

GVGExp: Communication-Constrained Multi-Robot Exploration System based on Generalized Voronoi Graphs

Kizito Masaba and Alberto Quattrini Li

Abstract—This paper presents GVGExp, a recurrent-connectivity exploration strategy for multi-robot systems to discover unknown environments under communication-constrained conditions. A robust multi-robot exploration strategy with communication constraints is important to accomplish several applications, e.g., underwater or planetary exploration. Mainstream multi-robot exploration strategies have considered unlimited communication. In addition, these strategies delegate the resolution of potential path collisions to local planners. In this paper, we explicitly focus on minimizing the number of communication events between robots given the limited bandwidth that can be available in real missions, as well as explicitly minimizing potential path interference between robots. GVGExp incrementally builds a Generalized Voronoi Graph (GVG), which is used by the robots to determine the topology of the environment. We introduce a novel property of the GVG called *gate* to identify a region (subtree) that is uniquely assigned to each robot, with no overlap, thus minimizing potential path interference. Whenever a robot finishes exploring a region, in a depth-first search fashion, or finds a loop connecting to other parts of the environment, the robot shares information with other robots that are in range to enable coordination. We performed numerous simulations to evaluate our proposed strategy and compared it with other state-of-the-art methods. Experimental results show that GVGExp is able to explore the environments in a relatively short amount of time, significantly reducing communication events and path interference.

I. INTRODUCTION

This paper tackles the problem where multiple robots are simultaneously exploring an unknown environment, but are constrained by limited communication resources and inhibited by potential path interference.

Important applications, such as underwater and planetary exploration [1], require mappings of physical features within unknown environments. The literature has shown multi-robot systems performing well with proper coordination in exploration tasks, because of execution parallelism [2]. However, a typical assumption for such robots is that they always share information with each other over high bandwidth communication. Recently, there has been work to include more realistic communication constraints [3] – such as ensuring connectivity between robots and information sharing with a base station, as part of the exploration system. Yet, these robots typically share all information when in communication range. In practical scenarios where a robust network infrastructure is unavailable, as in a disaster-stricken environment or in the wild, broadcasting all information at all

The authors are with Department of Computer Science, Dartmouth College, Hanover, NH USA {kizito.masaba.gr, alberto.quattrini.li}@dartmouth.edu

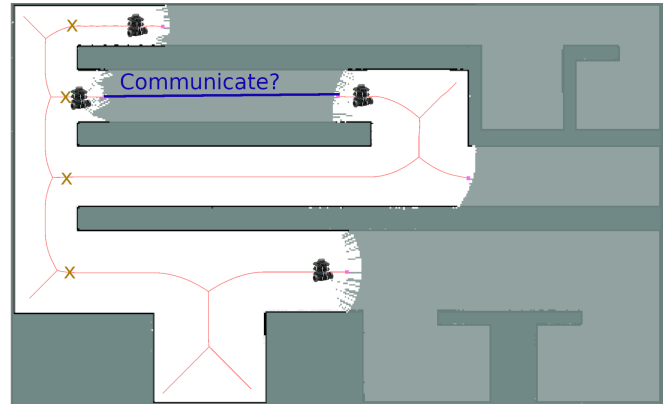


Fig. 1. A scenario highlighting the problem addressed by *GVGExp*: robots explore an unknown environment, guided by the Generalized Voronoi Graph (GVG) (red lines). The GVG captures the topology of the environment, which is used for allocating each robot to separate “branches”. Each allocation identifies a “gate”, i.e., an area assigned to one robot (yellow X’s). In addition, robots can communicate and coordinate when necessary; for instance, two robots exploring the same corridor might need to communicate with each other, so that they do not interfere with each other’s tasks.

times is not possible because of limits on bandwidth (e.g., in the case of a 900 MHz radio up to 2 Mbps [4] or of acoustic communication up to tens of kbps [5]). In addition, exploration strategies proposed over the years assigned robots based on information gain and distance [6]–[8], without necessarily accounting for the potential path interference between robots.

This paper proposes a multi-robot exploration strategy that aims to minimize the number of communication events, potential path interference, and area exploration overlap. To achieve these goals, the proposed strategy uses the topological structure of the environment from a Generalized Voronoi Graph (GVG), built on the partially known environment, to let robots decide where to explore and when to communicate. Our key insight is that most environments are structured in a way that naturally guides a robot during exploration. For example, typical corridors will constrain the robot exploration in at most two directions. Hence, communication is necessary only when robots are likely to conflict with each other (i.e., exploring similar areas). For instance, consider an environment with a number of corridors, which each robot can follow. At intersections, there is a chance for one robot to encounter another. At this moment, data sharing may be necessary and thus should be activated. In addition, only one robot should be assigned per corridor, so that they do not interfere with each other’s motion. Figure 1 shows an illustration of such a scenario.

Furthermore, while using topological graphs for exploring unknown environments has been studied [9], [10], common graphs across all robots may lead to robots exploring similar regions and result in collisions, increasing the exploration time, as reported in [10]. In our work, we address this challenge by introducing topological constraints that foster coordination of robots and minimize conflicts during exploration. The initial information obtained by the robots during deployment provides insights into the topology of the rest of the environment, such as directions in which the robots can explore the unknown area. Using the GVG, we encode this information into a novel GVG property that we call *gate*. It identifies a single direction in which the environment can be explored. Thus, it is analogous to an exit of a given area. Using the *gate* property, we define constraints that enable the deployment of robots into unique target regions, thereby minimizing spatial conflicts during exploration.

Overall, the paper provides the following contributions:

- 1) a novel distributed multi-robot exploration strategy called *GVGExp* that enables task allocation during exploration of unknown environments, where robots share similar topological graphs;
- 2) a recurrent communication strategy that exploits the topological structure provided by the GVG to enable communication and coordination at the appropriate time and place;
- 3) an implementation in ROS¹ and an experimental analysis comparing other methods that share information continuously or at fixed intervals.

The rest of the paper is structured as follows: the next section discusses related multi-robot exploration work. Section III formulates the exploration problem and Section IV describes the proposed exploration and communication strategies. Section V reports experimental results in ROS and a realistic 2D simulator. Finally, Section VI concludes the paper and suggests future work.

II. RELATED WORK

Exploration strategies and coordination methods have been developed over the years to enable multi-robot exploration [6], [11]–[14]. For related recent surveys, please look at, e.g., [15], [16]. Here we discuss multi-robot exploration work that includes communication constraints explicitly, as well as work that exploits topology or semantic information.

Literature includes different connectivity strategies between robots and, in some cases, a base station. Some of them enforce *continuous connection* [17]–[21], where the methods enforce the robots to be connected at all times. For example, Pei *et al.* [22] proposed an algorithm that takes into account bandwidth constraints for determining the relay chain, under a “disk” communication model, where robots can communicate within a given range. We do not require continuous connection, as robots can explore the environment more efficiently that way [23].

¹The code is opensource at <https://github.com/dartmouthrobotics/gvgexploration>.

Other strategies involve recurrent connectivity where robots disconnect from time to time and reconnect at specific instances. Recurrent connectivity work can be classified into two major categories, based on where the connection is established: *homing* and *rendezvous-based* connectivity.

The *homing* connectivity method requires robots to share data with the base station in specific scenarios. Spirin *et al.* [8] designed a method where robots explored and shared information with the base station when the value of the new information compared to the base station’s was over a certain threshold. Banfi *et al.* [7] designed asynchronous coordination strategies that allowed robots to form sub-teams to explore different parts of the environment, while enforcing recurrent connectivity with a base station when new information was acquired. Communication models were recently integrated explicitly in the exploration strategy [24], so that robots can be deployed while satisfying connectivity constraints.

A *rendezvous-based* approach establishes connections at predetermined locations in the environment. Okumura *et al.* [25] proposed a recurrent connectivity approach: at regular intervals during exploration, robots within a given communication range would continuously agree upon new rendezvous locations (called *anchor* points) where they would meet and share data. Amigoni *et al.* [26] proposed a method that switches between guaranteeing multihop connectivity to the base station and rendezvous. Empirically, it was shown that switching communication modalities balances exploration time, traveled distance, and disconnected time to the base station. Hollinger and Singh [23] proposed a method that made robots reconnect at fixed time intervals in the context of search.

While not enforcing a connected topology as for continuous connectivity, the above works constraining to recurrent connectivity assumed continuous sharing of data when in range and focused on maintaining a certain network topology and connectivity constraints. Our proposed method instead focuses on sharing information only when “needed”.

Some exploration methods reasoned on a graph representing the environment. GVGs construction of unknown environments was proposed for single robot exploration [9], [27]. Map segmentation was used to identify and assign regions to robots [28]. Exploration of unknown graphs was also theoretically studied; algorithms, with some theoretical bounds on the exploration time for two robots exploring a tree [29] were provided. Semantic information – i.e., spatial concepts given by humans, including ‘corridors’ and ‘rooms’ – was exploited in exploration strategies. A semantic topological-oriented map was used as a spatial model [30], where the exploration strategy exploited such a topological map to choose the next location to visit, for example to explore corridors first. Semantic information was also utilized to determine the number of robots to assign to a location (e.g., a room labeled as big would have more robots allocated than a room labeled as small) [14]. Our proposed method will use topological information to provide new information for exploration and communication decisions.

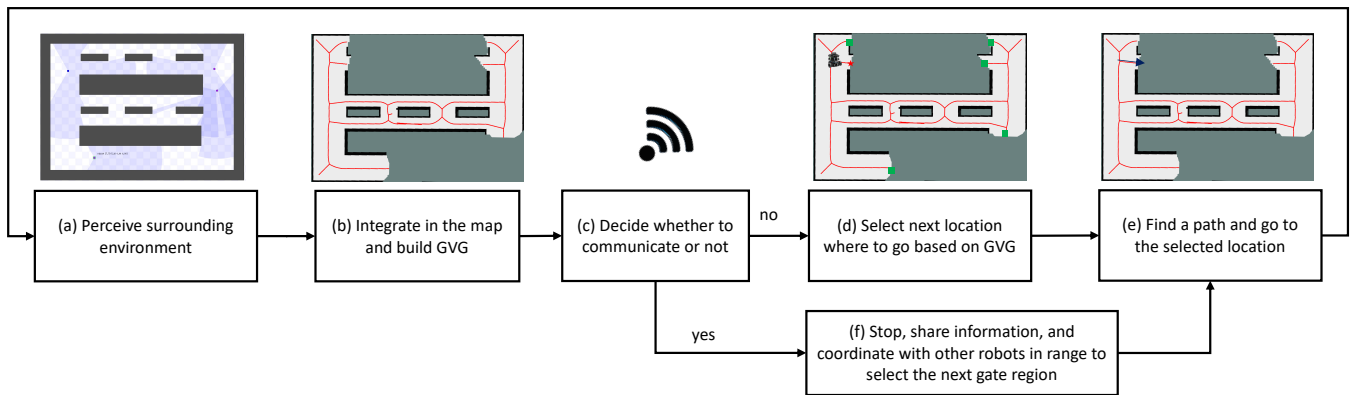


Fig. 2. Main steps of the multi-robot exploration system for each robot to explore an environment, with the communication strategy, which will pause the exploration to initiate data sharing and coordination between available robots.

III. PROBLEM STATEMENT

A set $Z = \{z_1, z_2, \dots, z_m\}$ of m mobile robots of radius, b is deployed in a 2D bounded environment, $\mathcal{W} \subset \mathbb{R}^2$ that is initially unknown, where points are obstacles \mathcal{W}_o or free space \mathcal{W}_f . Each robot is equipped with a finite-range laser sensor of scan radius r_l , field of view θ_{sc} , angle resolution θ_s , and a communication device with communication range r_c .

Each robot runs a graph-based Simultaneous Localization and Mapping (SLAM) algorithm [31] to localize the robots and merge perceived data in an occupancy grid. Each robot has global/local planners to enable their autonomy.

The exploration mission evolves over time $t \in \{0, \dots, T\}$ and the map of the environment is updated with local information acquired by each robot at every time step. Each robot can incorporate information from other robots into its map, when the information is shared.

The goal for each robot is to select the next informative location in the environment to explore, while coordinating and exchanging information, whenever necessary, so that the mission time and coordination (and hence communication) events are minimized.

IV. PROPOSED METHOD

Figure 2 gives the overview of the exploration system from the perspective of a single robot. In the following subsections, we describe each main step starting with the construction of the GVG and discussing then the exploration, coordination, and communication strategy.

A. GVG Generation

A GVG [9] is a locus of points in the navigable region \mathcal{W}_f , which are equidistant to at least two distinct obstacle points in \mathcal{W}_o . Any two neighboring points in the navigable region that are equidistant to two distinct obstacle points form an *edge* on the GVG. Freespace points that are equidistant to three or more obstacle points are called *meetpoints*, and the endpoints of the GVG are called *leaves*. Together, meetpoints and leaves form the nodes of the graph, which are connected by the edges in the GVG. Leaves that are close to the boundary between explored and unexplored regions are candidates

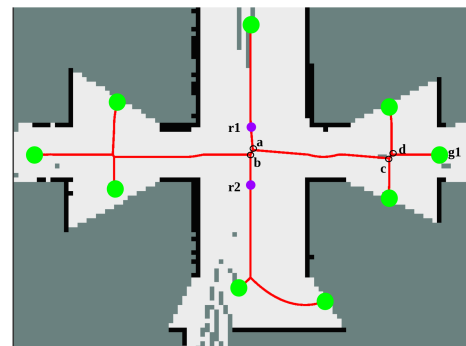


Fig. 3. GVG generated on a partial map for two robots (located at $r1$ and $r2$) in an indoor environment, with the GVG properties highlighted: *frontiers* are the green circles, *root nodes* are the violet circles. There are intermediate nodes (GVG meetpoints) a through d . When a robot is assigned to a frontier, then that becomes a *gate* (e.g., $g1$).

for exploration and are identified as *frontiers* F . Figure 3 shows the GVG generated on the partial map discovered by two robots: the red lines are edges, the intersections between red lines form the meetpoints, and the green dots are frontier leaves.

The data from the robot's LiDAR sensor (Figure 2-step (a)) is integrated into an occupancy grid by the SLAM module. *GVGExp* generates the GVG from that occupancy grid (Figure 2-step (b)). Note, the occupancy grid used to generate the GVG is down-scaled for real-time computation. In our experiments, if the cell resolution of the map was 0.05 m, then the GVG computation time was in the order of a few seconds (between 0.1s and 1.1s); while 0.4m cell resolution took to a few milliseconds (between 10ms to 60ms) of computation time. The down-scale factor is determined such that the cell size is two times the size b of the robot's base. This allows the robot to safely maneuver any passage in the environment. Moreover, this choice does not affect the quality of the GVG, as it captures the topology of the environment. In general, the uncertainty inherent in SLAM introduces noise in the occupancy grid, resulting in a noisy GVG and consequently affecting the exploration, coordination, and communication strategies. We address such

problems by keeping (1) edges and meetpoints that are only in the known navigable free space; (2) leaves or meetpoints that are less than b from their corresponding obstacle points are discarded, as robots cannot reach such locations.

We now introduce the concept of *gate*. A *gate* is a frontier on the GVG to which a robot is assigned when coordinating with other robots – see illustration in Figure 3. We use this concept to derive the robot deployment constraints. The intuition behind is that the topological subgraph rooted at the *gate* and discovered after the robot’s allocation and exploration should not be visited by other robots to avoid any physical interference. The area identified by that subgraph is called a *gate region* and is allocated to one robot. For many realistic environments assigning one region to a single robot is enough. We will investigate in future work how to determine the number of robots needed to explore a large gate region. The *gate* concept will be used as a basis for the proposed exploration, coordination, and communication strategies described in the following subsections.

B. GVG-based exploration strategy

Each robot, when operating independently, explores the environment using the GVG in a depth-first search manner (Figure 2-step (d)). In particular, the robot will choose an available leaf from the set of frontiers F that does not fall in any other robot’s *gate region*. The robot will use the path planner to find the path to the frontier (Figure 2-step (e)). The strategy keeps track of the visited nodes of the GVG so that the depth-first exploration can backtrack.

If the robot is at a GVG meetpoint, the GVG-based exploration strategy chooses the next frontier, f^* according to the information gain, which is measured as the largest unexplored region that would be scanned by the robot’s laser sensor at the candidate location:

$$f^* = \operatorname{argmax}_{f \in F} \left\{ \frac{r_l^2 \theta_{sc}}{2} - s^2 C_f \right\} \quad (1)$$

where C_f is the number of already explored cells at a frontier, f that lies within the scanned area of radius, r_l , θ_{sc} is the laser sensor’s field of view, and s is the resolution of the global occupancy grid map. This greedy exploration strategy has been shown to perform well in different environments [9], [11]. When this evaluation function is paired with the depth-first exploration, it biases the robots towards larger areas, without traversing known regions, unless it reaches a dead end or it closes a loop in the environment.

C. Communication

During GVG exploration, as presented in the previous subsection, the robot does not necessarily need to share information with other robots, because of the *gate* property. This communication strategy is different from other multi-robot exploration systems that continuously broadcast information. Our system minimizes the number of communication events by enabling communication only when it determines that it is necessary for the robots to coordinate (Figure 2-step (c)). The principle to identify such communication events during

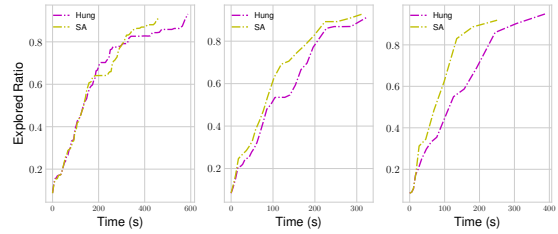


Fig. 4. Comparison of Sequential Allocation (SA) and Hungarian Algorithm (Hung) for deployment of 2 (left), 4 (center) and 6 (right) robots.

the exploration, is when there is an event that would result the robots to interfere with each other:

- 1) at the beginning, when the robots are deployed in the unknown environment;
- 2) a robot completes the exploration of the *gate* to which it is assigned. This is necessary, because it needs to determine the next frontier to explore and the corresponding *gate*;
- 3) the *gate region* connects to previously known part of the environment which might lead to overlapping with other robots’ paths; and
- 4) a robot is at a leaf that can be connected to another leaf located across an unknown space, such that their corresponding edges are collinear. The unknown space is determined by a least-square regression line. The insight is that if two explored locations are disconnected (according to the GVG) but are likely to be part of the same area (e.g., a corridor), it is worth communicating in order to exchange information on the unknown space and potentially end the exploration of that area sooner.

If at least one of the above conditions is met, the robot initiates a connection with other robots that are within the communication range and exchange their data collected so far and their current *gates*.

D. GVG-based robot coordination

Once the communication event is initiated, there are two main types of coordination that can occur (Figure 2-step (f)).

For the first two communication events, a new task allocation process is started by the initiator of the communication. To minimize overlap in an area explored, each robot should be deployed to a unique *gate*. However, some *gates* on the GVG are too close to each other whereas others are further apart. This may be attributed to either narrow spaces in the environment, or uncertainty in the occupancy grid, which may yield leaves that do not represent the topology of the environment.

GVGExp first identifies sparse *gates*, by clustering them using the DBSCAN clustering algorithm [32]. DBSCAN takes an array of points and maximum distance ϵ that determines neighborhood as inputs and returns the clusters in which each point lies. In our system, $\epsilon = \frac{r_l}{2}$ to ensure separation by the sensor radius. This value was empirically chosen based on the clustering accuracy.

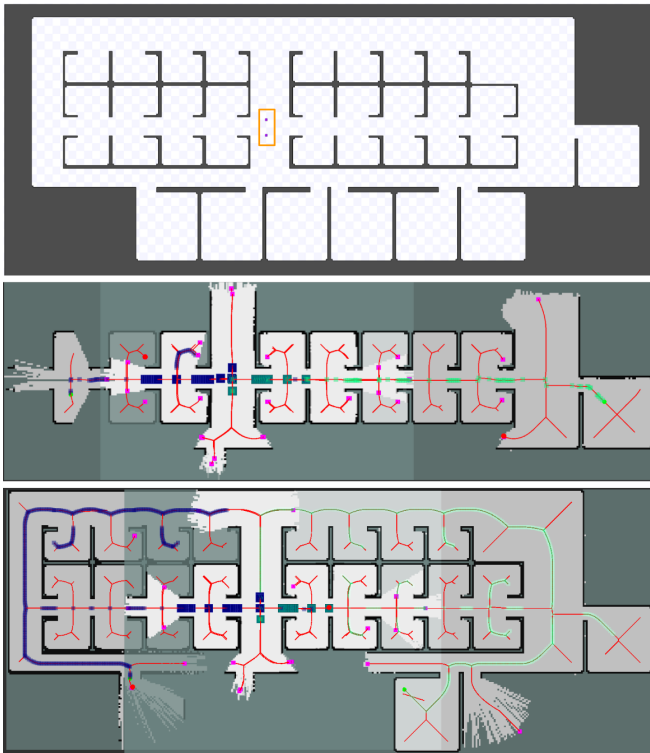


Fig. 5. *GVGExp* simulation in Stage. Top: Robot team (violet points enclosed in a yellow rectangle) in the indoor environment before deployment. Middle: Robots exploring their respective *gate* regions. Square shaped markers show the initial paths from the starting location to the *gates* to which each robot has been assigned, whereas the other green and blue markers trace the nodes traversed by each robot. Bottom: *GVGExp* coverage towards the end of exploration. The robots maintained separation between *gate* regions.

Secondly, it assigns one robot to each cluster such that the robot allocation is balanced across the clusters, subject to the number of available robots. This approach ensures that the robots spread out in the environment. We tested two task allocation methods. The Hungarian algorithm [33] performs a centralized allocation of robots to their respective closest frontiers. An alternative approach is sequential single-item auction [34] where a robot (the auctioneer) executes a number of auction rounds with a set of frontiers, other robots (the bidders) respond with their evaluation, and at each round, one frontier is assigned to a robot. We tested each method in *GVGExp* and compared their exploration time using three robot teams – see Figure 4. Sequential single-item auction registered the best performance across all robot teams, because at every round, priorities of the frontiers could be changed according to the cluster. Thus, we chose the sequential single-item auction for the task allocation method in our system.

For the last two communication events, a conflict resolution is initiated: the *gates* are evaluated by the communication initiator to identify robots that are sharing similar *gates*. If k robots are exploring the *gate*, then the communication initiator randomly selects a robot for that *gate* and informs the other $k - 1$ robots to find other *gates* and spread out. The auction-based task allocation described above will reinitiate

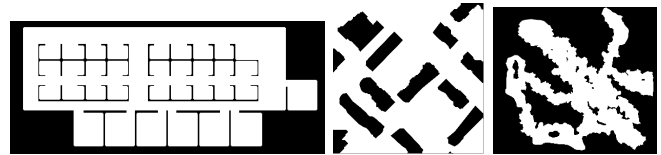


Fig. 6. The office ($82.2\text{ m} \times 34.8\text{ m}$), city ($65\text{ m} \times 65\text{ m}$), and cave ($64\text{ m} \times 57.2\text{ m}$) environments used for the experiments.

with the *gates* that were not assigned to any robot. The illustration of the *GVGExp* operation is in Figure 5.

The robots' allocation to the frontiers determines the *gates*, as introduced in the previous section: robots will only traverse and expand leaves that are within the *gate* region. Note that there might be scenarios where the number of frontiers are less than the number of robots. This results into under-usage of the robots. To address this issue while preserving the minimization of path interference, a robot assigned to a *gate* region can return to the location where the coordination happened and restructure the allocation of *gates*.

V. EXPERIMENTS AND RESULTS

We implemented the proposed exploration and communication strategy in ROS [35] so that the code is reusable across different systems. To validate our method, we ran an extensive set of simulations in Stage [36] – a realistic 2D simulator for multi-robot systems – with ROS-CBT [37] as a communication simulator. The communication simulator implements a communication model used to determine whether or not communication is possible between arbitrary robots. Simulated robots had wheel odometry and a 2D laser sensor with a range of 10 m and a 240° field of view. We selected a range-based communication model with a range of 10 m for all the experiments – a conservative value to ensure communication between robots in real environments with walls made of concrete and metal [38].

We compared our method with two other approaches. The first strategy is called *continuous connectivity* [11]: robots use a frontier-based exploration [6] and share data whenever they obtain new information and are within communication range with other robots. We refer to this strategy as the *baseline*, given that the robots will have more situational awareness than others. The second strategy is called *recurrent connectivity* [8]. This method prescribes robots to share data with a base station whenever a certain amount of new information is obtained. More formally,

$$\text{MapBase}_i / (\text{MapBase}_i + \text{MapNew}_i) < K$$

where, $0 \leq K < 1$ is the target threshold, MapBase_i and MapNew_i are maps, which robot i believes to be at the base station (according to the last shared information with the base station) and the new part of the map that the robot has discovered, respectively. MapBase is updated when the base station receives new information from exploring robots. If a multihop connection can be established with the base station, the robot will not go back to a location where the robot

is in direct connection with the base station. Despite being of different nature, we decided to include this strategy to evaluate the benefits of introducing a central base station that can provide global situational awareness. In our evaluation, we ran this method using two different thresholds, namely, $K = 0.50$ and $K = 0.90$. Rearranging the inequalities, $k = 0.50$ corresponds to a robot that needs twice the information compared to that at the base station before returning to the base station, and $k = 0.90$ corresponds to ~ 0.1 more information. As exploration progresses, robots will return back to the base station less often.

We selected three different environments, with diverse features, from public repositories, namely *office* (from Radish repository [39]), *city* (from a pathfinding benchmark²), and *cave* (using an environment generator³). Their sizes are $82.2\text{ m} \times 34.8\text{ m}$, $65\text{ m} \times 65\text{ m}$, $64\text{ m} \times 57.2\text{ m}$, respectively – see Figure 6.

Our experiment set included 5 runs for a robot team of 2, 4, and 6, deployed in each environment and all starting from the same region in the environment. All experiments were performed on a Ubuntu machine with an Intel i7 CPU with 32GB RAM. Each experiment ran until 90% of a given environment was explored.

We evaluated the performance of each exploration strategy using the following metrics:

- *exploration time* (also called *makespan*) aggregates the time taken to explore up to the target percentage of the environment;
- *established connections* measures the data exchange instances. This metric indicates connectivity and traffic on the network of the robot team;
- *pace* measures the average time robots take to explore one square meter of the environment. This is an indicator of potential path interference between robots.

Note that the number of established connections and pace implicitly assess the energy costs, since more travel time and connectivity directly affect the energy consumed by the robot during the mission. A complete analysis with real hardware is planned for future work.

The computation time for the tested strategies was not included, as they took less than a second to provide the output. The average travel distance is also not reported here, since it shows a similar trend as the exploration time.

The results for *makespan* are summarized in Figure 7. We observed that *GVGExp* had a comparable makespan to the *baseline*. It was followed by *Recurrent_50* ($K=0.5$) and *Recurrent_90* ($K=0.90$) came last. The high performance of *baseline* can be attributed to the continuous data sharing, allowing the robots to have complete situational awareness. However, as shown in Figure 8, the *baseline* method had the highest number of connections across all environments. This was followed by *GVGExp* and *Recurrent_90*, with *Recurrent_50* having the least number of connections. Although *continuous connectivity* had the shortest makespan in most

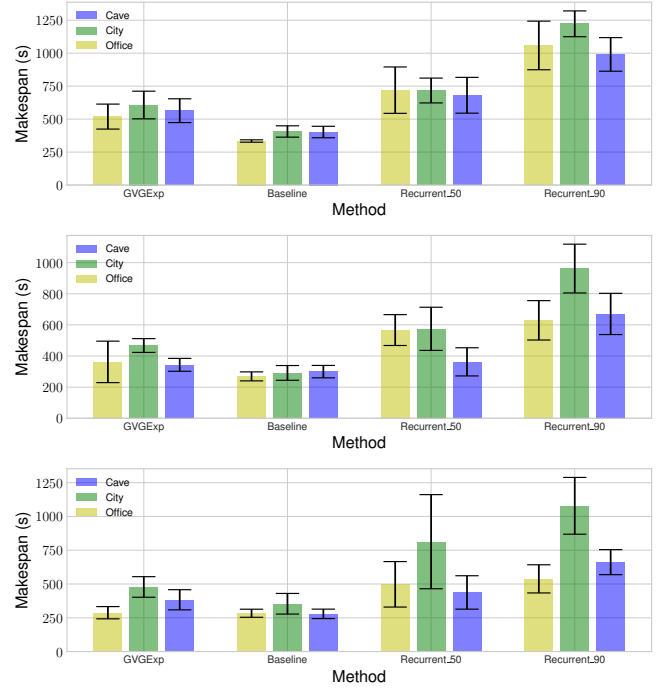


Fig. 7. Aggregated makespan after five runs in each environment with a robot team of 2 (top), 4 (middle) and 6 (bottom) robots.

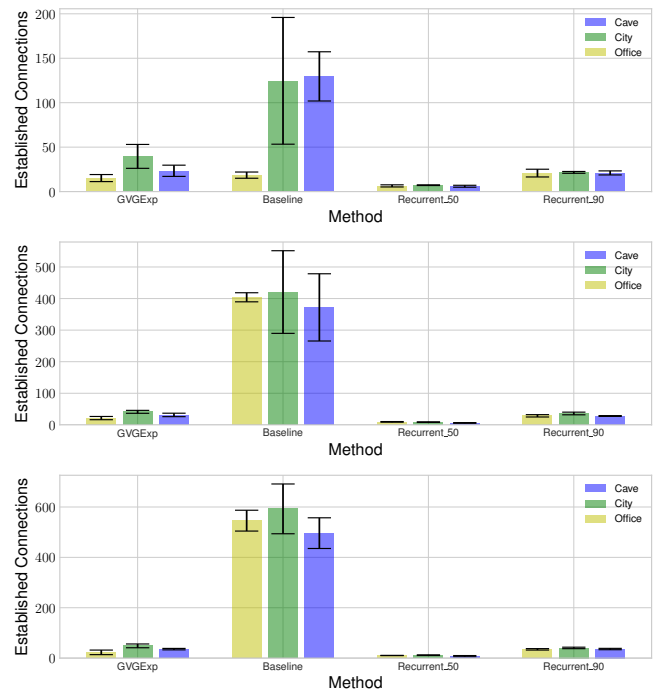


Fig. 8. Number of established connections in different environments with a robot team of 2 (top), 4 (middle) and 6 (bottom) robots.

²<https://movingai.com/benchmarks/index.html>

³<http://www.gozzys.com/cave-maps>

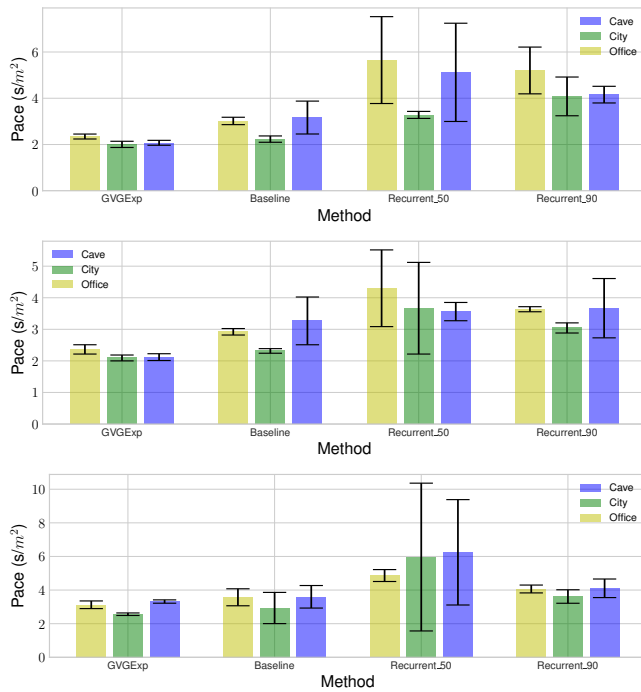


Fig. 9. Average pace for each method in various environments with a robot team of 2 (top), 4 (middle) and 6 (bottom) robots.

experiments, it was clear that it highly depended on the continuous data sharing. This can be a significant drawback especially in situations where network bandwidth is limited. On the other hand, *GVGExp* made much less connections, but with an approximately similar makespan to *continuous connectivity*, which makes it a preferable choice compared to the other methods in such environments.

In addition, the number of connections for the *baseline* method kept increasing as more robots were added to the team. *baseline* had a maximum of 200 connections when running a team of 2 robots. This maximum value grew to approximately 600 connections and 700 connections for 4 and 6 robots, respectively. In contrast, connections for *GVGExp* and the recurrent connectivity methods barely changed as the number of robots increased. This is because their connectivity is dependent on the environment rather than robot proximity as applies to the *baseline*.

Furthermore, we observed that *GVGExp* had the lowest makespan in the office environment. This is because *GVGExp* is highly dependent on the topology of the environment. Since the indoor environment is more structured than the other two environments, robot assignments were well guided by the environment topology.

The results for *pace* are summarized in Figure 9 – the lower the value, the better. *GVGExp* recorded the lowest value of pace followed by the *baseline*, *Recurrent_90* and finally *Recurrent_50*. An example of explored area over time is shown in Figure 10. This can be explained by the enforcement of the *gate*, which effectively spreads out the robots and does not allow them to explore regions that are close to each other. Thus, *GVGExp* is successful in

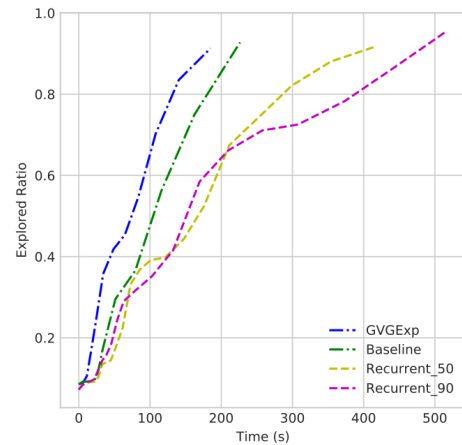


Fig. 10. Explored area over time with 6 robots in the Office environment in one simulation.

minimizing interference between robots.

VI. FUTURE WORK AND CONCLUSION

For future work, we will design a criterion to determine the number of robots that should be assigned to the same branch by further exploiting the topology of the environment, relaxing the current constraint that only one robot should be assigned to a gate region. We plan to explicitly assess the impact of uncertainty in SLAM to the exploration system, in particular the GVG. We also plan to study additional potential path interference cases not explicitly considered by the current method – including when robots are unable to communicate and yet choose the same target location. In addition, we will test the complete exploration system with a larger number of robots, varying communication range, and on real robots, including the Turtlebot 3 and Husarion ROSbot 2.0, with an analysis on power consumption. Finally, we plan to investigate a hybrid system, where both base station and independent-type exploration strategies are potential options for robots to take for exchanging data. The goal is to eventually export such proof-of-concepts to robotic systems in the wild that are heavily limited in communication resources – i.e., underwater, where commercial acoustic modems can transfer data only in the order of tens of kilobit per second – and to advance the autonomy and robustness of multi-robot systems in the field.

In summary, this work proposed a novel multi-robot exploration and communication strategy called *GVGExp*. Using a GVG, we introduced the concept of a *gate* to minimize path interference and to determine when it was necessary to share information for coordinating robots. Experimental results validated that *GVGExp* is an effective approach for exploration-based tasks. It demonstrated low exploration time, comparable to the baseline method based on continuous data sharing, outperforming the other method in terms of reduction in the number of connections made between robots. In addition, *GVGExp* had the best pace, showing that robots were able to minimize their path interference.

Therefore, *GVGExp* is ideal for recurrent connectivity in communication-constrained environments.

ACKNOWLEDGMENT

This work is supported in part by the Burke Research Initiation Award and NSF CNS-1919647, OIA-1923004, CNS-2024541.

The authors would like to thank Monika Roznere and Mingi Jeong for reviewing the manuscript.

REFERENCES

- [1] B. H. Wilcox, "Robotic vehicles for planetary exploration," *Applied Intelligence*, vol. 2, no. 2, pp. 181–193, 1992.
- [2] L. E. Parker, D. Rus, and G. S. Sukhatme, "Multiple mobile robot systems," in *Springer Handbook of Robotics*, Springer, 2016, pp. 1335–1384.
- [3] F. Amigoni, J. Banfi, and N. Basilico, "Multirobot exploration of communication-restricted environments: A survey," *IEEE Intelligent Systems*, vol. 32, no. 6, pp. 48–57, 2017.
- [4] S. Malebary, J. Moulton, A. Quattrini Li, and I. Rekleitis, "Experimental analysis of radio communication capabilities of multiple autonomous surface vehicles," in *OCEANS MTS/IEEE Charleston*, IEEE, 2018, pp. 1–6.
- [5] N. Nowsheen, C. Benson, and M. Frater, "A high data-rate, software-defined underwater acoustic modem," in *OCEANS MTS/IEEE SEATTLE*, IEEE, 2010, pp. 1–5.
- [6] 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.
- [7] J. Banfi, A. Quattrini Li, I. Rekleitis, F. Amigoni, and N. Basilico, "Strategies for coordinated multirobot exploration with recurrent connectivity constraints," *Autonomous Robots*, vol. 42, no. 4, pp. 875–894, 2018.
- [8] V. Spirin, S. Cameron, and J. De Hoog, "Time preference for information in multi-agent exploration with limited communication," in *Conference Towards Autonomous Robotic Systems*, Springer, 2013, pp. 34–45.
- [9] Q. Zhang, D. Whitney, F. Shkurti, and I. Rekleitis, "Ear-based exploration on hybrid metric/topological maps," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2014, pp. 3081–3088.
- [10] Z. Yan, N. Jouandeau, and A. A. Cherif, "Sampling-based multi-robot exploration," in *International Symposium on Robotics (ISR) and German Conference on Robotics (ROBOTIK)*, VDE, 2010, pp. 1–6.
- [11] B. Yamauchi, "Frontier-based exploration using multiple robots," in *International Conference on Autonomous Agents*, 1998, pp. 47–53.
- [12] R. Zlot, A. Stentz, M. B. Dias, and S. Thayer, "Multi-robot exploration controlled by a market economy," in *IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, vol. 3, 2002, pp. 3016–3023.
- [13] B. P. Gerkey and M. J. Matarić, "A formal analysis and taxonomy of task allocation in multi-robot systems," *The International Journal of Robotics Research*, vol. 23, no. 9, pp. 939–954, 2004.
- [14] A. Quattrini Li, R. Cipolleschi, M. Giusto, and F. Amigoni, "A semantically-informed multirobot system for exploration of relevant areas in search and rescue settings," *Autonomous Robots*, vol. 40, no. 4, pp. 581–597, 2016.
- [15] F. Amigoni, J. Banfi, and N. Basilico, "Multirobot exploration of communication-restricted environments: A survey," *IEEE Intelligent Systems*, vol. 32, no. 6, pp. 48–57, Nov. 2017.
- [16] A. Quattrini Li, "Exploration and mapping with groups of robots: Recent trends," *Current Robotics Reports*, pp. 1–11, 2020.
- [17] M. N. Rooker and A. Birk, "Multi-robot exploration under the constraints of wireless networking," *Control Engineering Practice*, vol. 15, no. 4, pp. 435–445, 2007.
- [18] E. A. Jensen, L. Lowmanstone, and M. Gini, "Communication-restricted exploration for search teams," in *Distributed Autonomous Robotic Systems*, Springer, 2018, pp. 17–30.
- [19] R. Pandey, A. K. Singh, and K. M. Krishna, "Multi-robot exploration with communication requirement to a moving base station," in *IEEE International Conference on Automation Science and Engineering (CASE)*, IEEE, 2012, pp. 823–828.
- [20] A. C. Kapoutsis, S. A. Chatzichristofis, L. Doitsidis, J. B. de Sousa, J. Pinto, J. Braga, and E. B. Kosmatopoulos, "Real-time adaptive multi-robot exploration with application to underwater map construction," *Autonomous robots*, vol. 40, no. 6, pp. 987–1015, 2016.
- [21] T. Nestmeyer, P. R. Giordano, H. H. Bühlhoff, and A. Franchi, "Decentralized simultaneous multi-target exploration using a connected network of multiple robots," *Autonomous robots*, vol. 41, no. 4, pp. 989–1011, 2017.
- [22] Y. Pei, M. W. Mutka, and N. Xi, "Connectivity and bandwidth-aware real-time exploration in mobile robot networks," *Wireless Communications and Mobile Computing*, vol. 13, no. 9, pp. 847–863, 2013.
- [23] G. A. Hollinger and S. Singh, "Multirobot coordination with periodic connectivity: Theory and experiments," *IEEE Transactions on Robotics*, vol. 28, no. 4, pp. 967–973, 2012.
- [24] F. Amigoni, J. Banfi, N. Basilico, I. Rekleitis, and A. Quattrini Li, "Online update of communication maps for exploring multirobot systems under connectivity constraints," in *Distributed Autonomous Robotic Systems*, Springer, 2019, pp. 513–526.
- [25] K. Okumura, Y. Tamura, and X. Défago, "Amoeba exploration: Coordinated exploration with distributed robots," in *International Conference on Aware-*

- ness Science and Technology (iCAST), IEEE, 2018, pp. 191–195.
- [26] F. Amigoni, J. Banfi, A. Longoni, and M. Luperto, “Online switch of communication modalities for efficient multirobot exploration,” in *European Conference on Mobile Robots (ECMR)*, IEEE, 2017, pp. 1–6.
- [27] J. Kim, F. Zhang, and M. Egerstedt, “A provably complete exploration strategy by constructing voronoi diagrams,” *Autonomous Robots*, vol. 29, no. 3-4, pp. 367–380, 2010.
- [28] K. M. Wurm, C. Stachniss, and W. Burgard, “Coordinated multi-robot exploration using a segmentation of the environment,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2008, pp. 1160–1165.
- [29] P. Brass, F. Cabrera-Mora, A. Gasparri, and J. Xiao, “Multirobot tree and graph exploration,” *IEEE Transactions on Robotics*, vol. 27, no. 4, pp. 707–717, 2011.
- [30] J. L. S. Rincon and S. Carpin, “Time-constrained exploration using toposemantic spatial models: A reproducible approach to measurable robotics,” *IEEE Robotics & Automation Magazine*, vol. 26, no. 3, pp. 78–87, 2019.
- [31] S. Thrun, W. Burgard, and D. Fox, *Probabilistic robotics*. Cambridge, Mass.: MIT Press, 2005.
- [32] E. Schubert, J. Sander, M. Ester, H. P. Kriegel, and X. Xu, “DBSCAN revisited, revisited: Why and how you should (still) use DBSCAN,” *ACM Transactions on Database Systems (TODS)*, vol. 42, no. 3, pp. 1–21, 2017.
- [33] H. W. Kuhn, “The Hungarian method for the assignment problem,” *Naval research logistics quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955.
- [34] S. Koenig, C. Tovey, M. Lagoudakis, V. Markakis, D. Kempe, P. Keskinocak, A. Kleywegt, A. Meyerson, and S. Jain, “The power of sequential single-item auctions for agent coordination,” in *AAAI Conference on Artificial Intelligence*, 2006, pp. 1625–1629.
- [35] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, “Ros: An open-source robot operating system,” in *ICRA workshop on open source software*, vol. 3, 2009, p. 5.
- [36] R. Vaughan, “Massively multi-robot simulation in stage,” *Swarm intelligence*, vol. 2, no. 2-4, pp. 189–208, 2008.
- [37] K. Masaba and A. Quattrini Li, “ROS-CBT: Communication benchmarking tool for the robot operating system,” in *IEEE International Symposium on Multi-Robot and Multi-Agent Systems (MRS)*, 2019, pp. 1–3.
- [38] A. Quattrini Li, P. K. Penumarthy, J. Banfi, N. Basilico, J. M. O’Kane, I. Rekleitis, S. Nelakuditi, and F. Amigoni, “Multi-robot online sensing strategies for the construction of communication maps,” *Autonomous Robots*, vol. 44, no. 3, pp. 299–319, 2020.
- [39] A. Howard and N. Roy, *The robotics data set repository (Radish)*, <http://radish.sourceforge.net/>, 2003.