with volumetric calculations [22, 23, 36–39], receding horizon techniques [23], multi-objective optimization [36, 37], information-theoretic schemes [40], learning-based methods [41], and approaches that account for the likelihood of accumulating localization drift [22, 42]. In multi-robot exploration, the seminal work in [30] presented a strategy for multi-robot coordination exploiting a grid map and a planning policy that tries to minimize the collective exploration time by considering both the cost of reaching a certain frontier cell and the “exploration utility” of each such cell as a function of the number of robots moving to that cell. Recent efforts have included distributed inference-based schemes [35], volumetric gain-driven multi-robot exploration [32], methods on the collaboration of ground and aerial systems [43], techniques that consider the effect of limited communication range and endurance [29, 31], concepts on how robots can work together to physically clear blocked paths [44] and more. The community has recently focused on the specific problem of subterranean exploration given the accelerating effects of the DARPA Subterranean Challenge. In response, groups around the world have proposed novel methods both for single- and multi-robot exploration. This includes techniques for quadruped robots [28], methods tailored to fast exploration using aerial platforms [27], schemes for both legged and flying systems [21], hierarchical frameworks to exploit dense local and sparse global information [45], multi-robot exploration strategies [34], approaches exploiting underground mine and cave topologies [46, 47], and significant field robotics work [48–51]. Motivated by the importance of autonomous exploration and tailored to the teamed deployment of legged and flying robots inside subterranean settings, this work contributes two methods, namely on teamed exploration coordination and single-robot planning that enable resilient multi-robot teaming and reliable single-robot operation when communication to and from a robot is not possible, ability to negotiate challenging terrain and capacity to map diverse and large-scale geometries.

III. PROBLEM FORMULATION

The overall problem considered in this work is that of autonomously exploring a bounded volume $V \subset \mathbb{R}^3$ enclosing a subterranean environment with a heterogeneous team of robots given models of their motion constraints, as well as of their onboard depth sensors $\{\mathbb{S}^i\}$ with horizontal and vertical Fields Of View (FoV) $[F_H^i, F_V^i]$, and effective range d_{\max}^i . As subterranean environments such as underground mines, subway infrastructure, and caves often consist of complex networks of branches, multiple levels, large rooms, vertical structures, steep slopes, and anomalous terrain, we concentrate on the teaming of legged and flying systems that

exploit their synergies, while maintaining a high degree of individual autonomy.

The total exploration problem is cast globally and refers to determining which parts of the initially unmapped space $V_{unm} \stackrel{init}{=} V$ are free $V_{free} \subset V$ or occupied $V_{occ} \subset V$. The environment is represented as an occupancy map \mathcal{M} discretizing the volume in (free, occupied or unknown) cubical voxels $m \in \mathcal{M}$ with edge size r_V . Since most depth sensing modalities cannot pass through objects, the environment may contain hollow, narrow, or more generally occluded sections that can not be explored:

Definition 1 (Residual Volume) Let Ξ be the simply connected set of collision free configurations and $\tilde{V}_m \subseteq \Xi$ the set of all configurations from which the voxel m can be perceived by a depth sensor \mathcal{S} . Then the residual volume is given as $V_{res} = \bigcup_{m \in \mathcal{M}} (m | \tilde{V}_m = \emptyset)$.

The multi-robot exploration problem is then defined as:

Problem 1 (Multi-Robot Volumetric Exploration) Given a bounded volume V and a team of N_R heterogeneous robots, find a set of N_R collision free paths σ_i , $i = 1, \dots, N_R$ starting at an initial configuration $\xi_{init} \in \Xi$ that leads to identifying the free and occupied parts V_{free} and V_{occ} , such that there does not exist any collision free configuration from which any piece of $V \setminus \{V_{free}, V_{occ}\}$ could be perceived ($V_{free} \cup V_{occ} = V \setminus V_{res}$). Feasible paths σ_i of this problem are subject to the limited Field of View (FoV) of the sensor, its modeled range, and applicable robot motion constraints.

As noted however, it is in the core principle of this work that multi-robot exploration takes place by robots demonstrating major individual autonomy since subterranean environments present topologies and materials that may prohibit continuous network connectivity. Respectively, we define the single-robot exploration problem given that a volume $V^{S_i^k} \subset V$ is allocated to a robot i at time instance t_k and for a duration T_k^i .

Problem 2 (Single-Robot Volumetric Exploration) Given a bounded volume $V^{S_i^k}$ and a robot i , find a collision free path $\sigma_{S_i^k}$ starting at an initial configuration $\xi_{init,i}^k \in \Xi$ that identifies the free and occupied parts $V_{free}^{S_i^k}$ and $V_{occ}^{S_i^k}$, such that there does not exist any reachable collision free configuration from which any piece of $V^{S_i^k} \setminus \{V_{free}^{S_i^k}, V_{occ}^{S_i^k}\}$ could be perceived. Thus, $V_{free}^{S_i^k} \cup V_{occ}^{S_i^k} = V^{S_i^k} \setminus V_{res}$. A feasible path must respect the sensor model, and applicable motion, namely a max yaw rate and - for legged robots - traversability limitations requiring that \mathcal{M} presents supportive ground along every path and its inclination is affordable.

IV. PROPOSED APPROACH

The presented contribution outlines a strategy for multi-robot ground and flying exploration of predominantly subterranean environments that exploits increased levels of individual robot autonomy based on a comprehensive exploration planner. We detail both the new single-robot Graph-Based exploration Planner (GBPlanner 2.0) and the CoOperation for Heterogeneous Robot Teams (COHORT) planning method.

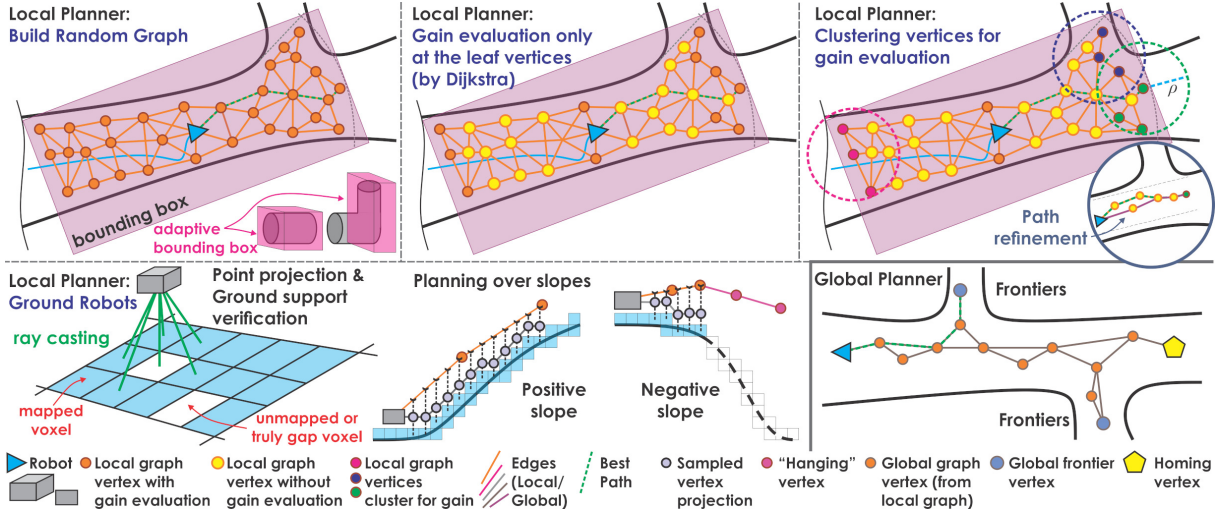


Fig. 2. Outline of the key functional steps of the revised graph-based subterranean exploration planner for aerial and ground (legged) robots.

A. Graph-based Subterranean Exploration Revisited

At the core of the presented policy for autonomous subterranean exploration through ground and aerial robot teaming is a path planner for single-robot autonomous exploration of assigned, initially unmapped, underground volumes $V^{S_i^k}$. The method builds on top of our previous open-source work on graph-based subterranean exploration [21], which we will refer to as “GBPlanner1”. The new GBPlanner version presented in this paper, “GBPlanner2”, has been heavily improved to best handle challenging subterranean geometries and steep slopes, and to provide improved computational performance. GBPlanner2 retains the bifurcated local/global planning architecture originally proposed, with the local planner being responsible for identifying efficient exploration paths that respect the robot motion and perception constraints, while the global planner is triggered to reposition the robot towards a previously perceived frontier of the exploration space when the local layer reports inability to find a path of significant exploration gain or to ensure that the robot returns home within its endurance limits. Figure 2 provides a graphical overview of GBPlanner2. Like GBPlanner1, the new method considers exploration gain, as the new volume to be observed by the robot i if it moves along a certain path $\sigma_{L,best}$ given a depth sensor \mathbb{S}^i with FoV $[F_H^i, F_V^i]$ and modeled range d_{max}^i (Algorithm 1).

As compared to GBPlanner1, most changes are in the local exploration step. For ground robots, the method now explicitly handles the case of (steep) positive or negative slopes, vertical obstacles, and ditches, thus offering true 3D exploration capabilities for ground systems. Exploiting the volumetric map representation based on Voxblox [52], the method randomly samples vertices in the known free space within a local bounding box that is adaptively calculated to best fit the locally explored geometry. To identify the best fitting bounding box D_L , Principal Component Analysis (PCA) is performed on the locally aggregated point cloud and the respective normalized eigenvectors $[v_x, v_y, v_z]$ are scaled by $\mu_V > 0$ (tunable). Accordingly, a random local graph \mathbb{G}_L can be built. For a flying robot, admissible edge connections

between two vertices are simply collision-free straight line segments. Conversely, when planning for a ground robot, the method checks each edge between two vertices by a) projecting points along the edge onto the ground and verifying that ground exists and is mapped, thus being able to support the robot’s traversal, and b) verifying that the path inclination is within feasible bounds for the given robot (Algorithm 2). Notably, the projection also queries points on a circle around the shortest point to the ground as ground voxels may occasionally be incorrectly mapped as “free” (e.g., due to high LiDAR incidence angle, water puddles). This key feature also allows for the organization of vertices in those that have supportive ground, and those that are in free space but do not have any ground (occupied) voxels below them, called “hanging” vertices. They can be connected to other vertices in \mathbb{G}_L through edges adhering to the edge inclination limit and are used for volumetric gain purposes, but their gain is scaled by $e^{-\gamma_H}$, $\gamma_H > 0$ and the edges to them are not allowed to be commanded to the robot.

Algorithm 1 Local Planner

```

1:  $\xi_0 \leftarrow \text{GetCurrentConfiguration}()$ 
2:  $\mathbb{G}_L \leftarrow \text{BuildLocalGraph}(\xi_0)$   $\triangleright$  Algorithm 2
3:  $\Sigma_L \leftarrow \text{GetDijkstraShortestPaths}(\mathbb{G}_L, \xi_0)$ 
4:  $\text{ComputeVolumetricGain}(\mathbb{G}_L)$   $\triangleright$  Algorithm 3
5:  $g_{best} \leftarrow 0$ 
6:  $\sigma_{L,best} \leftarrow \emptyset$ 
7: for all  $\sigma \in \Sigma_L$  do
8:    $g_\sigma \leftarrow \text{ExplorationGain}(\sigma)$ 
9:   if  $g_\sigma > g_{best}$  then
10:     $g_{best} \leftarrow g_\sigma$ ;  $\sigma_{L,best} \leftarrow \sigma$ 
11:  $\sigma_{L,best} \leftarrow \text{ImprovePath}(\sigma_{L,best})$ 
12: return  $\sigma_{L,best}$ 

```

Provided these modifications, the method then builds a local random graph as in GBPlanner1. Once the graph is built, Dijkstra’s algorithm is used to derive shortest paths from the robot’s location to all vertices. Along these paths Σ_L , volumetric exploration gain is calculated over each of

Algorithm 2 Build a Local Graph \mathbb{G}_L ($\xi = [x, y, z, \psi]^T$)

```
1: function BUILDLOCALGRAPH( $\xi_0$ )
2:    $\mathbb{V} \leftarrow \{\xi_0\}; \mathbb{E} \leftarrow \emptyset$   $\triangleright \xi_0[x_0, y_0, z_0, \psi_0]^T$ 
3:    $\mathbb{G}_L = (\mathbb{V}, \mathbb{E})$ 
4:    $D_L \leftarrow \text{CalculateAdaptiveLocalBound}(\xi_0)$ 
5:   while  $N_{\mathbb{V}} \leq N_{\mathbb{V}, \max}$  and  $N_{\mathbb{E}} \leq N_{\mathbb{E}, \max}$  do
6:      $\xi_{rand} \leftarrow \text{SampleFree}(D_L)$ 
7:     if GroundRobot then
8:        $\xi_{proj} \leftarrow \text{ProjectPoint}(\xi_{rand})$ 
9:        $\xi_{proj}.Hanging \leftarrow \text{False}$ 
10:      if not GroundAttached( $\xi_{proj}$ ) then
11:         $\xi_{proj}.Hanging \leftarrow \text{True}$ 
12:         $\xi_{nearest} \leftarrow \text{NearestVertex}(\mathbb{G}_L, \xi_{proj})$ 
13:         $e \leftarrow \text{ProjectEdge}(\xi_{proj}, \xi_{nearest})$ 
14:         $\xi_{new} \leftarrow \xi_{proj}$ 
15:      else
16:         $\xi_{nearest} \leftarrow \text{NearestVertex}(\mathbb{G}_L, \xi_{rand})$ 
17:         $e \leftarrow \text{LineSegment}(\xi_{rand}, \xi_{nearest})$ 
18:         $\xi_{rand}.Hanging \leftarrow \text{False}$ 
19:         $\xi_{new} \leftarrow \xi_{rand}$ 
20:      if AdmissibleEdge( $e$ ) then
21:         $\mathbb{V} \leftarrow \mathbb{V} \cup \{\xi_{new}\}; \mathbb{E} \leftarrow \mathbb{E} \cup \{e\}$ 
22:         $\Xi_{near} \leftarrow \text{NearestVertices}(\mathbb{G}_L, \xi_{new}, \delta)$ 
23:        for all  $\xi_{near} \in \Xi_{near}$  do
24:          if GroundRobot then
25:             $e_{near} \leftarrow \text{ProjectEdge}(\xi_{new}, \xi_{near})$ 
26:          else
27:             $e_{near} \leftarrow \text{LineSegment}(\xi_{new}, \xi_{near})$ 
28:          if AdmissibleEdge( $e_{near}$ ) then
29:             $\mathbb{E} \leftarrow \mathbb{E} \cup \{e_{near}\}$ 
30:  return  $\mathbb{G}_L = (\mathbb{V}, \mathbb{E})$ 
```

Algorithm 3 Compute Volumetric Gain

```
1: function COMPUTEVOLUMETRICGAIN( $\mathbb{G}_L$ )
2:    $\mathbb{V}_l \leftarrow \text{GetLeafVertices}(\mathbb{G}_L)$ 
3:   while  $\mathbb{V}_l \neq \emptyset$  do
4:      $v \leftarrow \mathbb{V}_l[0]$ 
5:      $\mathbb{V}_{near} \leftarrow \text{NearestVertices}(\mathbb{G}_L, v, \rho)$ 
6:      $v.Gain \leftarrow \text{VolumetricGain}(v)$ 
7:     if  $v.Hanging$  then
8:        $v.Gain \leftarrow e^{-\gamma_H} v.Gain$ 
9:     for all  $v_{near} \in \mathbb{V}_{near}$  do
10:       $v_{near}.Gain \leftarrow v.Gain$ 
11:      if  $v_{near}.Hanging$  then
12:         $v_{near}.Gain \leftarrow e^{-\gamma_H} v_{near}.Gain$ 
13:   $\mathbb{V}_{near} \leftarrow \mathbb{V}_{near} \setminus v_{near}$ 
```

their vertices and accumulated to derive the total gain of a path as in [21]. Since this process is among the most computationally demanding within the method, GBPlanner2 offers the option to calculate the gain only on leaf vertices of the Dijkstra paths and also cluster these vertices using a radius $\rho > 0$ thus allowing to approximate the gain of some vertices based on the calculated gain of a nearby vertex.

This modification is tailored to computationally-constrained micro flying robots operating in very wide, long and tall environments as ray casting, used to identify the number of unknown voxels to be observed by a depth sensor from a new vertex location, can be very demanding. Algorithms 1-3 offer an overview of the revised local planning in GBPlanner2. The best path $\sigma_{L, best}$ is derived and conducted by the robot. Beyond utilizing the revised edges from the local layer and other small improvements, the global planning stage of GBPlanner2 is identical to that of GBPlanner1 as in [21].

B. Cooperation for Heterogeneous Robot Teams

COHORT enables teamed operation by heterogeneous robotic systems, each empowered with single-agent exploration autonomy based on GBPlanner2. A prerequisite for this task is that all robots' poses and maps are expressed in the same inertial frame \mathcal{I}_C , an ability facilitated by the global multi-modal, multi-robot mapping (M3RM) and Voxgraph frameworks detailed in [48, 53, 54]. M3RM requires software to run both onboard the robot and on a Centralized Computational Hub (C2H), with the robot iteratively sending selective and compressed local map data and the C2H improving and aligning the map estimates of the robots thus requiring a (possibly intermittently available) communication link. COHORT operates as outlined in Figure 3 in order to support the exploration of the unknown volume V (and map \mathcal{M}) by a team of N_R robots. At any time, every robot i , $i = 1, \dots, N_R$ builds a local "submap" \mathcal{M}_i^l which is the volumetric representation of the environment over a predefined window as developed by Voxblox [52] with resolution r_C . The submap window is defined based both on a fixed time-slot (here 45 s) and a maximum robot displacement from the point of starting a new submap (here 20 m) with a new submap initiated when any of these thresholds is exceeded.

Once submaps are received and aligned with the global map estimate \mathcal{M}^C using Voxgraph [54], they are processed by COHORT. Upon receiving a new submap, wavefront propagation is performed to identify map frontiers. As wavefront we define the list of unvisited voxels neighboring visited ones in this submap. The wavefront propagation utilizes Breadth-First Search (BFS) and results in getting back adjacent voxels forming common frontiers which are defined as a collection of neighboring frontier points (centroids of unknown voxels neighboring a free mapped voxel). Using the extremities of the visited voxels above, a local bounding box $D_{\mathcal{M}_i^l}$ is derived per submap. Within $D_{\mathcal{M}_i^l}$ the method randomly samples a fixed number of vertices expressed in the local frame \mathcal{B}_i^l of that submap which are also checked for lying in free space. Using the sampled set of free points ($[x, y, z]^T$) in the submap, a graph $\mathbb{G}_{\mathcal{M}_i^l}$ (called "subgraph") with N_C vertices and all collision-free edges within a radius $\rho_C > 0$ of each of the sampled points, alongside sampled points over the trajectory of the robot within \mathcal{M}_i^l is constructed. Finally, the built $\mathbb{G}_{\mathcal{M}_i^l}$ is added to the COHORT's global graph of all robots \mathbb{G}_C which in turn grows to expand through all the explored space from all the robots as subgraphs are integrated into it.

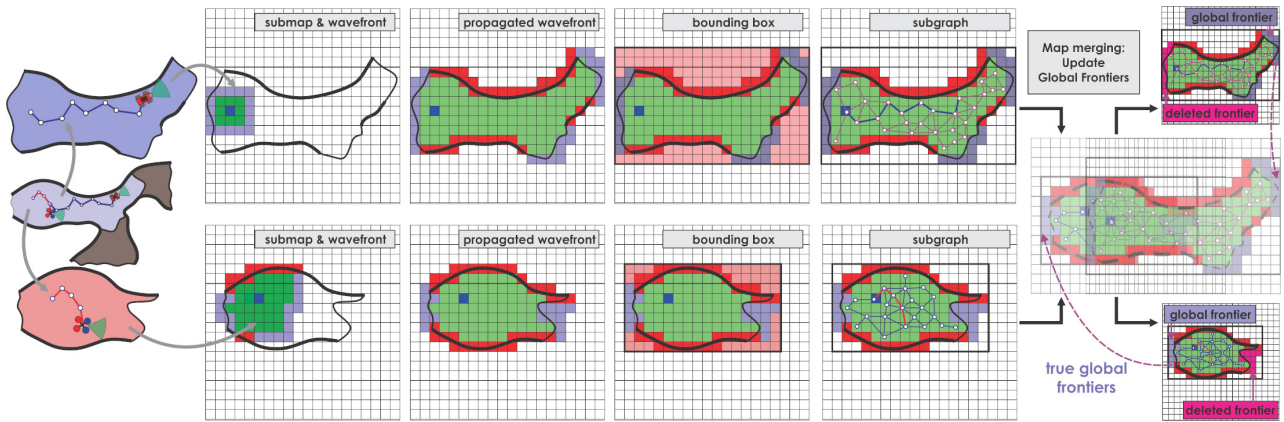


Fig. 3. Outline of how COHORT combines submaps from multiple robots to identify global frontiers and build individual subgraphs, that are merged to form a global graph to allow re-positioning to those frontiers in the global map that are likely to best enable future exploration.

When a new submap is added or the transformation to the submaps changes due to updated multi-robot map alignment, each frontier point is checked in the global map \mathcal{M}^C for occupancy. If a frontier point for a submap turns out to lie in a known voxel for another submap, that frontier point is deleted. The frontier points are then clustered into frontiers by identifying all connected components via BFS with 26-connectivity [55]. Frontiers that are smaller than a set minimum size $f_{\min} > 0$ (here, $f_{\min} = 250$) are discarded and their frontier points are deleted. Once each frontier is re-clustered, then the global graph is checked to identify the points of \mathbb{G}_C that are near the geometric centroid of the points forming the frontier. These points are linked to the frontier and facilitate multi-robot exploration coordination.

Specifically, multi-robot exploration in COHORT takes place by exploiting the above data structure organization and the services of GBPlanner2. In particular, any robot in the team operates by exploring the environment using GBPlanner2 but periodically seeks to update where it explores based on COHORT by re-positioning itself towards a frontier of the map \mathcal{M}^C as explored by the team of robots. In detail, as long as a robot is in communication range to the C2H (possibly via multiple network-hops through other robots or breadcrumbed communication nodes as in [48]), at every T_C seconds of exploration the following steps are taken: First, the distances from all robots to all points on the COHORT’s global graph \mathbb{G}_C and the distances from the home location (ξ_{init}) to all \mathbb{G}_C vertices are calculated using Dijkstra. Then, the frontier points are all expressed into the global frame \mathcal{I}_C and re-clustered such that overlapping frontiers from different submaps are not treated separately and an accurate estimate of the size of each frontier is derived. The re-clustered frontiers are sorted and the best $n_C\%$ are derived. Among the top $n_C\%$ frontiers, the one that is closest to the robot calling for re-positioning is derived and using \mathbb{G}_C the path for the robot to traverse is found, expressed in the robot’s coordinate frame, and commanded to the system. COHORT also provides to the robot a new (fixed-size) bounding box $D^{S_i^k}$ updating the GBPlanner2 exploration goal volume $V^{S_i^k}$ such that it is now around the targeted frontier region. Upon arrival to the

selected frontier, GBPlanner2 is re-triggered using the new $V^{S_i^k}$ ($D^{S_i^k}$) as goal volume to explore. After T_C seconds of GBPlanner2-based exploration, COHORT will be called again to update where the robot will be exploring, while the procedure continues iteratively on this robot and works in an identical fashion across all robots. Notably, when a robot interacts with COHORT it also stores onboard the location it was at during that interaction. If at a subsequent attempt to interact with COHORT the communication channel is not available, the system will check if the whole assigned $V^{S_i^k}$ is explored and if so, it will return to the point that last offered a communication link (or iteratively proceed to previous such points if needed). If $V^{S_i^k}$ is not fully explored yet, the robot continues for an extra T_C^e seconds using GBPlanner2 before it backtracks to a previous communication point.

V. EVALUATION STUDIES

To systematically evaluate the proposed contributions, we present two sets of experimental and simulation results. The first involves the field deployment and exploration of different sections of the Hagerbach underground mine in Switzerland using an ANYmal C quadruped robot [56, 57], and a custom flying robot called “RMF-Obelix” (built as a follow-up of the work in [58]) with both implementing GBPlanner2 onboard. ANYmal C integrated a Velodyne VLP-16 which was used for the purposes of mapping and volumetric gain evaluation. RMF-Obelix is a lightweight (1.4 kg) aerial robot integrating an OUSTER OS0-64 LiDAR. Table I summarizes the GBPlanner2 parameters for ANYmal C and RMF-Obelix, alongside the total robot endurance T_{end} , mission time T_M , average processing time t_p and mapping resolution r_V , while $\mu_V = 50$ m. It is noted that both systems rely on our work in [59] for localization and mapping, but have very diverse processing capabilities, namely an Intel i7 8850H CPU with 6 cores (12 threads) on ANYmal and an Amlogic A311D ARM big.LITTLE (4× A73 @2.2 GHz, 2× A53 @1.8 GHz) CPU for RMF-Obelix, and thus have different GBPlanner2 settings. Figure 4 presents the results of this mission with ANYmal C exploring for 37 min and RMF-Obelix for 8 min, both without any manual interruptions, automatically re-positioning to different sections of the mine and safely returning to home. We also provide detailed

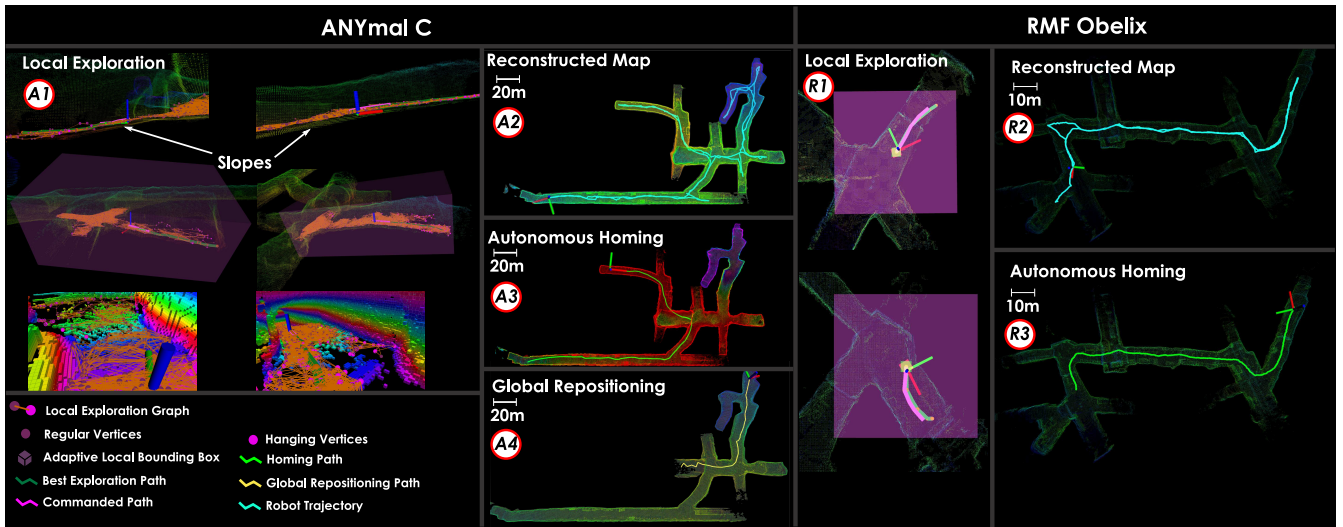


Fig. 4. Autonomous exploration using the ANYmal C legged robot and RMF-Obelix flying system running GBPlanner2 in the Hagerbach underground mine in Switzerland. ANYmal C conducted exploration for 37 minutes demonstrating navigation over slopes, global repositioning and homing, whereas RMF-Obelix performed an 8 minute mission including homing. For ANYmal C’s mission, $\gamma_H = 5$ is used to tune the weight of hanging vertices.

Parameter	ANYmal C	RMF-Obelix
r_V	0.2 m	0.3 m
$N_{V,max}$	1000	500
$[F_H, F_V]$, used d_{max}	$[360, 30]^\circ$, 30 m	$[360, 90]^\circ$, 30 m
ρ	0 (disabled)	1 m
T_{end}	70 min	8.5 min
T_M	37.6 min	8.2 min
t_p	605.53 ms	278.59 ms

TABLE I

GBPLANNER2 PARAMETERS FOR BOTH ROBOTS.

video overview of these missions, available at <https://s.ntnu.no/exploration>.

To evaluate COHORT and its inter-operation with GBPlanner2, we present a large-scale simulation scenario involving 3 flying robot models that our team has openly released in the DARPA Subterranean Challenge repositories [60] (Robot 1: RMF-Obelix [61], Robots 2,3: M100 [62]) but with “virtual endurance” of [2000, 2500, 2500] s respectively. Figure 5 presents the results for teamed exploration and shows the path of each robot, instances where COHORT re-positioned robots to global frontiers of the commonly explored map (with $r_C = 0.4$ m, $N_C = 200$, $\rho_C = 4.0$ m, $n_C\% = 50\%$), and examples of exploration outside of communications range. Every time COHORT commands a robot to re-position itself towards a frontier, GBPlanner2 is engaged ($T_C = 50$ s, $D^{S^k} = 360 \times 360 \times 20$ m) onboard the robot. As communication constraints are also modeled, at certain instances GBPlanner2 continues the exploration for an additional $T_C^e = 150$ s before commanding the robot to backtrack to a node where communications were available.

VI. CONCLUSIONS

This paper presented a complete framework for the autonomous teamed exploration of complex environments such as large-scale, multi-branching subterranean settings involving challenging terrain. The method relies on the harmonic synergy between resilient exploration path planning at the level of each individual robot, combined with a multi-robot

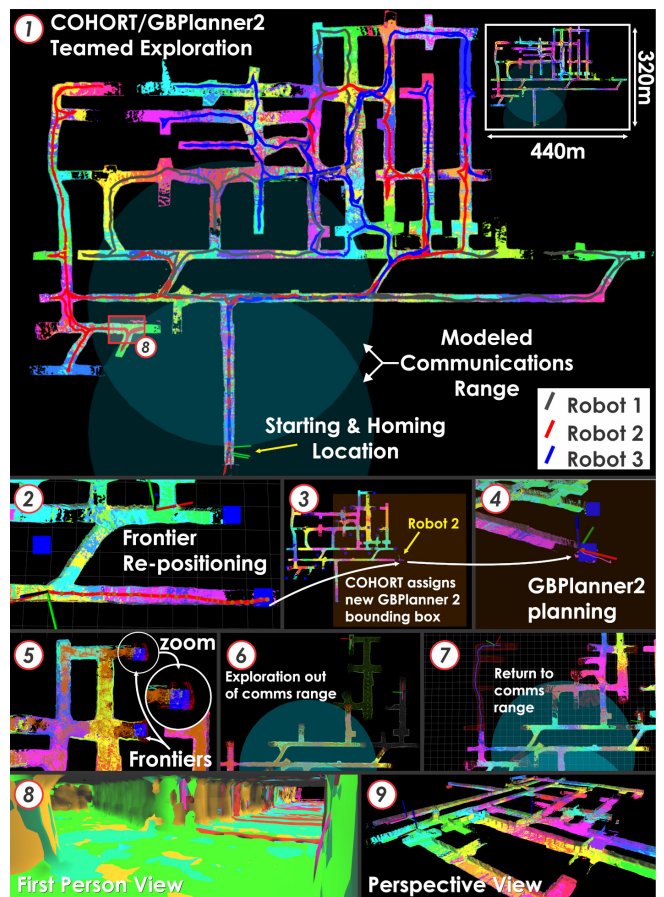


Fig. 5. Instances of teamed exploration with 3 flying robots showing the combined map at the C2H (1), frontiers (5), and path to frontier outside communication range with new bounds calculated by COHORT (2,3) using its global graph. GBPlanner2 explores outside communication range (shown as cyan circles) and returns after T_C^e (4,6,7). Map 3D views in (8,9).

exploration coordination framework that exploits aligned maps to identify the global frontiers of the mapped environment and is robust against communications-deprived environments. To evaluate our contribution, a set of large-scale field deployments and simulations are presented.

REFERENCES

- [1] H. Balta, J. Bedkowski, S. Govindaraj, K. Majek, P. Musialik, D. Serano, K. Alexis, R. Siegwart, and G. De Cubber, "Integrated data management for a fleet of search-and-rescue robots," *Journal of Field Robotics*, vol. 34, no. 3, pp. 539–582, 2017.
- [2] T. Tomic, K. Schmid, P. Lutz, A. Domel, M. Kassecker, E. Mair, I. L. Grix, F. Ruess, M. Suppa, and D. Burschka, "Toward a fully autonomous uav: Research platform for indoor and outdoor urban search and rescue," *IEEE robotics & automation magazine*, vol. 19, no. 3, pp. 46–56, 2012.
- [3] N. Michael, S. Shen, K. Mohta, Y. Mulgaonkar, V. Kumar, K. Nagatani, Y. Okada, S. Kiribayashi, K. Otake, K. Yoshida *et al.*, "Collaborative mapping of an earthquake-damaged building via ground and aerial robots," *Journal of Field Robotics*, vol. 29, no. 5, pp. 832–841, 2012.
- [4] 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, oct 2019. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.21887>
- [5] A. Bircher, M. Kamel, K. Alexis, M. Burri, P. Oettershagen, S. Omari, T. Mantel and R. Siegwart, "Three-dimensional coverage path planning via viewpoint resampling and tour optimization for aerial robots," *Autonomous Robots*, pp. 1–25, 2015.
- [6] A. Bircher, K. Alexis, M. Burri, P. Oettershagen, S. Omari, T. Mantel and R. Siegwart, "Structural inspection path planning via iterative viewpoint resampling with application to aerial robotics," in *IEEE International Conference on Robotics and Automation (ICRA)*, May 2015, pp. 6423–6430. [Online]. Available: <https://github.com/ethz-asl/StructuralInspectionPlanner>
- [7] G. Caprari, A. Breitenmoser, W. Fischer, C. Hürzeler, F. Tâche, R. Siegwart, O. Nguyen, R. Moser, P. Schoeneich, and F. Mondada, "Highly compact robots for inspection of power plants," *Journal of Field Robotics*, vol. 29, no. 1, pp. 47–68, 2012.
- [8] C. Balaguer, A. Giménez, J. M. Pastor, V. Padron, and M. Abderrahim, "A climbing autonomous robot for inspection applications in 3d complex environments," *Robotica*, vol. 18, no. 3, pp. 287–297, 2000.
- [9] J. Sawada, K. Kusumoto, Y. Maikawa, T. Munakata, and Y. Ishikawa, "A mobile robot for inspection of power transmission lines," *IEEE Transactions on Power Delivery*, vol. 6, no. 1, pp. 309–315, 1991.
- [10] C. Gehring, P. Fankhauser, L. Isler, R. Diethelm, S. Bachmann, M. Potz, L. Gerstenberg, and M. Hutter, "Anymal in the field: Solving industrial inspection of an offshore hvdc platform with a quadrupedal robot," in *12th Conference on Field and Service Robotics (FSR 2019)*, 2019.
- [11] M. Hutter, R. Diethelm, S. Bachmann, P. Fankhauser, C. Gehring, V. Tsounis, A. Lauber, F. Guenther, M. Bjelonic, L. Isler *et al.*, "Towards a generic solution for inspection of industrial sites," in *Field and Service Robotics*. Springer, 2018, pp. 575–589.
- [12] H. Kolvenbach, G. Valsecchi, R. Grandia, A. Ruiz, F. Jenelten, and M. Hutter, "Tactile Inspection of Concrete Deterioration in Sewers with Legged Robots," in *Field and Service Robots (FSR)*, Tokyo, Japan, 2019, p. 14. [Online]. Available: <https://youtu.be/fdGKRgVYAtg>
- [13] B. Grocholsky, J. Keller, V. Kumar, and G. Pappas, "Cooperative air and ground surveillance," *IEEE Robotics & Automation Magazine*, vol. 13, no. 3, pp. 16–25, 2006.
- [14] K. Schilling and C. Jungius, "Mobile robots for planetary exploration," *Control Engineering Practice*, vol. 4, no. 4, pp. 513–524, 1996.
- [15] J. Bares, M. Hebert, T. Kanade, E. Krotkov, T. Mitchell, R. Simmons, and W. Whittaker, "Ambler: An autonomous rover for planetary exploration," *Computer*, vol. 22, no. 6, pp. 18–26, 1989.
- [16] M. J. Schuster, S. G. Brunner, K. Bussmann, S. Büttner, A. Dömel, M. Hellerer, H. Lehner, P. Lehner, O. Porges, J. Reill *et al.*, "Towards autonomous planetary exploration," *Journal of Intelligent & Robotic Systems*, vol. 93, no. 3, pp. 461–494, 2019.
- [17] R. Welch, D. Limonadi, and R. Manning, "Systems engineering the curiosity rover: A retrospective," in *2013 8th international conference on system of systems engineering*. IEEE, 2013, pp. 70–75.
- [18] J. Balaram, M. Aung, and M. P. Golombek, "The ingenuity helicopter on the perseverance rover," *Space Science Reviews*, vol. 217, no. 4, pp. 1–11, 2021.
- [19] P. A. Boeder and C. E. Soares, "Mars 2020: mission, science objectives and build," in *Systems Contamination: Prediction, Control, and Performance 2020*, vol. 11489. International Society for Optics and Photonics, 2020, p. 1148903.
- [20] Defense Advanced Research Projects Agency (DARPA), "DARPA Subterranean Challenge." [Online]. Available: <https://subtchallenge.com/>
- [21] T. Dang, M. Tranzatto, S. Khattak, F. Mascarich, K. Alexis, and M. Hutter, "Graph-based subterranean exploration path planning using aerial and legged robots," *Journal of Field Robotics*, vol. 37, no. 8, pp. 1363–1388, 2020.
- [22] C. Papachristos, S. Khattak, and K. Alexis, "Uncertainty-aware receding horizon exploration and mapping using aerial robots," in *IEEE International Conference on Robotics and Automation (ICRA)*, May 2017.
- [23] A. Bircher, M. Kamel, K. Alexis, H. Oleynikova and R. Siegwart, "Receding horizon "next-best-view" planner for 3d exploration," in *IEEE International Conference on Robotics and Automation (ICRA)*, May 2016. [Online]. Available: <https://github.com/ethz-asl/nbvplanner>
- [24] L. Yoder and S. Scherer, "Autonomous exploration for infrastructure modeling with a micro aerial vehicle," in *Field and Service Robotics*. Springer, 2016, pp. 427–440.
- [25] C. Connolly *et al.*, "The determination of next best views," in *IEEE International Conference on Robotics and Automation 1985*. IEEE, 1985.
- [26] B. Yamauchi, "A frontier-based approach for autonomous exploration," in *CIRA'97*. IEEE, 1997, pp. 146–151.
- [27] M. R. Dharmadhikari, T. Dang, L. Solanka, J. B. Loje, D. H. Nguyen, N. Khedekar, and K. Alexis, "Motion primitives-based path planning for fast and agile exploration using aerial robots," in *IEEE International Conference on Robotics and Automation (ICRA)*, May 2019.
- [28] I. D. Miller, F. Cladera, A. Cowley, S. S. Shivakumar, E. S. Lee, L. Jarin-Lipschitz, A. Bhat, N. Rodrigues, A. Zhou, A. Cohen *et al.*, "Mine tunnel exploration using multiple quadrupedal robots," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2840–2847, 2020.
- [29] Y. Pei, M. W. Mutka, and N. Xi, "Coordinated multi-robot real-time exploration with connectivity and bandwidth awareness," in *2010 IEEE international conference on robotics and automation*. IEEE, 2010, pp. 5460–5465.
- [30] W. Burgard, M. Moors, C. Stachniss, and F. E. Schneider, "Coordinated multi-robot exploration," *IEEE Transactions on robotics*, vol. 21, no. 3, pp. 376–386, 2005.
- [31] K. Cesare, R. Skeeel, S.-H. Yoo, Y. Zhang, and G. Hollinger, "Multi-uav exploration with limited communication and battery," in *2015 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2015, pp. 2230–2235.
- [32] M. Corah and N. Michael, "Volumetric objectives for multi-robot exploration of three-dimensional environments," *arXiv preprint arXiv:2103.11625*, 2021.
- [33] K. Goel, M. Corah, C. Boirum, and N. Michael, "Fast exploration using multirotors: Analysis, planning, and experimentation," in *Field and Service Robotics*. Springer, 2021, pp. 291–305.
- [34] K. Goel, W. Tabib, and N. Michael, "Rapid and high-fidelity subsurface exploration with multiple aerial robots," *arXiv preprint arXiv:2012.10788*, 2020.
- [35] A. J. Smith and G. A. Hollinger, "Distributed inference-based multi-robot exploration," *Autonomous Robots*, vol. 42, no. 8, pp. 1651–1668, 2018.
- [36] T. Dang, C. Papachristos, and K. Alexis, "Visual saliency-aware receding horizon autonomous exploration with application to aerial robotics," in *IEEE International Conference on Robotics and Automation (ICRA)*, May 2018.
- [37] M. R. Dharmadhikari, H. Deshpande, T. Dang, and K. Alexis, "Hypergame-based adaptive behavior path planning for combined exploration and visual search," in *IEEE International Conference on Robotics and Automation (ICRA)*, May 2021.
- [38] F. Yang, D.-H. Lee, J. Keller, and S. Scherer, "Graph-based topological exploration planning in large-scale 3d environments," *arXiv preprint arXiv:2103.16829*, 2021.
- [39] E. M. Lee, J. Choi, H. Lim, and H. Myung, "Real: Rapid exploration with active loop-closing toward large-scale 3d mapping using uavs," *arXiv preprint arXiv:2108.02590*, 2021.
- [40] W. Tabib, M. Corah, N. Michael, and R. Whittaker, "Computationally efficient information-theoretic exploration of pits and caves," in *2016*

- IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 3722–3727.
- [41] R. Reinhart, T. Dang, E. Hand, C. Papachristos, and K. Alexis, “Learning-based path planning for autonomous exploration of subterranean environments,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 1215–1221.
- [42] L. Schmid, V. Reijgwart, L. Ott, J. Nieto, R. Siegwart, and C. Cadena, “A unified approach for autonomous volumetric exploration of large scale environments under severe odometry drift,” *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 4504–4511, 2021.
- [43] J. Delmerico, E. Mueggler, J. Nitsch, and D. Scaramuzza, “Active autonomous aerial exploration for ground robot path planning,” *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 664–671, 2017.
- [44] T. Andre and C. Bettstetter, “Collaboration in multi-robot exploration: to meet or not to meet?” *Journal of Intelligent & Robotic Systems*, vol. 82, no. 2, pp. 325–337, 2016.
- [45] C. Cao, H. Zhu, H. Choset, and J. Zhang, “Tare: A hierarchical framework for efficiently exploring complex 3d environments.”
- [46] S. S. Mansouri, C. Kanellakis, P. Karvelis, D. Kominiak, and G. Nikolakopoulos, “Mav navigation in unknown dark underground mines using deep learning,” in *2020 European Control Conference (ECC)*. IEEE, 2020, pp. 1943–1948.
- [47] P. Petracek, V. Kratky, M. Petrlik, T. Baca, R. Kratochvil, and M. Saska, “Large-scale exploration of cave environments by unmanned aerial vehicles,” *IEEE Robotics and Automation Letters*, 2021.
- [48] M. Tranzatto, F. Mascarich, L. Bernreiter, C. Godinho, M. Camurri, S. M. K. Khattak, T. Dang, V. Reijgwart, J. Loeje, D. Wisth *et al.*, “Cerberus: Autonomous legged and aerial robotic exploration in the tunnel and urban circuits of the darpa subterranean challenge,” *Field Robotics*, 2021.
- [49] A. Agha, K. Otsu, B. Morrell, D. D. Fan, R. Thakker, A. Santamaria-Navarro, S.-K. Kim, A. Bouman, X. Lei, J. Edlund *et al.*, “Nebula: Quest for robotic autonomy in challenging environments; team costar at the darpa subterranean challenge,” *arXiv preprint arXiv:2103.11470*, 2021.
- [50] T. Rouček, M. Pecka, P. Čížek, T. Petříček, J. Bayer, V. Šalanský, D. Heřt, M. Petrlik, T. Báča, V. Spurný *et al.*, “Darpa subterranean challenge: Multi-robotic exploration of underground environments,” in *International Conference on Modelling and Simulation for Autonomous Systems*. Springer, Cham, 2019, pp. 274–290.
- [51] S. Scherer, V. Agrawal, G. Best, C. Cao, K. Cujic, R. Darnley, R. DeBortoli, E. Dexheimer, B. Drozd, R. Garg, I. Higgins, J. Keller, D. Kohanbash, L. Nogueira, R. Pradhan, M. Tatum, V. K. Viswanathan, S. Willits, S. Zhao, H. Zhu, D. Abad, T. Angert, G. Armstrong, R. Boirum, A. Dongare, M. Dworman, S. Hu, J. Jaekel, R. Ji, A. Lai, Y. Hsuan Lee, A. Luong, J. Mangelson, J. Maier, J. Picard, K. Pluckter, A. Saba, M. Saroya, E. Scheide, N. Shoemaker-Trejo, J. Spisak, J. Teza, F. Yang, A. Wilson, H. Zhang, H. Choset, M. Kaess, A. Rowe, S. Singh, J. Zhang, G. A. Hollinger, and M. Travers, “Resilient and modular subterranean exploration with a team of roving and flying robots,” *Field Robotics*, 2021.
- [52] 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)*, 2017.
- [53] T. Schneider, M. Dymczyk, M. Fehr, K. Egger, S. Lynen, I. Gilitschenski, and R. Siegwart, “maplab: An open framework for research in visual-inertial mapping and localization,” *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1418–1425, 2018.
- [54] V. Reijgwart, A. Millane, H. Oleynikova, R. Siegwart, C. Cadena, and J. Nieto, “Voxgraph: Globally consistent, volumetric mapping using signed distance function submaps,” *IEEE Robotics and Automation Letters*, vol. 5, no. 1, pp. 227–234, 2019.
- [55] A. Rosenfeld and J. L. Pfaltz, “Sequential operations in digital picture processing,” *Journal of the ACM (JACM)*, vol. 13, no. 4, pp. 471–494, 1966.
- [56] ANYbotics, “ANYmal Legged Robots,” available at <https://www.anybotics.com/>.
- [57] T. Miki, J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, “Learning robust perceptive locomotion for quadrupedal robots in the wild,” *Science Robotics*, vol. 7, no. 62, jan 2022. [Online]. Available: <https://www.science.org/doi/10.1126/scirobotics.abk2822>
- [58] P. De Petris, H. Nguyen, M. Kulkarni, F. Mascarich, and K. Alexis, “Resilient collision-tolerant navigation in confined environments,” in *IEEE International Conference on Robotics and Automation (ICRA)*, May 2021.
- [59] S. Khattak, H. Nguyen, F. Mascarich, T. Dang, and K. Alexis, “Complementary multi-modal sensor fusion for resilient robot pose estimation in subterranean environments,” in *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2020, pp. 1024–1029.
- [60] OpenRobotics. Subt tech repo. Open Robotics. [Online]. Available: <https://subtchallenge.world/openrobotics/fuel/collections/SubT%20Tech%20Repo>
- [61] M. Kulkarni and D. P. Paolo. (September) Cerberus_rmf_sensor_config_1. Autonomous Robots Lab. [Online]. Available: https://github.com/ntnu-arl/subt/tree/submitted_models/cerberus_rmf_obelix_sensor_config_1/submitted_models/cerberus_rmf_obelix_sensor_config_1
- [62] OpenRobotics. (June) Cerberus m100 sensor config 1. Open Robotics. [Online]. Available: https://fuel.ignitionrobotics.org/1.0/OpenRobotics/models/CERBERUS_M100_SENSOR_CONFIG_1