

Article

Occupancy Grid Mapping via Resource-Constrained Robotic Swarms: A Collaborative Exploration Strategy

Andrew Rogers ^{*}, Kasra Eshaghi , Goldie Nejat  and Beno Benhabib

Department of Mechanical and Industrial Engineering, University of Toronto, Toronto, ON M5S 3G8, Canada

^{*} Correspondence: a.rogers@mail.utoronto.ca

Abstract: This paper addresses the problem of building an occupancy grid map of an unknown environment using a swarm comprising resource-constrained robots, i.e., robots with limited exteroceptive and inter-robot sensing capabilities. Past approaches have, commonly, used random-motion models to disperse the swarm and explore the environment randomly, which do not necessarily consider prior information already contained in the map. Herein, we present a collaborative, effective exploration strategy that directs the swarm toward ‘promising’ frontiers by dividing the swarm into two teams: landmark robots and mapper robots, respectively. The former direct the latter, toward promising frontiers, to collect proximity measurements to be incorporated into the map. The positions of the landmark robots are optimized to maximize new information added to the map while also adhering to connectivity constraints. The proposed strategy is novel as it decouples the problem of directing the resource-constrained swarm from the problem of mapping to build an occupancy grid map. The performance of the proposed strategy was validated through extensive simulated experiments.

Keywords: swarm robotics; motion control; exploration; swarm mapping; occupancy grid map



Citation: Rogers, A.; Eshaghi, K.; Nejat, G.; Benhabib, B. Occupancy Grid Mapping via Resource-Constrained Robotic Swarms: A Collaborative Exploration Strategy. *Robotics* **2023**, *12*, 70. <https://doi.org/10.3390/robotics12030070>

Academic Editor: Guanghui Wen

Received: 22 March 2023

Revised: 3 May 2023

Accepted: 5 May 2023

Published: 9 May 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Swarm Robotic Systems (SRSs) comprise multiple fully-autonomous or semi-autonomous robots that collaborate with each other in order to accomplish a task that is often too complex for an individual robot to complete on its own. Numerous works have categorized common problems and applications specific to SRS [1–5]. Behaviors of SRSs have been, commonly, investigated using ground-based robots, including millimeter-scale robots known as millirobots [6–11]. As well, strategies have also been specifically developed for swarms comprising aerial vehicles [12–17]. Many applications of SRS have been studied including object manipulation [2,18,19], environmental monitoring [20,21], and search and rescue [22–24]. In this paper, our focus is on the use of ground-based millirobots that lack extensive proximity-sensing capabilities (e.g., omnidirectional laser scanners) often seen on larger robot platforms. Namely, such SRSs are resource-constrained, having limited exteroceptive sensing (e.g., unidirectional sensors, local inter-robot proximity sensing) and, thus, mainly rely on inter-agent communications to accomplish a goal.

The specific swarm problem investigated, herein, is the mapping of an environment using millirobot-based SRSs with limited proximity-sensing capabilities. In this context, occupancy grid mapping is a potential approach that can be utilized [25]. Building such maps, however, would require robots to accurately estimate their own locations and gather significant amounts of proximity-sensing information from their environment, both of which may be non-trivial tasks for a resource-constrained SRS. Thus, in this paper, we propose the development of an exploration strategy that utilizes a resource-constrained SRS to build an occupancy grid map of an unknown environment.

Exploration of an environment via a SRS can be achieved by having each robot use random motion to disperse itself throughout the environment [5]. However, in order for

robots not to “wander” away from other robots within the swarm, the environment must remain “enclosed”, or the random motion otherwise constrained in some way. The pertinent literature, investigating the use of resource-constrained swarms to build occupancy grid maps, has proposed random motion exploration to diffuse robots only throughout an enclosed environment [26–28]. In these works, all robots use the same random motion model to disperse themselves independently of the other robots in the swarm. Furthermore, information contained within the map is not used to influence the exploration area of the swarm.

In contrast to the abovementioned methods, herein, we propose a unique exploration strategy for a resource-constrained SRS whereby the mapping of an unknown environment may be directed to maximize new information added to an occupancy grid map. The swarm is divided into two teams: a decentralized team of mapper robots and a centralized team of landmark robots that may communicate with a central controller. The mapper robots explore the environment using a random motion model and gather information to be incorporated into the map, although they are constrained to stay nearby the team of landmark robots. The team of landmark robots, on the other hand, receive motion commands from a central controller and direct the exploration of the swarm toward a particular area of the environment.

Our strategy proposes the first use of a resource-constrained swarm, divided into two teams, to physically decouple the problems of exploration and mapping of unknown environments. The swarm’s exploration can be directed towards areas that maximize new information added to the map while still maintaining the overall scalability of the strategy to a large number of robots. Directing the swarm, toward the most promising areas to explore, is performed via planning and executing the landmark robots’ motions. Mapping of those areas is performed via a separate team of mapper robots, each following a random motion model to disperse themselves and collect information from their local environment. The swarm is, therefore, able to follow an efficient frontier-based exploration of the environment while the map is built using robots with limited exteroceptive sensing.

The strategy is novel as it allows the problem of directing a swarm to a new area of the environment to be addressed separately from the problem of exploring and mapping that new area using robots with limited environment perception abilities. This physical decoupling promotes swarm scalability as the landmark robots’ motions can be planned independently of the number of mapper robots.

2. Related Works

Herein, the pertinent literature related to exploration and mapping of unknown environments using SRSs is detailed in Sections 2.1–2.4. A comparison of the proposed strategy to each of the categories described is provided in Section 2.5.

2.1. Metric Mapping Using Homogeneous Swarms with Limited Exteroceptive-Sensing

Metric map creation using resource-constrained swarms, as addressed herein, presents a unique challenge compared to topological map creation since generating metric maps (e.g., occupancy grid maps), generally, requires accurate environment perception and localization. Previous studies, investigating the creation of metric maps using resource-constrained swarms, have used various random-motion models to explore the environment [26–28]. Random-motion models specialized in either an intensive exploration of a small area or coverage of a larger area were investigated in the context of building an occupancy grid map using odometry-based localization alone [27]. Other studies have allowed the swarm to share relative proximity measurements to improve the overall swarm localization estimate. However, each robot in the swarm still uses a random motion model to disperse themselves without considering which areas of the environment may already have been explored [26,28]. In these works, the swarm’s motion is unconstrained within an enclosed environment.

In contrast to the abovementioned works, our proposed work allows the resource-constrained swarm's exploration to be directed toward unmapped areas in the environment by dividing the swarm into a team of mapper robots and a team of landmark robots, respectively. The latter directs the decentralized mapper robots. The novelty of our proposed work is based on this division of the swarm into two teams, one to direct the swarm toward new areas and the other to map those new areas. Namely, the landmark robots are moved to unexplored areas of the environment to facilitate the building of a map as efficiently as possible. The use of a separate team of landmark robots enables the mapper robots to remain decentralized while still being directed toward the unmapped areas of the environment.

2.2. Topological Mapping Using Swarms with Limited Exteroceptive-Sensing

Topological maps are a sparse representation of an environment capturing the topological relationship of obstacles or features, similar to a roadmap [29]. In direct pertinence to our work, topological maps have been built using resource-constrained swarms whose individual robots may not be able to position themselves or incorporate sensor measurements into a map on their own (e.g., [30–36]). The relative simplicity of each robot and the large number of robots within these swarms prohibit the use of existing multi-robot SLAM methods.

Topological maps have been built using a swarm that navigates in an environment with many landmarks, whose density and distribution in the environment are known [30]. Similarly, the construction of physical topological maps has also been investigated, where the swarm's topology represents the map itself [31].

There also have been approaches that build maps offline once the resource-constrained swarm has gathered data from the environment. In [32], for example, the swarm remains unlocalized and records information, such as time of encounter with an obstacle, in order to map a feature-of-interest. Alternatively, in [33–35], a topological map is built by having each robot move randomly through the environment and record when it encounters another swarm robot, while each robot remains unlocalized. In these studies, the location of robots encountering each other are converted into point clouds and, later, into topological maps, using metric estimation and topological data analysis (TDA) techniques [33–35]. Random-motion swarms and TDA techniques have also been used to extract the number of obstacles inside an unknown environment [36]. Millirobot-scale ground swarms are more commonly used in this category since extensive environment perception or accurate localization is not required. Our work, however, focuses on the creation of metric maps as opposed to topological maps using resource-constrained swarms.

2.3. Mapping Using Robotic Systems with “High-Information” Exteroceptive-Sensors

Cooperative exploration of unknown environments using SRSs has been previously investigated for swarms equipped with “high-information” sensors (e.g., omnidirectional laser scanners), where each robot can explore and map their local environment individually [37–41].

Such systems often use existing simultaneous localization and mapping (SLAM) methods and collaborate to map the environment faster than each robot otherwise could individually. In this regard, the cooperative exploration and mapping problem for these swarms can be seen as a subset of the more general multi-robot SLAM problem [42,43], with emphasis on the scalability of methods for SRSs [44].

In the above context, maps can be built by storing one centralized map that each robot contributes information to or by having each robot store their own individual map of the environment. Centralized maps have been used with efficient decentralized frontier-based exploration methods to quickly disperse agents throughout the environment [37]. Alternatively, each swarm robot may store their own map while exchanging information with their neighbors to merge their neighbors map with their own and better plan their own trajectories [38–40]. In [41], for example, decentralized exploration and distributed

mapping are used in combination to map dynamic environments using teams of wall-climbing robots and a swarm of ground-level mapper robots. Our work, instead, focuses on resource-constrained swarms, whose limited sensing capabilities prohibit the member robots from building, or contributing to, a map individually. Such swarms must collaborate with each other to achieve these tasks.

2.4. Area Coverage Using Homogeneous Swarms with Limited Exteroceptive-Sensing

Area coverage of an environment has been investigated using SRS where the primary problem being addressed is the efficient exploration and coverage of a target area. Although maps of the environment are not necessarily built, the exploration strategies used to cover the environment are still of interest. In this regard many strategies have investigated the use of random motion models by each robot in the swarm to disperse the robots across the target area [45–48].

In [45], the strategy combines a random motion model based on the Lévy distribution (Lévy flight) with an artificial potential field to improve the exploration efficiency, with the potential field generating a repulsive vector between nearby robots to promote dispersion of the swarm. In [46], a control law is proposed based on pheromones to switch the random motion models between Brownian motion for local searching and Lévy flight motion for global exploration. In [47], an improved Lévy flight random motion model is proposed that adaptively changes the models step size based on the swarm’s density and is shown to provide a more uniform coverage of an environment than when using a standard Lévy flight random motion model alone. In [48], different random motion models were tested in a variety of environments using a swarm of Kilobots to reveal which motion model provided the most effective coverage of a particular environment. Our work, however, constrains the random motion of mapper robots to only cover the area surrounding landmarks, and considers how measurements taken during this random exploration can be used to build a map of the environment.

2.5. Comparison Summary

In order to summarize and compare the pertinent literature with the strategy presented herein, Table 1 provides a short description of each of the above categories as well as a list of pros and cons related to each. As can be seen from the table, our proposed decoupling strategy for a swarm of robots provides clear advantages over existing strategies that also have limited exteroceptive sensing capabilities, including creating high-fidelity metric maps of the environment using a frontier-based optimally directed exploration strategy that maintains the overall scalability to a large number of robots.

Table 1. Comparison of our proposed strategy with others in the pertinent literature.

Category	Description	Pros	Cons
Proposed strategy: Heterogenous swarms with limited exteroceptive-sensing	<ul style="list-style-type: none"> • Uses a group of landmark robots to optimally direct a separate team of mapper robots. • Mapper robots follow random motion. • Builds occupancy grid maps by localizing individual robots and incorporating their proximity measurements into the map. 	<ul style="list-style-type: none"> • Frontier-based optimally directed exploration of the environment: <ul style="list-style-type: none"> • Robots maintain connectivity, allowing for accurate position estimates using cooperative localization strategies. • Strategy is scalable since only the landmark robots’ motions need to be planned by a central controller. • High-fidelity metric maps are built using information from relatively simple exteroceptive sensors. 	<ul style="list-style-type: none"> • Motion planning is not fully decentralized. • Map building is centralized.

Table 1. Cont.

Category	Description	Pros	Cons
Homogeneous swarms with limited exteroceptive-sensing [26–28,45–48]	<ul style="list-style-type: none"> Each robot follows a random motion, and exploration of the environment occurs as a consequence of this (random) motion. Builds occupancy grid maps by localizing individual robots and incorporating their proximity measurements into the map. 	<ul style="list-style-type: none"> Fully decentralized motion planning. Strategy is scalable. High-fidelity metric maps are built using information from relatively simple exteroceptive sensors. 	<ul style="list-style-type: none"> Exploration is undirected and, thus, sub-optimal: <ul style="list-style-type: none"> Robots explore previously mapped areas. Map building is centralized. Robots may lose connectivity, which may lead to inaccurate position estimates and, thus, inaccurate maps.
Topological mapping using swarms with limited exteroceptive-sensing [30–36]	<ul style="list-style-type: none"> Robots explore the environment through random motion. Data is first collected by a swarm and later used to build a topological map (e.g., roadmap): <ul style="list-style-type: none"> Data collected includes time-of-encounter with other robots or obstacles. 	<ul style="list-style-type: none"> Fully decentralized motion planning. Strategy is scalable. Minimal exteroceptive sensing required. Topological maps can be built even with no position estimates of the robots. 	<ul style="list-style-type: none"> Large numbers of robots, that need to disperse uniformly throughout the environment, are required. Map building is centralized. Only low-fidelity map of the environment can be achieved.
Robotic systems with “high-information” exteroceptive-sensors [37–41]	<ul style="list-style-type: none"> Similar to multi-robot simultaneous localization and mapping. Directs exploration to map an area. Each robot individually builds a map of an area without relying on other robots. Strategies are made for and implemented with a few robots (e.g., 2 to 5 robots). 	<ul style="list-style-type: none"> Frontier-based directed exploration of the environment: Efficient exploration can be achieved due to complex sensors. High-fidelity metric maps are built. 	<ul style="list-style-type: none"> Computationally infeasible for large-sized swarms (i.e., not scalable). Requires high-information, complex and, often, large-sized sensors and, thus, large-scaled robots.

3. Problem Formulation

As noted above, this paper focuses on the development of a generalized exploration strategy to build an occupancy grid map of an a priori unknown environment via a resource-constrained swarm robotic system (SRS). It is assumed that the environment may include a priori unknown obstacles and may not be enclosed, i.e., bordered.

The swarm is to be divided into two teams prior to the start of a task. The first is a team of n_L (mobile) landmark robots that use a centralized architecture to communicate with a central controller, where each landmark robot can be denoted by the unique label, $L_j, j \in \{1, \dots, n_L\}$. The second is a team of n_m mapper robots that use a decentralized architecture and communicate with the landmark robots, where each mapper robot can be denoted by the unique label, $R_i, i \in \{1, \dots, n_m\}$. The landmark robots are responsible for guiding the exploration of the swarm by physically positioning themselves in promising areas to explore. The mapper robots are responsible for acquiring the sensor measurements to build a map of the environment surrounding the landmark robots.

It is assumed that the mapper robots can acquire information about their local environment through on-board sensing hardware, such as laser range finders, RGB cameras. Let us denote the exteroceptive sensory measurement that acquires information (e.g., distance to obstacle) about the surrounding environment taken by mapper robot R_i at time t as $z_{R_i}^t$.

The swarm is assumed to be initially positioned at a (fixed home) base, ${}^G H = (x_H, y_H)$, defined by a global frame, ${}^G F$. It is also assumed that a static sensor that can actively sense the swarm could be placed at the home base prior to execution. The sensory capabilities of each robot in the swarm include their ability to uniquely identify and measure relative

ranges and bearings to their neighbors within a predefined circle-of-radius, ρ_{sense} . Swarm robots within ρ_{sense} of each other are said to be connected. Our work considers all robots, including both landmark and mapper robots, to be equipped with inter-robot sensing technology that allows them to measure distance and bearing to their neighbors. The measurement taken by mapper robot R_i of some other mapper robot R_j can be denoted as:

$$s_{R_i R_j} = (\hat{d}_{R_i R_j}, \hat{b}_{R_i R_j}), \quad (1)$$

where $s_{R_i R_j}$ is the inter-robot sensing of Robot R_j by Robot R_i : $\hat{d}_{R_i R_j}$ and $\hat{b}_{R_i R_j}$ are the relative distance and bearing between these robots, as observed by R_i , respectively. In practice, proximity measurements would be the sum of the true relative distance and bearing between robots, plus some additional zero-mean Normally-distributed noise:

$$\hat{d}_{R_i R_j} = d_{R_i R_j} + N(0, \sigma_{int}^2), \text{ and} \quad (2)$$

$$\hat{b}_{R_i R_j} = b_{R_i R_j} + N(0, \sigma_{int}^2) \quad (3)$$

where $d_{R_i R_j}$ and $b_{R_i R_j}$ are the true distances and bearings between the robots, and σ_{int}^2 is the variance of the additive Normally-distributed noise. Since all robots in the swarm are equipped with inter-robot sensing capabilities, measurements can equivalently be taken between mapper and landmark robots. All inter-robot sensing measurements taken by Robot R_i of other nearby robots at time t can be combined into a set and denoted as $S_{R_i}^t$:

$$S_{R_i}^t = \{s_{R_i A}\} \forall \text{ robots } A \text{ s.t. } dist(R_i, A) < \rho_{sense}, \quad (4)$$

where A represents a general label for either a mapper or landmark robot and the function $dist(R_i, A)$ represents the distance between mapper Robot R_i and mapper or landmark Robot A .

Mapper robots continually broadcast measurements in the form of *data packets* that include both inter-robot distance and bearing measurements as well as distance measurements to their local environment. Namely, each data packet includes a tuple of measurements from each type of sensor mentioned above as well as the identifier of the mapper robot itself. Landmark robots receive these data packets and forward the information to a central controller along with their own inter-robot measurements, at which point the inter-robot distance and bearing measurements may be fused to obtain an estimate of the robots' locations in the environment using cooperative swarm localization techniques [49–60]. Let us denote the data packet sent by mapper Robot R_i at time t as $D_{R_i}^t = \{S_{R_i}^t, z_{R_i}^t, i\}$.

Per the discussion above, the team of mapper robots is responsible for locally exploring and acquiring data packets from the area surrounding the landmark robots. The team of landmark robots, on the other hand, can communicate with a central controller and is responsible for directing the exploration of the swarm toward a particular area of the environment. One must note that, while the mapper robots explore and broadcast data packets, the landmark robots remain stationary until given motion commands by a central controller. When the landmarks receive motion commands, the mapper robots follow as they are constrained to stay nearby the landmarks. Thus, by directing the landmark robots toward a particular area of the environment, the effective exploration area of the mapper robots is indirectly controlled via the central controller.

The main problem to be addressed is, thus, exploration of the environment through optimal motion planning for both the mapper and landmark robots, in order to acquire sensor measurements and build an *occupancy grid map*. The occupancy grid map is a 2D matrix of cells with each cell having an associated occupancy probability denoting the probability that the cell is obstructed by an obstacle [61]. The central controller builds the

map, M , fixed with respect to the global frame ${}^G F$, which is to be updated by the latest proximity measurements.

Per abovementioned, the swarm exploration problem comprises two main sub-tasks: mapper-robot motion planning and landmark-robot motion planning, as discussed below in Sections 3.1 and 3.2, respectively.

3.1. Mapper-Robot Motion Planning

For the proposed strategy, the mapper robots need to explore the local area surrounding the landmark robots and "follow" them to new areas in the environment, all the while broadcasting measurements in the form of data packets. Since the mapper robots can only sense the landmark robots within a limited radius, ρ_{sense} , their motion should be constrained to stay connected to the landmark robots within ρ_{sense} .

Due to the decentralized nature of their control, each mapper robot's controller must independently address the following problems: (i) dispersing itself throughout the explorable area while broadcasting data packets and avoiding obstacles, and (ii) staying connected to the landmark robots, even as the latter move to new areas in the environment.

In order to avoid the mapper robots from becoming disconnected, as the landmark robots move, it may be necessary to further constrain their motion to less than the maximum sensing radius to account for cases in which the landmark robot moves outside of the sensing range before the mapper robot could react. One can define the radius of allowable exploration away from the landmark robots, $\rho_{explore}$, as a function of ρ_{sense} , using a safety factor, $s_f > 1$:

$$\rho_{explore} = \frac{\rho_{sense}}{s_f}. \quad (5)$$

3.2. Landmark-Robot Motion Planning

The landmark robots' motions are planned within the central controller and transmitted to them in the form of motion commands, comprising two stages:

1. *Position planning*: Given the current map, M , a set of positions (nodes) for the landmarks need to be determined. In our work, the positions of the landmarks indirectly dictate where exploration of the environment would occur.
2. *Path planning*: The landmark robots' paths, from their current positions to the next selected set of planned nodes, need to be determined, given the known obstacles and free spaces within the map. Motion commands for the execution of each path are generated and sent to the landmark robots individually by the central controller.

3.2.1. Position Planning

In the proposed strategy, the positions of the landmark robots affect where exploration of the environment will occur. In order to explore the entirety of an environment, the landmark robots must move to within exploration range, $\rho_{explore}$, of previously unmapped areas so that the mapper robots can explore those unmapped regions.

The positions of the landmark robots should be constrained such that they remain connected. One can define this connectivity in the context of graph theory, where the positions of the landmarks form the nodes of a graph with edges between nodes being less than ρ_{sense} of each other. The landmark robots should, then, be constrained such that their corresponding graph remains connected. This connectivity requirement allows for the team of landmark robots to share relative proximity and bearing to each other and, thus, improve their localization estimate using cooperative strategies [49–60]. Furthermore, it ensures that the mapper robots can explore the region surrounding all the landmark robots.

In this regard, the k th set of planned landmark robot nodes, ${}^G L_k$, with n_L nodes can be defined as

$${}^G L_k = \left\{ {}^G x_{L_j} \right\}_{j=1}^{n_L}, \quad (6)$$

where the position of the nodes ${}^G\mathbf{x}_{L_j}$ are defined with respect to the global frame, GF . The connectivity constraint can, then, be given as

$$\left\{ \text{dist}\left({}^G\mathbf{x}_{L_j}, {}^G\mathbf{x}_{L_{j+1}}\right) \right\}_{j=1}^{n_L-1} \leq \rho_{\text{sense}}, \quad (7)$$

where $\text{dist}\left({}^G\mathbf{x}_{L_j}, {}^G\mathbf{x}_{L_{j+1}}\right)$ represents the Euclidean distance between consecutive landmark nodes.

Additionally, given a set of planned nodes, ${}^G\mathbf{L}_k$, within a map, M , the area surrounding the nodes within a circle-of-radius ρ_{explore} can be explored by the mapper robots. One can define the subset of this exploration area that has not been mapped as the *information gain*, $I({}^G\mathbf{L}_k)$. The information gain represents how much new area would potentially be added to the map if the landmark robots were to move to their respective planned nodes, ${}^G\mathbf{L}_k$. Estimates of $I({}^G\mathbf{L}_k)$, given M , can be determined simply as the area of unknown occupancy grid cells within ρ_{explore} of each planned node since it is this area that would be newly added to the map [62].

Thus, the primary problem to be addressed at this stage is to determine the best possible planned positions of the landmark robots, ${}^G\mathbf{L}_k^*$. The optimization objective function for this problem could be, for example, to maximize the information gain, ${}^G\mathbf{L}_k$,

$$\max I\left({}^G\mathbf{L}_k\right) = f\left({}^G\mathbf{L}_k, M, \rho_{\text{explore}}\right). \quad (8)$$

Above, the function f describes the dependance of information gain on the positions of the planned nodes, the latest map, and the radius of exploration surrounding the landmark robots.

3.2.2. Path Planning

Once the optimal nodes for the landmark robots have been determined, the next problem at hand is to build point-to-point (PTP) paths for their motions between their current positions and next planned nodes. For each landmark robot, a path consisting of a series of intermediate points, from its current position to a planned node position, should be determined in order to avoid known obstacles in the environment. The main sub-problems to be addressed are: (i) building collision-free paths, and (ii) ensuring continuous connectivity of the landmark robot team with the mapper robots, while moving along their PTP path.

The path-planning stage should consider the decentralized controller used by the mapper robots. The speed at which the landmark robots traverse the environment may need to be adjusted to ensure that the mapper robots stay connected to the landmark robots as they move through the environment.

4. Proposed Methodology

The proposed mapping strategy requires a swarm to be divided into two distinct groups: a team of (decentralized) mapper robots that would gather information about the environment, and a team of (centralized) landmark robots that direct the mapper robots towards specific areas of the environment. A high-level overview of the proposed strategy is shown in Figure 1. Mapping the environment is an iterative process, with each iteration involving local mapping of the area surrounding the landmark robots and then, moving the landmark robots to a new area to explore. The former is achieved by constraining the motion of the mapper robots to be nearby the landmark robots and incorporating the mapper robots' measurements into the map, as detailed in Section 4.1. Once the latest area at hand has been explored, the landmark robots move to another 'promising' area, as detailed in Section 4.2.

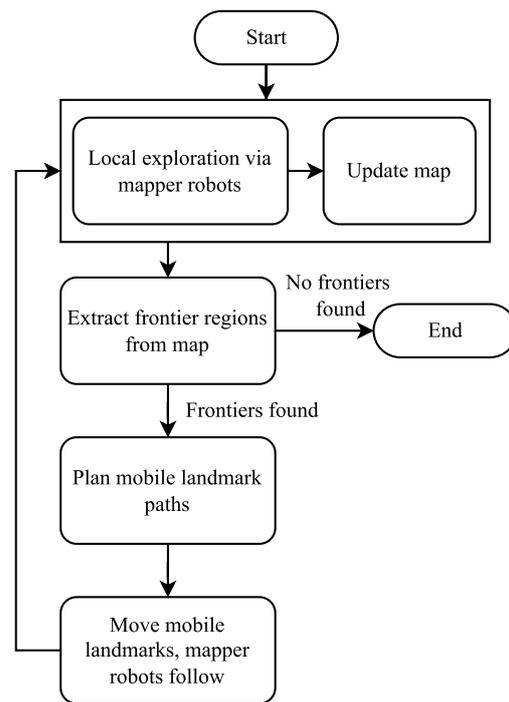


Figure 1. Flowchart of mapping method.

The proposed strategy allows the swarm to prioritize exploration and mapping of unknown areas in the environment. The novelty of the strategy is in the decoupling of the swarm into two teams: one to direct the swarm toward an area in the environment, and the other to acquire measurements to build the map of that area. A team of landmark robots directs the swarm toward an area of the environment that maximizes the potential for new information to be added to the map. A separate team of decentralized mapper robots gather measurements to be incorporated into the map, while staying nearby the landmark robots. This decoupling allows the landmark robots to define where exploration and mapping should occur, independent from the mapper robots.

4.1. Mapper-Robot Motion Planning

As the first stage of the proposed strategy, the local area surrounding the landmark robots is explored using a decentralized team of mapper robots. The sensing capabilities, as well as the motion controller, for the mapper robots is outlined below in Section 4.1.1. Localization of the mapper robots is, then, discussed in Section 4.1.2 and the updating process of the map is discussed in Section 4.1.3, respectively.

4.1.1. Mapper-Robot Sensing and Motion Models

The mapper robots must remain within communication range, ρ_{sense} , of the landmark robots. Our work considers all robots to be equipped with inter-robot sensing technology that allows them to measure the distance and bearing to their neighbors as defined in Section 3.

We assume that each mapper robot is equipped with an additional exteroceptive sensor capable of acquiring information from its local environment. It is measurements from this sensor that, once localized to the global frame of the map ${}^G F$, will be incorporated into the map. In our work, mapper robots are equipped with a single laser distance sensor, pointing straight ahead, capable of measuring distance to an object within range z_{max} and field-of-view β_z . The distance sensor measurement taken at time t by Robot R_i is denoted as $z_{R_i}^t$.

In order to diffuse themselves throughout the local environment surrounding the landmark robots, herein, it is assumed that the mapper robots use a random-walk behavior

to decide in which direction each robot should move, similar to other random motion swarm exploration methods noted in the literature (e.g., [26–28,32–34]). In contrast these random motion exploration methods, however, the mapper robots, in our work, are constrained to stay within a fixed exploration radius of the landmark robots, $\rho_{explore}$. Each mapper robot defines a random direction to follow, α , and moves in this direction until either an obstacle is detected or it detects the nearest landmark to be further than its predefined exploration radius, $\rho_{explore}$. In such cases, a new direction is selected to avoid collisions or stay nearby the landmark robots. Figure 2 outlines this random motion model in the form of a finite-state machine.

$\hat{d}_{landmark}, \hat{b}_{landmark}$ = distance & bearing to nearest mobile landmark

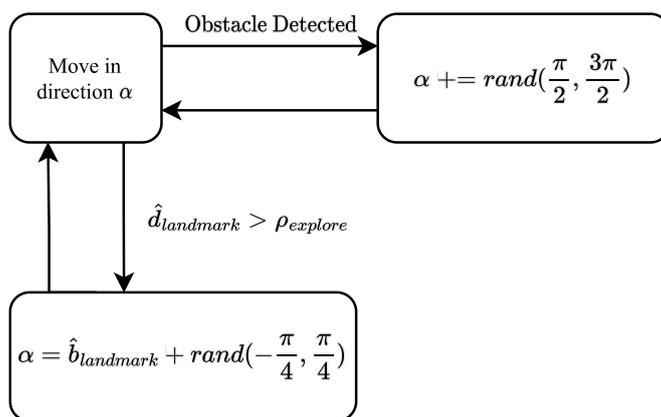


Figure 2. A finite-state machine representing mapper robot motion.

4.1.2. Swarm Localization

Localization of both the mapper robots and landmark robots is performed in the proposed strategy by a central controller by fusing inter-robot sensor measurements, $s_{R_i R_j}$. First, though, estimates of the landmark robots’ positions, ${}^G \hat{L}_k$, are obtained prior to the local exploration conducted by the mapper robots.

Accurate estimates of the landmark robots’ positions with respect to the global frame, ${}^G F$, can be obtained, without the use of onboard global positioning sensors (GPS), via a wireless tether of mobile sensors as proposed in our previous work [11]. A team of mobile sensors are positioned to indirectly connect the landmark robots to a static home base, ${}^G H$, at the swarm’s starting location. Using the inter-robot measurements from the mobile sensors, landmark robots, and static home base, an estimate of the landmark robots’ positions, ${}^G \hat{L}_k$, can be obtained through the swarm localization approach proposed in our previous work [49].

An estimate of a mapper robot’s position needs to be obtained when incorporating a distance measurement, $z_{R_i}^t$, into the map. In this regard, once a mapper robot makes a distance measurement, it broadcasts a data packet to be received by the landmark robots. Thereafter, this data packet is relayed to the central controller.

The data packet contains the mapper robot’s distance measurement, $z_{R_i}^t$, as well as inter-robot distance and bearing measurements to any landmark robots within sensing range, $s_{R_i L_j}$. Additionally, upon receiving the data packet, the landmark robots append their own inter-robot distance and bearing measurements to the mapper robot, $s_{L_j R_i}$, prior to relaying the packet to the central controller. The k th data packet to arrive at the central controller is represented as $D_k = \{s_{R_i L_j}, s_{L_j R_i}, z_{R_i}^t, i\}$ and can be used to estimate the position of (mapper) Robot R_i , when it obtains the distance reading $z_{R_i}^t$.

Fusing the distance and bearing sensing data between the mapper robots and landmark robots, to obtain an estimate of the mapper robots’ positions, follows the approach developed in our previous work [10], and can be summarized as follows. Using both sets

of inter-robot sensing data from a data packet received by the central controller, an estimate of mapper Robot R_i 's location with respect to the local frame of the landmark robots can be determined. Since estimates of the positions of the landmark robots with respect to the global frame have already been obtained, ${}^G\hat{\mathbf{L}}$, the transformation from the local frame of the landmark robots to the global frame can be obtained and used to determine the estimated position of Robot R_i with respect to the global frame, ${}^G\hat{\mathbf{x}}_{R_i}$. Once an estimate of the mapper robot's position, where it obtained a distance sensor measurement, is determined, it can be incorporated into the occupancy grid map M .

4.1.3. Updating the Occupancy Grid Map

The occupancy grid map, M , is stored and updated by the central controller using measurements obtained from the mapper robots. The map is a 2D matrix of cells with each cell at (row, column) index (i, j) having an associated occupancy probability $p_{i,j}$, in the range $(0, 1)$. Prior to the incorporation of any data, each cell within the initial map, $M = M_0$, has a probability of $p_0 = 0.5$. Based on the occupancy probability, the true state of cell can be determined through the following rule: if $p_{i,j} > p_0$, the cell is occupied; if $p_{i,j} < p_0$ the cell is free; and, if $p_{i,j} = p_0$, the cell is unknown.

The map is updated continuously as data packets are obtained from the swarm using a standard map-update method [63]. The n th packet of data containing a mapper Robot R_i 's distance measurement, $z_{R_i}^t$, and used to obtain the estimated position at the time of obtaining this distance measurement, ${}^G\hat{\mathbf{x}}_{R_i}$, is incorporated into the previous map, M_{n-1} :

$$M_n = M_{n-1} + ISM\left(z_{R_i}, {}^G\hat{\mathbf{x}}_{R_i}\right) - M_0, \quad (9)$$

where the function $ISM(z_{R_i}, {}^G\hat{\mathbf{x}}_{R_i})$ represents the inverse sensor model that converts the robots distance measurement into a grid representation of occupancy.

As more data is incorporated into the map, the area within the map surrounding the landmark robots becomes better "known". However, since the motion of the mapper robots is random, the landmark robots must wait for an a priori unknown amount of time for them to gather sufficient data to map the entirety of their exploration area. A stopping criterion for when enough data has been collected to warrant moving the landmark robots to a new area can be defined based on how much of the expected area that can be mapped has been mapped.

The exploration area of the mapper robots is estimated simply as the area within line-of-sight and within radius $\rho_{explore}$ of the landmark robots. From the map, M , and using the estimated positions of the landmark robots ${}^G\hat{\mathbf{L}}_k$, within the map, the total exploration area of the mapper robots can be calculated, $m_{explore} \subset M$, assuming only unknown area within line-of-sight of the landmark robots is traversable. Any obstacles that would block line-of-sight from the landmark robots reduce the size of the exploration area accordingly. Furthermore, the subset of the total exploration area that has been mapped can be determined, $m_{known} \subset m_{explore}$. The stopping criterion can, then, be formulated as:

$$\frac{|m_{known}|}{|m_{explore}|} \geq w_s, \quad (10)$$

where w_s is a positive constant in the range $[0, 1]$ representing how much of the normalized exploration area should be mapped prior to moving the landmark robots to a new area. For example, in the case where $w_s = 1$, the entirety of the estimated exploration area must be mapped prior to moving the landmark robots. It should be noted that the known and explorable areas, m_{known} and $m_{explore}$, are updated as soon as new data packets are incorporated into the map. This allows the explorable area to remain accurate as new obstacles are observed since the calculation of the explorable area only considers area within the line-of-sight of the landmark robots. Figure 3 provides an illustrative procedure for incorporating measurements from the mapper robots into the map.

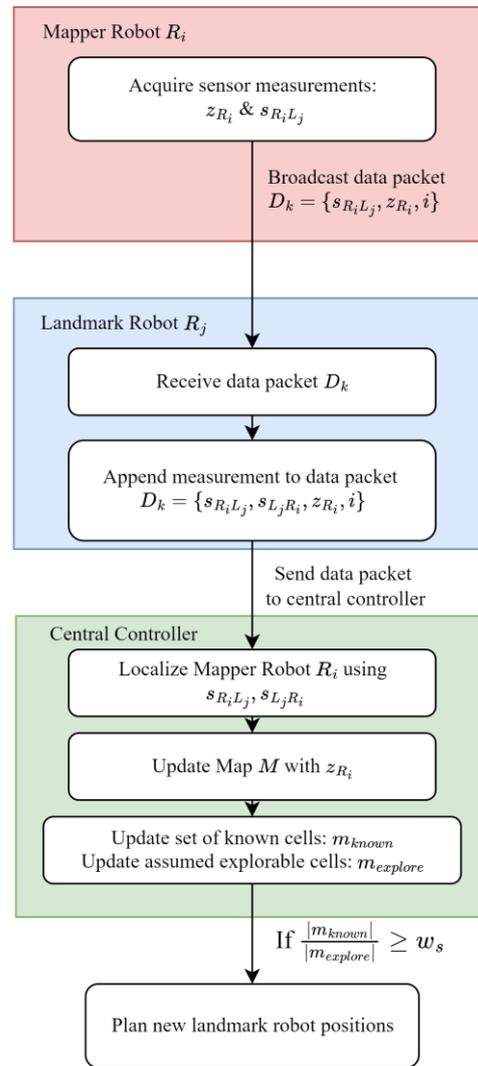


Figure 3. Flowchart for incorporating mapper robot distance measurement z_{R_i} into the map M .

4.2. Landmark-Robot Motion Planning

Once the local exploration of the environment nearby the landmark robots has been completed, the landmark robots should move towards a new area of the environment that has yet to be mapped. Planning a set of destination nodes for the landmark robots is outlined in Section 4.2.1 below. Motion planning and execution of the planned path for the landmark robots is, in turn, are outlined in Section 4.2.2 below.

4.2.1. Position Planning

Landmark robots should be positioned such that the mapper robots can access previously unexplored areas and, therefore, gather data to expand the known area of the map. Before a new set of landmark robot destination nodes can be planned, distinct frontier regions in the map should be found to determine where unexplored areas can be accessed and where configurations should be planned.

The set of frontier cells within the map need to be first determined. A frontier cell, f , is any free cell adjacent to an unknown cell in the map. A frontier region, F^M , represents a set of m frontier cells within the map that are adjacent to each other [64]:

$$F^M = \{f_1, f_2, \dots, f_m\}. \tag{11}$$

Figure 4 provides an illustration of distinct frontier regions found in an occupancy grid map.

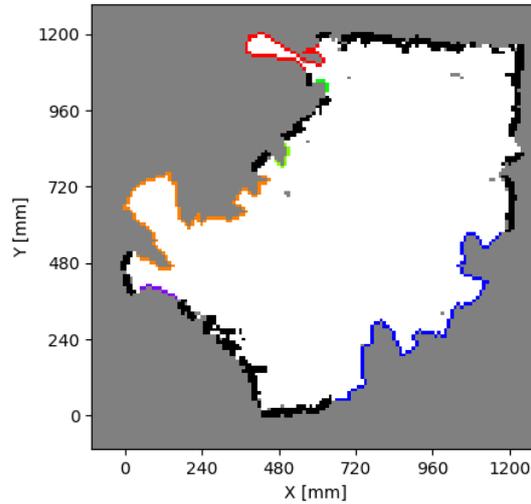


Figure 4. Occupancy grid map with distinct frontier regions shown in different colors. Each frontier region, F^M , represents a set of connected, freely traversable cells in the map.

A set of planned landmark robot positions, q , with $n \leq n_L$ landmark robots, can be defined as a matrix of (x, y) nodes that correspond to planned positions for each landmark robot. These nodes can also be defined with respect to the occupancy grid map as a matrix of [row, column] nodes:

$$q(n) = \begin{bmatrix} q_1 \\ q_2 \\ \dots \\ q_n \end{bmatrix} = \begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ \dots & \dots \\ x_n & y_n \end{bmatrix}; x, y \in \mathbb{R} \Rightarrow q_{grid} = \begin{bmatrix} r_1 & c_1 \\ r_2 & c_2 \\ \dots & \dots \\ r_n & c_n \end{bmatrix}; r, c \in \mathbb{Z}^+. \quad (12)$$

The objective of planning landmark-robot positions is to allow the mapper robots to explore and gather data from as large an unknown area as possible, such that the occupancy grid map can be expanded as much as possible. It is, therefore, useful to define a measure for the expected amount that the map will expand by.

Given some potential landmark-robot nodes, one can define the amount of new information that could be added by mapping the surrounding area as the *information gain*, $I(q)$. The mapping of the surrounding area is performed by the mapper robots, which only gather data from within a limited range of the landmark robots, $\rho_{explore}$. The information gain can therefore be estimated simply as the area of unknown occupancy grid cells within $\rho_{explore}$ and line-of-sight of each planned node since it is this area that would be newly added to map. Figure 5 below illustrates how the information gain is estimated given a potential set of landmark robot nodes. In this example $I(q) = 0.0992 \text{ m}^2$. The information gain is used to evaluate the performance of a particular set of nodes and is maximized for each set of nodes planned at a frontier region.

Optimally planning the position of every landmark robot is not always necessary. It may be sufficient to optimally plan only a subset of nodes if these nodes still allow the frontier region to be fully explored. A frontier region may be fully explored by fewer than the total number of landmark robots, especially, if the region is sufficiently small or there are many landmark robots available. If there are many landmark robots, optimally planning the position of every robot directly may become computationally infeasible. Therefore, the minimum number of landmark robots to explore a frontier region as well as their corresponding optimal positions need to be determined.

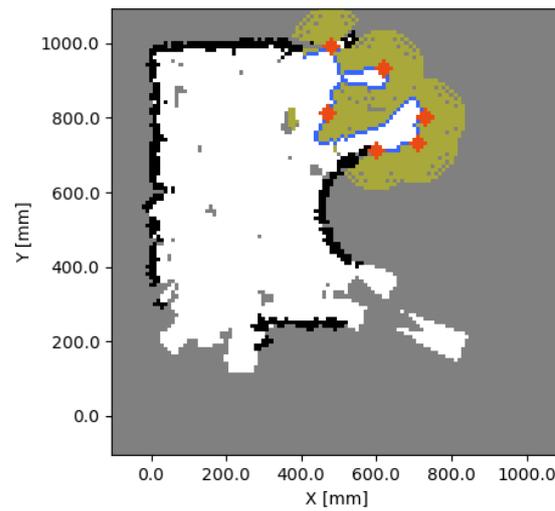


Figure 5. An occupancy grid map (free space, white, obstacles, black, unknown area, grey) with a frontier highlighted in blue and planned landmark-robot nodes overlaid as red circles. The yellow area corresponds to the estimated information gain, $I(q)$, and is the area of unknown cells within exploration range of the nodes.

The proposed algorithm for finding the optimal landmark-robot nodes, at a given frontier region is a nested-loop procedure, Figure 6: The outer loop (red dashed line) seeks to find the optimal number of landmark-robot nodes by evaluating configurations with a feasible number of nodes (i.e., number of nodes no greater than the total number of landmark robots, n_L); and, the inner loop (blue dashed line) determines the optimal placements of the landmark-robot nodes for the number of nodes considered by the outer loop.

In this work, landmark robots explore one frontier region at a time. Once all sets of nodes are planned at each frontier region in the map, one of these sets is chosen for the landmark robots to move towards the next region, ${}^G L_{k+1}$. Amongst the sets of nodes for each frontier region, let the set with the highest information gain be denoted as q_{\max} . This set is selected as the planned set of nodes for the landmark robots to move towards the next set, $q_{\max} \rightarrow {}^G L_{k+1}$.

Additional nodes may be added to the selected set of nodes q_{\max} , until the total number of nodes equals the total number of available landmark robots, n_L . These additional nodes are added at the centroid of the selected set to ensure connectivity of the landmark robots. Motion from the landmark robots' current estimated positions ${}^G \hat{L}_k$ to the planned set of nodes, ${}^G L_{k+1}$, needs to be planned, as to be detailed in Section 4.2.2.

Outer Loop

Given a frontier region consisting of a cluster of frontier points, the outer optimization loop seeks to find the minimum number of landmark-robot nodes, $n \leq n_L$, that maximizes the information gain. For a (feasible) number of landmark-robot nodes considered in this loop, the information gain is determined by finding the corresponding optimal node placement $q^*(n)$ using the inner optimization loop described below in inner loop section.

As detailed in Appendix A, when evaluating the information gain between two sets of landmark-robot nodes, with different number of nodes, the set with more nodes would have the larger information gain. However, as the number of nodes is increased, the area within mapping range of the nodes covers more of the frontier region which may become saturated if there are enough nodes to cover most of the unknown area surrounding the frontier. Since the size of the frontier region is finite, the unknown area within mapping range along the frontier region is also finite. A set of landmark-robot nodes would have an information gain that corresponds to some subset of this finite unknown area. As the number of nodes is increased and the frontier region becomes saturated with landmark

robots, the information gain approaches a maximum value corresponding to the total unknown area within mapping range of the frontier region.

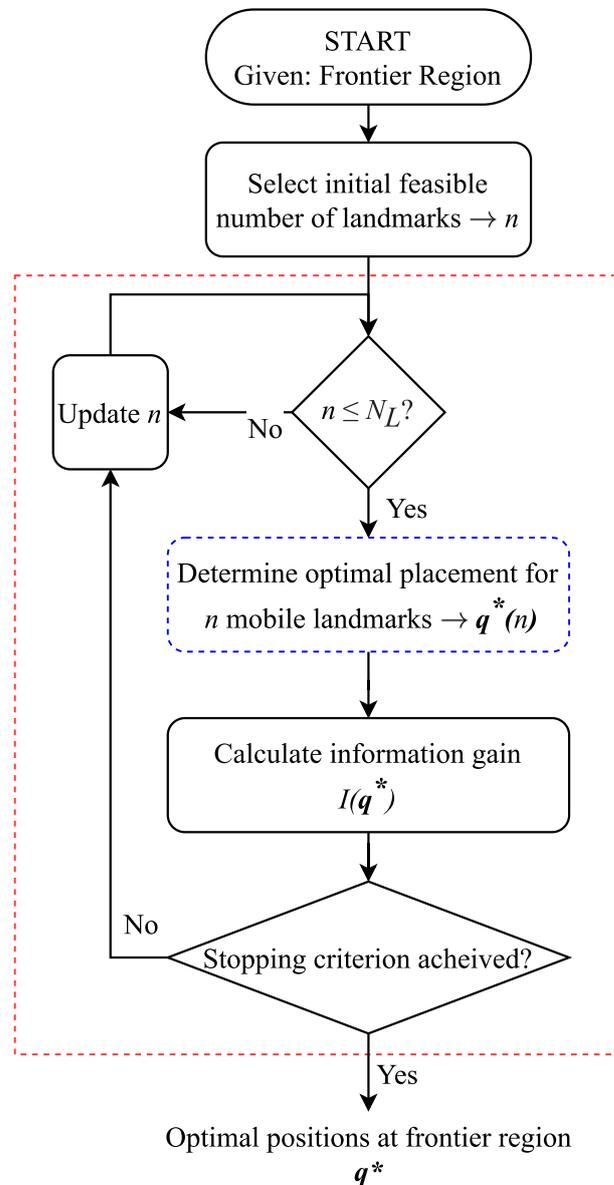


Figure 6. Flowchart of planning landmark-robot nodes at a given frontier region.

More formally, a frontier region can be defined as “saturated” when an optimally placed set of n nodes are planned such that the information gain of the optimal set of nodes, $I(q^*(n))$, surpasses a threshold, $w_i = [0, 1]$, of the maximum information gain possible I_{max} . Appendix A provides a more detailed description of determining the maximum information gain, I_{max} :

$$I(q^*(n)) \geq w_i * I_{max}. \tag{13}$$

The threshold w_i can be tuned to favour sets of nodes with more information gain at the expense of more landmark robots being used. This optimization, then, seeks to determine the minimum number of landmark-robot nodes to saturate the given frontier cluster. The optimal number of landmark-robot nodes, n^* , is, then:

$$n^* = \min(n) \text{ s.t. } I(q^*(n)) \geq w_i * I_{max} \tag{14}$$

$$n \in \mathbb{Z}^+,$$

where n is constrained by the total number of available robots $n \leq n_L$.

Determining the optimal number of landmark robots could be conducted using a simple, single-variable search engine that searches through the discrete set of integers from 1 up to the total available robots n_L . The search is terminated prior to reaching n_L , if the number of landmark robots saturates the frontier region.

Inner Loop

Given a frontier region, $F^M = \{f_1, f_2, \dots, f_m\}$, and the number of landmark-robot nodes to be planned, n , one would need to determine the set of node positions, $q^*(n)$, that maximizes the information gain. However, due to the limited sensing range of the landmark robots, and therefore limited exploration range of the mapping robots, constraints must be imposed on the planned nodes. The constraints imposed on a set of n nodes, $q(n)$, include the nodes being connected (as defined in Section 3.2.1) and the positions of all the n nodes being in free space nearby the frontier F^M .

The inner-loop optimization solves for:

$$q^*(n) = \max(I(q(n))), \quad (15)$$

where the set of nodes $q(n) = [q_1, q_2, \dots, q_n]^T$ is subject to the following constraints:

Connectivity constraint :

$$\{dist(q_i, q_{i+1})\}_{i=1}^{n-1} \leq \rho_{sense}, \quad (16)$$

Nearby-frontier constraint :

$$\{dist(q_i, F^M)\}_{i=1}^n \leq \rho_{explore}, \quad (17)$$

Free-space constraint :

$$q_{grid} \in M_{free}. \quad (18)$$

Above, the distance between a node, q_i , and set of frontier points, F , (i.e., $dist(q_i, F)$) is the minimum Euclidean distance between the node and the set of frontier points. Additionally in the free-space constraint above, M_{free} represents the subset of points in the occupancy grid that are known to be free (i.e., not unknown and contain no obstacles).

The nearby-frontier and free-space constraints (Equations (17) and (18), respectively) limit the placement of all the nodes to the same subset of occupancy grid cells. This subset of cells can be determined prior to beginning the optimization and can be used to specify the domain of each variable (node) to a discrete set. The connectivity constraint represents a non-linear constraint between each pair of nodes so that the landmark robots remain within inter-robot sensing range of each other.

This inner optimization loop represents a nonlinearly constrained discrete optimization problem. It is, therefore, recommended to use a combinatoric method, such as a variation of a genetic algorithm to obtain an optimal solution [65].

In this context, one method of obtaining an initial high-quality solution is to build a set of nodes sequentially, solving a single-variable discrete optimization problem consecutively until the number of required nodes is reached. Determining the position of each node involves solving a single-variable optimization problem for the occupancy grid cell with the highest information gain. Initially, the first node can be placed at the single location with the highest information gain that satisfies Equations (17) and (18). The next nodes are added by assuming all previous nodes remain fixed and, therefore, reducing the domain of the search for the next node to cells that satisfy all three above constraints, Equations (16)–(18). Figure 7 illustrates this method of finding an initial high-quality solution.

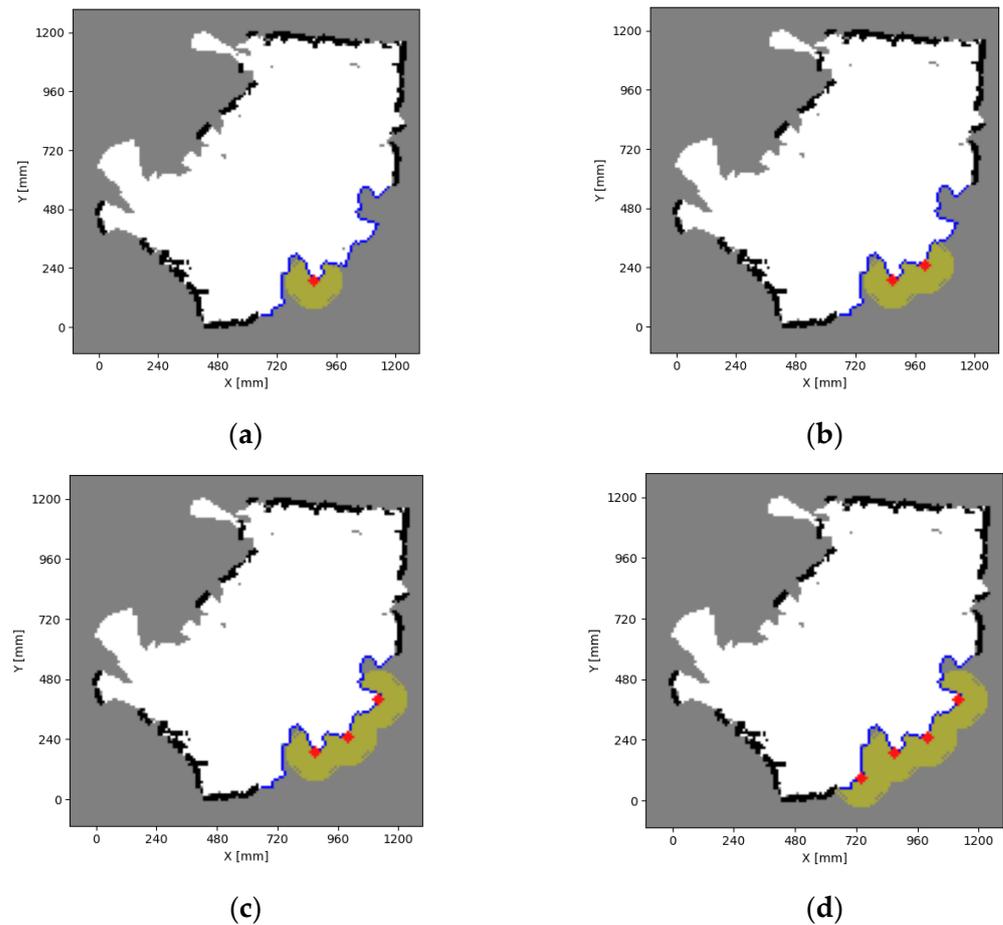


Figure 7. An initial high-quality solution, when maximizing the information gain $I(q(n))$ of a set of planned nodes at a frontier region, can be obtained by planning nodes sequentially. (a–d) depict an occupancy grid map with nodes (red) planned sequentially at a frontier region (blue) with the maximized unknown area surrounding each node (yellow).

4.2.2. Path Planning

Once a set of nodes corresponding to destinations for the landmark robots has been determined, obstacle-free paths must be found between the landmark robots' current positions and destination nodes. Additionally, the landmark robots should remain connected to each other during their motion.

An initial obstacle-free path, between the current centroid of the landmark-robot positions and centroid of the planned nodes, can be determined using an A* planner [66]. Motion commands can, then, be supplied to the landmark robots to direct them from their current position, along the planned path, toward their destination node. By having all the landmark robot use the same planned paths, rather than planning an individual path for each robot, the landmark robots remain nearby each other during their motion (i.e., connected).

5. Results

Simulated examples are included herein to illustrate the working of the proposed mapping strategy in building a 2D occupancy grid map of an unknown environment using robotic swarms. The proximity sensing and motion characteristics of our mROBerTO millirobot were used as the robots in the simulations [6–8]. Namely, all simulations were completed with inter-robot sensing noise variance of $\sigma_{int}^2 = 2 \text{ mm}^2$ with a maximum range $\rho_{sense} = 200 \text{ mm}$. Additionally, each mapper robot was equipped with a simulated forward-facing 1D distance (laser) sensor on the robot. The simulated distance sensor had a max

range of $z_{\max} = 200$ mm, and noise of $\sigma_{\text{laser}}^2 = 9$ mm² and measured the shortest distance to an obstacle within a field-of-view of $\beta = 22^\circ$.

In the simulations, the landmark robots had a constant speed of $v_{\text{landmark}} = 60$ mm/s, while the mapper robots had a faster constant speed of $v_{\text{mapper}} = 80$ mm/s in order to ensure they remained connected to the landmark robots as the swarm moved to new areas of the environment. The occupancy grid map had an individual cell size of 10 mm. The parameter, w_s , was set to 0.95 and described what percentage of the explorable area surrounding the landmark robots should be mapped prior to moving the landmark robots. The parameter, w_i , was set to 0.85 and described what percentage of the maximum information gain at a frontier should a potential set of landmark nodes have before considering the set of nodes to allow for most of the frontier to be explored (see Appendix A).

5.1. Illustrative Example—1200 × 1200 mm² Enclosed Environment

For the first simulation, discussed herein, the swarm comprised of four landmark robots and ten mapper robots and explored a 1200 × 1200 mm² enclosed environment, Figure 8. A fixed home base reference was included in the environment, located at (650, 450) mm, needed for the landmark robots to use a tether-based motion strategy to move to new areas.

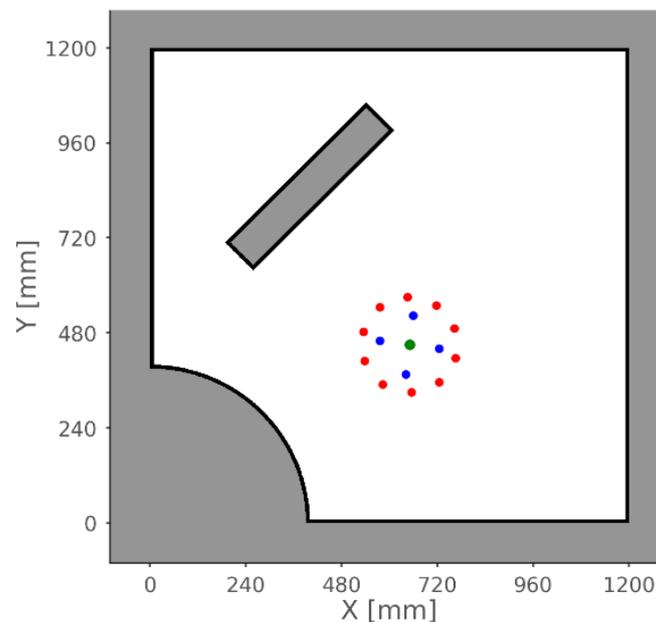


Figure 8. True Environment with initial positions of landmark robots shown as *blue* circles, and mapper robots shown as *red* circles. The home base is shown in *green*.

The proposed exploration strategy begins by using the decentralized mapper robots to gather distance measurements around the local area surrounding the landmark robots. Once 95% of the area within exploration range of the landmark robots has been mapped ($w_s = 0.95$), the landmark robots move to the next area of the map that maximizes the amount of new information potentially added to the map, and the process repeats. This iterative procedure repeats until no more frontiers can be found in the map.

Figure 9 shows the occupancy grid maps after each iteration for a total of nine iterations. The final map is shown in Figure 9i. Figure 10 depicts the percentage of the environment mapped over time. The marked points in Figure 10 correspond to the percentage of the environment mapped after each iteration. The percentage mapped between marked points (iterations) initially increases rapidly as the local area surrounding the landmark robots is mapped. Prior to the next iteration, however, the curve flattens since during this time the landmark robots move to a new area in the environment.

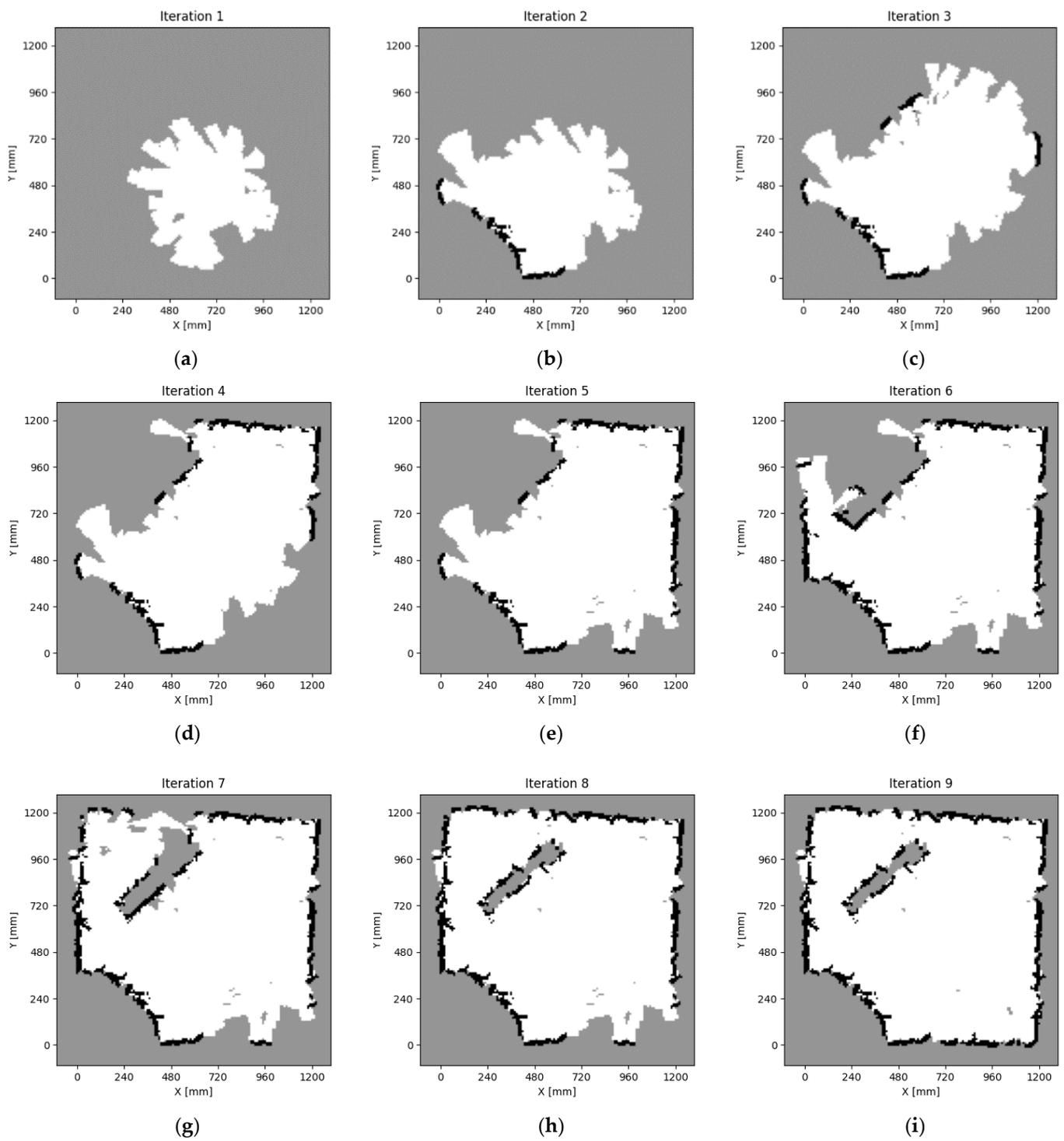


Figure 9. Occupancy grid maps after each iteration of the proposed exploration strategy. (a–i) depict the map after iterations one to nine respectively.

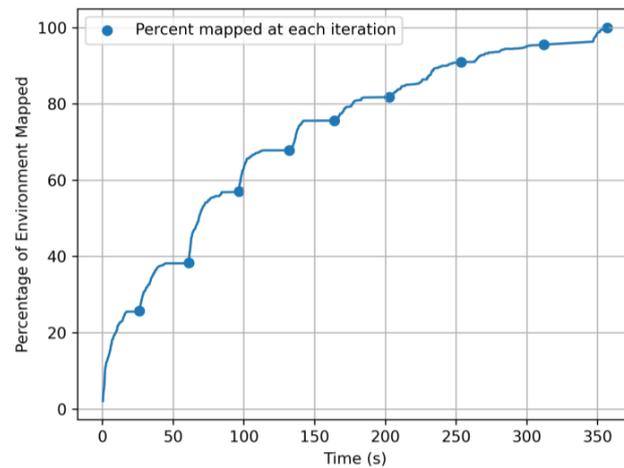


Figure 10. Percentage of the environment (shown in Figure 8) mapped over time. The percentage after each iteration is additionally shown as circular markers.

5.2. $1000 \times 1000 \text{ mm}^2$ Non-Enclosed Environment

In this simulation, the swarm comprised four landmark robots and ten mapper robots and explored a $1000 \times 1000 \text{ mm}^2$ non-enclosed environment, Figure 11a. A fixed home base reference was included in the environment, located at (300, 400) mm, needed for the landmark robots to use a tether-based motion strategy to move to new areas.

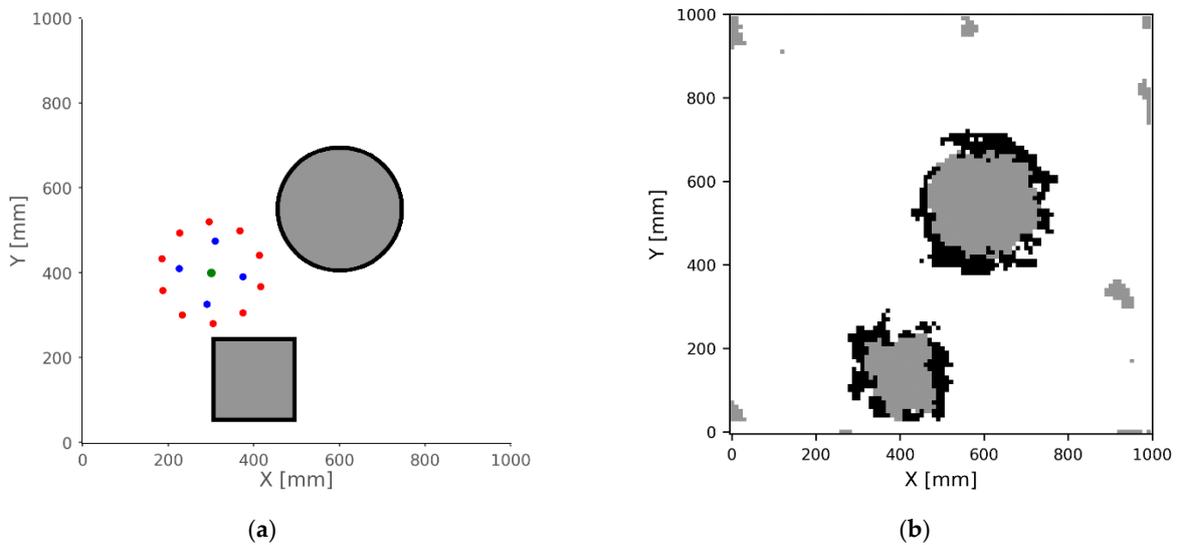


Figure 11. $1000 \times 1000 \text{ mm}^2$ non-enclosed environment: (a) True Environment with initial positions of landmark robots shown as blue circles, and mapper robots shown as red circles. The home base is shown in green. (b) Final occupancy grid map.

The environment did not have walls to prevent the swarm from moving outside of the desired area to be mapped. The area outside of the true environment shown in Figure 11a was freely traversable by the swarm. In order to prevent the swarm from trying to explore regions outside of the $1000 \times 1000 \text{ mm}^2$ environment, the occupancy grid was initialized to represent only a $1000 \times 1000 \text{ mm}^2$ region prior to exploration. Frontiers and, therefore, planned positions of the landmark robots were always discovered inside the desired area to be mapped. Figure 11b shows the final map obtained through our proposed exploration strategy.

Despite the environment not having walls, the landmark robots were always planned by the central controller to stay inside the $1000 \times 1000 \text{ mm}^2$ occupancy grid map. Since the

mapper robots were constrained to stay nearby the landmark robots, none of the robots in the swarm could travel too far outside the area to be mapped.

5.3. Impact of Relative Number of Landmark Robots

In order to assess the impact of the number of landmark robots, n_L , on the exploration and mapping speed, we conducted a series of simulations with a fixed total number of robots in the swarm. We assigned a subset of the swarm as landmark robots, while the remaining members were designated as mapper robots. Varying the number of landmark robots, n_L , we determined the time required to complete the mapping of the environment and the total number of iterations needed for each simulation.

The objective of this study was to identify the relative ratio of landmark robots to total number of robots in the swarm that can achieve the fastest mapping of the environment with the fewest iterations possible. One may recall that, in our exploration strategy, each iteration involves locally exploring the area surrounding the landmark robots and, then, planning and moving the landmarks to a new frontier in the environment. It is, therefore, desirable to minimize the number of iterations in order to minimize the frequency at which the landmark robots' motions need to be planned.

The time to complete mapping the environment and the number of iterations were each normalized and a weighted sum objective function, that combined the values from both goals, was utilized:

$$H(n_L) = w_{time} * T_{norm} + w_{iter} * I_{norm}, \tag{19}$$

where w_{time} and w_{iter} represent the weights associated with the normalized completion time, T_{norm} , and normalized number of iterations, I_{norm} , respectively.

Figures 12 and 13 below depict the normalized completion time and total number of iterations, as well as the objective function using weights of $w_{time} = w_{iter} = 1$ for simulations conducted in the environment shown in Sections 5.1 and 5.2, respectively. As can be noted in Figures 12 and 13, as the number of landmarks is increased, the overall time taken to map an environment also increased, while the total number of iterations required decreased. When the time to complete mapping the environment and total number of iterations were weighted equally, the number of landmark robots that minimized the heuristic objective function $H(n_L)$ is approximately half the total number of robots in the swarm. Under these conditions, the environment is mapped using fewer iterations.

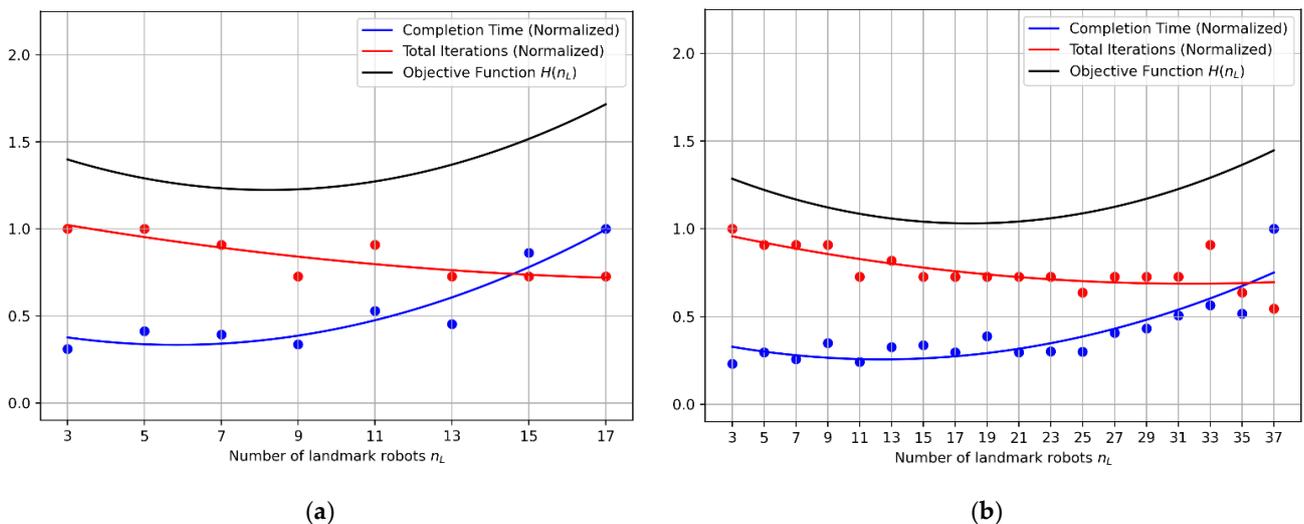


Figure 12. Results from simulations in the enclosed environment from Section 5.1 for (a) $N = 20$, and (b) $N = 40$ robots, respectively. Objective function $H(n_L)$ calculated using $w_{time} = 1$, $w_{iter} = 1$ and a second-degree polynomial fit of normalized values.

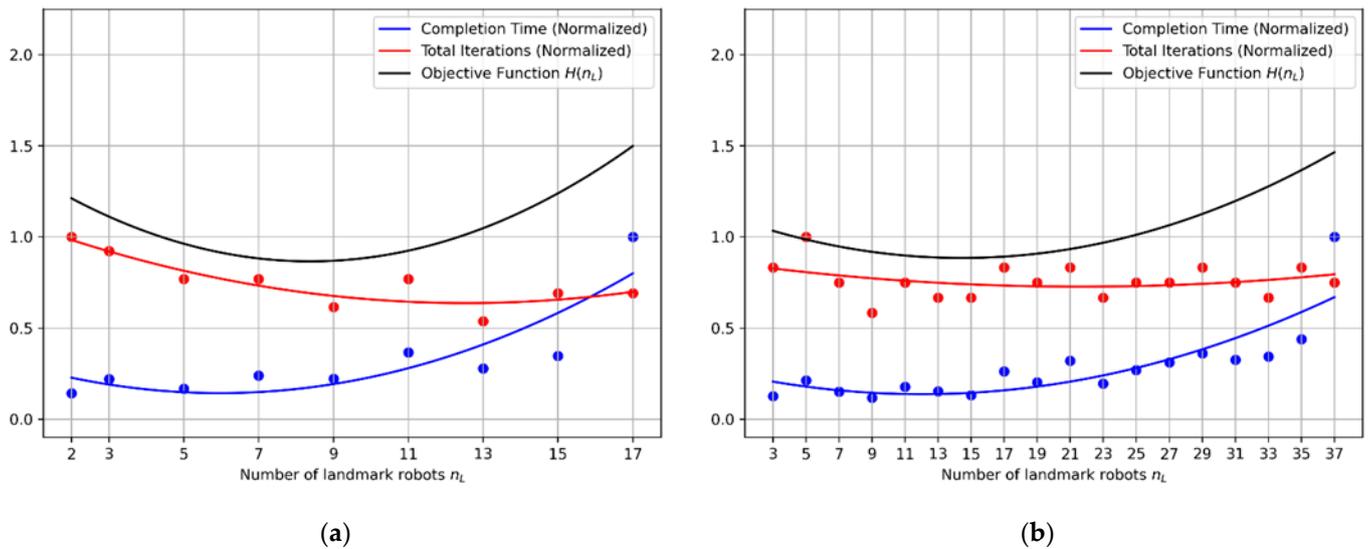


Figure 13. Results from simulations in the unbounded environment from Section 5.2 for (a) $N = 20$, and (b) $N = 40$ robots, respectively. Objective function $H(n_L)$ calculated using $w_{time} = 1$, $w_{iter} = 1$ and a second-degree polynomial fit of normalized values.

In order to map the environment even faster, more weight can be given to the completion time. In this case, approximately one third of the total number of robots in the swarm can be given the role of landmark robot while the rest are given the role of mapper robot. Conversely, when more than half of the robots are landmark robots, the least number of iterations would be required to complete mapping the environment, at the expense of an increased overall time to map.

During each iteration every landmark robot's motion must be planned by a central controller and sent to the robot in the form of motion commands to be executed. Depending on the specific swarm robots utilized, it may be worth the increase in the time to map the environment, if the landmark robots' motions need to be planned and executed less frequently.

5.4. Comparison with Random Landmark Placement

Our strategy optimizes the positions of the landmark robots to maximize new information being added to the occupancy grid map. In order to validate the effectiveness of this optimization, a comparative study, with an alternate method of planning the positions of the landmark robots, was conducted. This alternate method, hereafter, denoted the random-placement method, places landmark robots (randomly) nearby a randomly selected frontier region in the environment. It should be noted that the overall structure of the exploration strategy, namely, using mapper robots to first locally map an area surrounding the landmark robots and, then, moving the landmark robots to a new area in the environment, remains the same.

The random-placement method, first, uniformly selects a frontier region from the set of frontier regions found in the map. Next, a node, representing the planned position of one landmark robot, is randomly selected from the set of occupancy grid cells satisfying the *nearby-frontier* and *free-space* constraints (Equations (17) and (18), respectively). This ensures that the frontier region can at least be partially explored by the mapper robots. Subsequent nodes are randomly selected from the set of cells satisfying the previously mentioned constraints as well as the *connectivity* constraint given in Equation (15) until the total number of nodes equal the total number of landmark robots available. The random-placement method of positioning the landmark robots satisfies all of the same constraints as our proposed optimally-planned-placement method presented in Section 4.2.1.

Figure 14a,b below depict the results from exploring the environments shown in Sections 5.1 and 5.2, respectively, using both the optimal planned positions of the landmark robots and the random placement method. The swarm comprised 20 robots in total, of which $n_L = 10$ were landmark robots. As can be noted in Figure 14, optimally planning the positions of the landmark robots allows for an overall tangibly faster exploration and mapping of the environment.

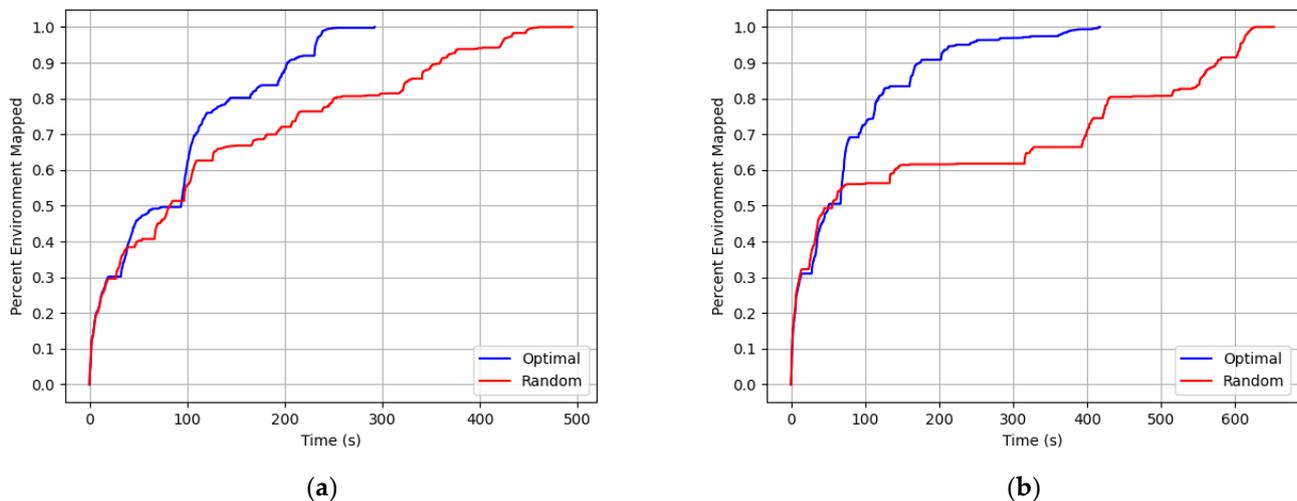


Figure 14. Validating the effectiveness of the optimal placement method (*red*) when planning the position of the landmark robots by comparing with the random placement method (*blue*) for: (a) the enclosed environment shown in Section 5.1, and (b) the unbounded environment shown in Section 5.2.

6. Conclusions

This paper presents a novel collaborative exploration strategy that allows a resource-constrained swarm to build an occupancy grid map of an unknown environment. The proposed strategy divides the swarm into two teams, mapper robots and landmark robots. The mapper robots map the local environment surrounding the landmark robots. The landmark robots direct the exploration and mapping of the swarm toward a particular area of the environment by physically positioning themselves in that area. The novelty of our strategy is in the physical decoupling of a swarm into these two teams: landmark and mapper robots, respectively. This decoupling allows the problem of directing a swarm to a new area of the environment to be addressed separately from the problem of exploring that new area using robots with limited environment perception abilities. Namely, the landmark robots' motions are planned and executed to maximize new information added to the map without impeding how the mapper robots explore the area. The mappers robots are, therefore, can still use a decentralized, scalable, random-motion model to explore the local area surrounding the landmark robots while the swarm, as a whole, follows an efficient frontier-based exploration of the environment.

In contrast, other methods, for building occupancy grid maps using resource-constrained swarms, do not direct exploration. Instead, they use random motion alone to disperse the swarm and, therefore, may explore the environment inefficiently by allowing the robots to visit previously mapped areas [26–28].

The positions of the landmark robots are optimized herein using a frontier-based technique to maximize new information added to the map while also adhering to connectivity constraints. These connectivity constraints ensure the landmark robots can communicate and share relative position estimates between themselves and nearby mapper robots. By ensuring connectivity between the landmark robots, their positions can be accurately estimated using the tether-based strategy presented in our previous work [11]. The landmark robots do not need to rely on external sensing infrastructure to provide them with absolute position estimates.

The proposed collaborative exploration strategy was evaluated through a series of simulated experiments. These validated the effectiveness of the proposed method in building an accurate occupancy grid map. Future work may consider the use of alternative random-motion models for the mapper robots as well as investigating the creation of global maps of an environment using map merging techniques from local maps generated solely from exploration of the nearby area surrounding landmark robots.

Author Contributions: Individual contributions from the authors of this research paper are as follows: Conceptualization, A.R. and B.B.; methodology, A.R.; software, A.R. and K.E.; validation, A.R.; formal analysis, A.R.; investigation, A.R.; resources, G.N. and B.B.; data curation, A.R.; writing—original draft preparation, A.R.; writing—review and editing, A.R., K.E., G.N. and B.B.; visualization, A.R.; supervision, G.N. and B.B.; project administration, A.R. and B.B.; funding acquisition, G.N. and B.B. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) and the Canada Research Chairs Program.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Nomenclature

Symbol	Description
${}^G F$	Fixed global reference frame
${}^G H$	Location of home base with respect to global reference frame
n_L	Number of landmark robots
n_m	Number of mapper robots
R_i	Label of mapper robot $i \in \{1, \dots, n_m\}$
L_j	Label of landmark robot $j \in \{1, \dots, n_L\}$
${}^G L_k$	Set of planned landmark positions at k th iteration
${}^G x_{L_i}$	Planned position of landmark robot L_i
${}^G \hat{L}_k$	Estimated landmark positions at k th iteration
${}^G \hat{x}_{L_i}$	Estimated position of landmark robot L_i
M	Occupancy grid map
p_{ij}	Probability of occupancy of cell in map M located at i -th row and j -th column
F^M	Distinct frontier region in map
q	Arbitrary set of landmark robot nodes
$I(q)$	Information gain of set of nodes q
ρ_{sense}	Maximum distance between robots for inter-robot sensing and communication
$\rho_{explore}$	Radius of maximum allowable distance away from the landmark robots for the mapper robots to travel
$s_{R_i R_j}$	Inter-robot sensor measurement of distance and bearing to Robot R_j as measured by Robot R_i
$S_{R_i}^t$	Set of inter-robot sensor measurements taken by Robot R_i of all robots within ρ_{sense}
$z_{R_i}^t$	Exteroceptive sensor measurement of the environment (i.e., measurement from laser distance sensor) taken by Robot R_i at time t
$D_{R_i}^t$	Data packet of sensor measurements sent by Robot R_i at time t
w_s	Pre-set threshold for what percent of the exploration area surrounding the landmark robots should be mapped before moving the landmarks
w_i	Pre-set threshold on normalized information gain to consider a frontier as fully explorable (Appendix A)

Appendix A. Maximum Information Gain

In this Appendix, we seek to estimate if a set of landmark robot nodes, q , planned at a frontier region F^M will allow for the exploration of most of the explorable unknown area surrounding the frontier region. Given a frontier region, corresponding to a finite set of frontier cells F^M , within an occupancy grid map, M , the total potentially explorable unknown region surrounding the frontier can be defined as the unknown region of the map

within exploration range of the frontier, i.e., a distance of $\rho_{explore}$ away from the frontier. A set of landmark robot nodes, q , planned at this frontier, F^M , will allow for the exploration of part of this total unknown region. One can calculate what fraction of the total unknown region will the set of nodes q allow to be explored. A minimum value of this fraction can, then, be set to define if q allows for the exploration of most of the unknown region surrounding the frontier.

In this context, when planning landmark robot nodes, one can estimate how much of this unknown area will be explored by computing the landmark robot nodes' information gain, $I(q)$. The information gain is computed simply as the unknown area of the map that is within line-of-sight and within a circle-of-radius, $\rho_{explore}$, of these nodes. Each frontier cell in the frontier region F^M , has a finite amount of unknown area surrounding it within a radius of $\rho_{explore}$. Additionally, since there is a finite number of frontier cells in F^M , there is a finite amount of unknown area surrounding the frontier region and, therefore, there is maximum potential information gain of any set of landmark robot nodes planned at this frontier region. Let the maximum information gain of the frontier region be denoted as $I_{max}^{F^M}$.

The maximum potential information gain can be obtained by finding the unknown area that is within exploration range of all points satisfying the nearby frontier constraint (Equation (17)) and free space constraint (Equation (18)). This is equivalent to finding the information gain of a set of nodes where each node is placed at every grid cell nearby the frontier region. Let this set of nodes be denoted as q' :

$$q' = \left\{ c \in M_{free} \text{ s.t. } dist(c, F^M) \leq \rho_{explore} \right\}, \quad (A1)$$

where M_{free} represents the subset of cells in the map M that are known to be free (i.e., not unknown and contain no obstacles) and the function $dist(c, F^M)$ represents the distance between a cell c and set of frontier points F^M (i.e., the minimum Euclidean distance between the cell and the set of frontier points). The maximum potential information gain can, then, be formulated as:

$$I_{max} = I(q'). \quad (A2)$$

Figure A1 illustrates the maximum potential information gain of different frontier regions in an occupancy grid map. The (colored) shaded regions correspond to the maximum unknown areas that are visible from any point nearby the respective frontier.

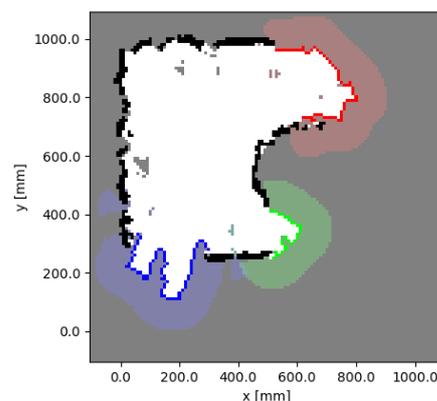


Figure A1. An occupancy grid map with frontier regions overlaid in different colours.

The maximum potential information gain can be used to determine if a given set of landmark robot nodes, q , allow for the exploration of most of the frontier region F^M .

A configuration information gain can first be normalized with respect to the maximum potential information gain I_{\max}^{FM} :

$$I_{norm}(q) = \frac{I(q)}{I_{\max}^{FM}}. \quad (A3)$$

A set of landmark-robot nodes can be said to allow for the exploration of most of the frontier region if its corresponding normalized information gain surpasses a threshold value w_i . The set of nodes q allows for most of F^M to be explored if:

$$I_{norm}(q) > w_i. \quad (A4)$$

Equation (A4) can be rearranged as follows by substituting in Equation (A3):

$$I(q) > w_i * I_{\max}^{FM} \quad (A5)$$

The normalized information gain values for the *red*, *green*, and *blue* frontier clusters, shown in Figure A1, are noted in Figure A2 versus the number of planned landmark robot nodes. An example threshold value of 0.85 was set and configurations with the minimum number of landmark robots to surpass this threshold are shown in the figures.

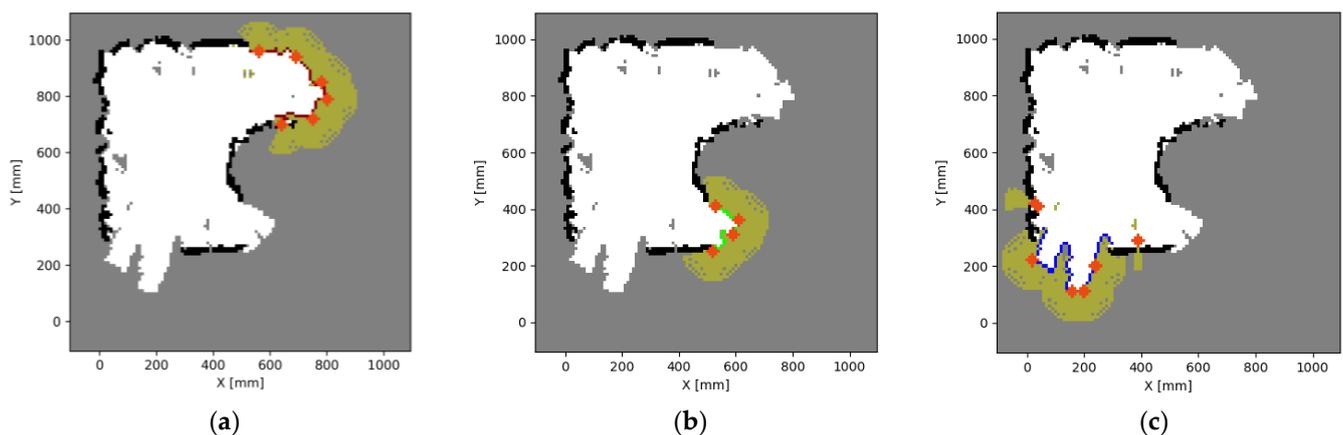


Figure A2. Sets of landmark robot nodes and information gain scores at each frontier region seen: (a) *Red* frontier region in Figure A1; (b) *Green* frontier region in Figure A1; (c) *Blue* frontier region in Figure A1.

References

1. Şahin, E. Swarm Robotics: From Sources of Inspiration to Domains of Application. In *Swarm Robotics*; Springer: Berlin/Heidelberg, Germany, 2005; pp. 10–20. [CrossRef]
2. Heinrich, M.K.; Soorati, M.D.; Kaiser, T.K.; Wahby, M.; Hamann, H. Swarm robotics: Robustness, scalability, and self-X features in industrial applications. *Inf. Technol.* **2019**, *61*, 159–167. [CrossRef]
3. Schranz, M.; Umlauf, M.; Sende, M.; Elmenreich, W. Swarm Robotic Behaviors and Current Applications. *Front. Robot. AI* **2020**, *7*, 36. Available online: <https://www.frontiersin.org/articles/10.3389/frobt.2020.00036> (accessed on 1 September 2022). [CrossRef] [PubMed]
4. Brambilla, M.; Ferrante, E.; Birattari, M.; Dorigo, M. Swarm robotics: A review from the swarm engineering perspective. *Swarm Intell.* **2013**, *7*, 1–41. [CrossRef]
5. Hamann, H. *Swarm Robotics: A Formal Approach*; Springer International Publishing: Cham, Switzerland, 2018; ISBN 978-3-319-74526-8.
6. Kim, J.Y.; Colaco, T.; Kashino, Z.; Nejat, G.; Benhabib, B. mROBerTO: A modular millirobot for swarm-behavior studies. In Proceedings of the 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Daejeon, Republic of Korea, 9–14 October 2016; pp. 2109–2114. [CrossRef]
7. Kim, J.Y.; Kashino, Z.; Colaco, T.; Nejat, G.; Benhabib, B. Design and implementation of a millirobot for swarm studies—mROBerTO. *Robotica* **2018**, *36*, 1591–1612. [CrossRef]
8. Eshaghi, K.; Li, Y.; Kashino, Z.; Nejat, G.; Benhabib, B. mROBerTO 2.0—An Autonomous Millirobot with Enhanced Locomotion for Swarm Robotics. *IEEE Robot. Autom. Lett.* **2020**, *5*, 962–969. [CrossRef]

9. Drisdelle, R.; Kashino, Z.; Pineros, L.; Kim, J.Y.; Nejat, G.; Benhabib, B. Motion Control of a Wheeled Millirobot. In Proceedings of the 4th International Conference of Control, Dynamic Systems, and Robotics (CDSR 2017), Toronto, ON, Canada, 21–23 August 2017.
10. Eshaghi, K.; Kashino, Z.; Yoon, H.J.; Nejat, G.; Benhabib, B. An inchworm-inspired motion strategy for robotic swarms. *Robotica* **2021**, *39*, 2283–2305. [[CrossRef](#)]
11. Eshaghi, K.; Rogers, A.; Nejat, G.; Benhabib, B. Closed-Loop Motion Control of Robotic Swarms—A Tether-Based Strategy. *IEEE Trans. Robot.* **2022**, *38*, 3564–3581. [[CrossRef](#)]
12. Chung, S.-J.; Paranjape, A.A.; Dames, P.; Shen, S.; Kumar, V. A Survey on Aerial Swarm Robotics. *IEEE Trans. Robot.* **2018**, *34*, 837–855. [[CrossRef](#)]
13. Abdelkader, M.; Güler, S.; Jaleel, H.; Shamma, J.S. Aerial Swarms: Recent Applications and Challenges. *Curr. Robot. Rep.* **2021**, *2*, 309–320. [[CrossRef](#)]
14. Wang, X.; Tan, G.; Dai, Y.; Lu, F.; Zhao, J. An Optimal Guidance Strategy for Moving-Target Interception by a Multirotor Unmanned Aerial Vehicle Swarm. *IEEE Access* **2020**, *8*, 121650–121664. [[CrossRef](#)]
15. Shi, G.; Hönig, W.; Yue, Y.; Chung, S.-J. Neural-Swarm: Decentralized Close-Proximity Multirotor Control Using Learned Interactions. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA 2020), Paris, France, 31 May 2020–31 August 2020; pp. 3241–3247. [[CrossRef](#)]
16. Shi, G.; Hönig, W.; Shi, X.; Yue, Y.; Chung, S.-J. Neural-Swarm2: Planning and Control of Heterogeneous Multirotor Swarms Using Learned Interactions. *IEEE Trans. Robot.* **2022**, *38*, 1063–1079. [[CrossRef](#)]
17. Hönig, W.; Preiss, J.A.; Kumar, T.K.S.; Sukhatme, G.S.; Ayanian, N. Trajectory Planning for Quadrotor Swarms. *IEEE Trans. Robot.* **2018**, *34*, 856–869. [[CrossRef](#)]
18. Vardy, A. Orbital Construction: Swarms of Simple Robots Building Enclosures. In Proceedings of the 2018 IEEE 3rd International Workshops on Foundations and Applications of Self* Systems (FAS*W), Trento, Italy, 3–7 September 2018; pp. 147–153. [[CrossRef](#)]
19. Eschke, C.; Heinrich, M.K.; Wahby, M.; Haman, H. Self-Organized Adaptive Paths in Multi-Robot Manufacturing: Reconfigurable and Pattern-Independent Fibre Deployment. In Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 3–8 November 2019; pp. 4086–4091. [[CrossRef](#)]
20. Innocente, M.S.; Grasso, P. Self-organising swarms of firefighting drones: Harnessing the power of collective intelligence in decentralised multi-robot systems. *J. Comput. Sci.* **2019**, *34*, 80–101. [[CrossRef](#)]
21. Neumann, P.P.; Hirschberger, P.; Baurzhan, Z.; Tiebe, C.; Hofmann, M.; Hüllmann, D.; Bartholmai, M. Indoor Air Quality Monitoring using flying Nanobots: Design and Experimental Study. In Proceedings of the 2019 IEEE International Symposium on Olfaction and Electronic Nose (ISOEN), Fukuoka, Japan, 26–29 May 2019; pp. 1–3. [[CrossRef](#)]
22. Macwan, A.; Nejat, G.; Benhabib, B. Target-Motion Prediction for Robotic Search and Rescue in Wilderness Environments. *IEEE Trans. Syst. Man Cybern. Part B Cybern.* **2011**, *41*, 1287–1298. [[CrossRef](#)]
23. Macwan, A.; Nejat, G.; Benhabib, B. Optimal deployment of robotic teams for autonomous wilderness search and rescue. In Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, San Francisco, CA, USA, 25–30 September 2011; pp. 4544–4549. [[CrossRef](#)]
24. Bakhtari, A.; Naish, M.D.; Eskandari, M.; Croft, E.A.; Benhabib, B. Active-vision-based multisensor surveillance—An implementation. *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.* **2006**, *36*, 668–680. [[CrossRef](#)]
25. Thrun, S. Robotic Mapping: A Survey. In *Exploring Artificial Intelligence in the New Millennium*; Lakemeyer, G., Nebel, B., Eds.; Morgan Kaufmann Publishers Inc.: San Francisco, CA, USA, 2003; pp. 1–35, ISBN 978-1-55860-811-5.
26. Chaves, R.; Rezeck, P.; Chaimowicz, L. SwarMap: Occupancy Grid Mapping with a Robotic Swarm. In Proceedings of the 2019 19th International Conference on Advanced Robotics (ICAR), Belo Horizonte, Brazil, 2–6 December 2019; pp. 727–732. [[CrossRef](#)]
27. Kegeleirs, M.; Garzón Ramos, D.; Birattari, M. Random Walk Exploration for Swarm Mapping. In *Towards Autonomous Robotic Systems*; Althoefer, K., Konstantinova, J., Zhang, K., Eds.; Springer International Publishing: Cham, Switzerland, 2019; pp. 211–222. [[CrossRef](#)]
28. Pires, A.G.; Rezeck, P.A.F.; Chaves, R.A.; Macharet, D.G.; Chaimowicz, L. Cooperative Localization and Mapping with Robotic Swarms. *J. Intell. Robot. Syst.* **2021**, *102*, 47. [[CrossRef](#)]
29. Thrun, S.; Bü, A. Integrating Grid-Based and Topological Maps for Mobile Robot Navigation. In Proceedings of the 13th National Conference on Artificial Intelligence—Volume 2, Portland, OR, USA, 4–8 August 1996; pp. 944–950.
30. Teymouri, M.S.; Bhattacharya, S. Landmark-based Distributed Topological Mapping and Navigation in GPS-denied Urban Environments Using Teams of Low-cost Robots. *arXiv* **2021**, arXiv:2103.03741. Available online: <http://arxiv.org/abs/2103.03741> (accessed on 10 May 2022).
31. Ramaithitima, R.; Whitzer, M.; Bhattacharya, S.; Kumar, V. Automated Creation of Topological Maps in Unknown Environments Using a Swarm of Resource-Constrained Robots. *IEEE Robot. Autom. Lett.* **2016**, *1*, 746–753. [[CrossRef](#)]
32. Ramachandran, R.K.; Elamvazhuthi, K.; Berman, S. An Optimal Control Approach to Mapping GPS-Denied Environments Using a Stochastic Robotic Swarm. In *Robotics Research*; Bicchi, A., Burgard, W., Eds.; Springer International Publishing: Cham, Switzerland, 2018; Volume 2, pp. 477–493. [[CrossRef](#)]
33. Dirafzoon, A.; Lobaton, E. Topological mapping of unknown environments using an unlocalized robotic swarm. In Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, 3–7 November 2013; pp. 5545–5551. [[CrossRef](#)]

34. Dirafzoon, A.; Betthausen, J.; Schornick, J.; Benavides, D.; Lobaton, E. Mapping of unknown environments using minimal sensing from a stochastic swarm. In Proceedings of the 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, Chicago, IL, USA, 14–18 September 2014; pp. 3842–3849. [CrossRef]
35. Dirafzoon, A.; Bozkurt, A.; Lobaton, E. A framework for mapping with biobotic insect networks: From local to global maps. *Robot. Auton. Syst.* **2017**, *88*, 79–96. [CrossRef]
36. Ramachandran, R.K.; Wilson, S.; Berman, S. A Probabilistic Topological Approach to Feature Identification Using a Stochastic Robotic Swarm. In *Distributed Autonomous Robotic Systems*; Groß, R., Kolling, A., Berman, S., Frazzoli, E., Martinoli, A., Matsuno, F., Gauci, M., Eds.; Springer International Publishing: Cham, Switzerland, 2018; Volume 6, pp. 3–16. [CrossRef]
37. Batinović, A.; Oršulić, J.; Petrović, T.; Bogdan, S. Decentralized Strategy for Cooperative Multi-Robot Exploration and Mapping. *IFAC-PapersOnLine* **2020**, *53*, 9682–9687. [CrossRef]
38. Choudhary, S.; Carlone, L.; Nieto, C.; Rogers, J.; Christensen, H.I.; Dellaert, F. Distributed mapping with privacy and communication constraints: Lightweight algorithms and object-based models. *Int. J. Robot. Res.* **2017**, *36*, 1286–1311. [CrossRef]
39. Ramachandran, R.K.; Kakish, Z.; Berman, S. Information Correlated Lévy Walk Exploration and Distributed Mapping Using a Swarm of Robots. *IEEE Trans. Robot.* **2020**, *36*, 1422–1441. [CrossRef]
40. Bayer, J.; Faigl, J. Decentralized Topological Mapping for Multi-robot Autonomous Exploration under Low-Bandwidth Communication. In Proceedings of the 2021 European Conference on Mobile Robots (ECMR), Bonn, Germany, 31 August 2021–3 September 2021; pp. 1–7. [CrossRef]
41. Kit, J.L.; Dharmawan, A.G.; Mateo, D.; Foong, S.; Soh, G.S.; Bouffanais, R.; Wood, K.L. Decentralized Multi-Floor Exploration by a Swarm of Miniature Robots Teaming with Wall-Climbing Units. *arXiv* **2019**, arXiv:1908.05822. Available online: <http://arxiv.org/abs/1908.05822> (accessed on 11 May 2022).
42. Rone, W.; Ben-Tzvi, P. Mapping, localization and motion planning in mobile multi-robotic systems. *Robotica* **2013**, *31*, 1–23. [CrossRef]
43. Saeedi, S.; Trentini, M.; Seto, M.; Li, H. Multiple-Robot Simultaneous Localization and Mapping: A Review. *J. Field Robot.* **2016**, *33*, 3–46. [CrossRef]
44. Barca, J.C.; Sekercioglu, Y.A. Swarm robotics reviewed. *Robotica* **2013**, *31*, 345–359. [CrossRef]
45. Sutantyo, D.K.; Kernbach, S.; Levi, P.; Nepomnyashchikh, V.A. Multi-robot searching algorithm using Lévy flight and artificial potential field. In Proceedings of the 2010 IEEE Safety Security and Rescue Robotics, Bremen, Germany, 26–30 July 2010; pp. 1–6. [CrossRef]
46. Schroeder, A.; Ramakrishnan, S.; Kumar, M.; Trease, B. Efficient spatial coverage by a robot swarm based on an ant foraging model and the Lévy distribution. *Swarm Intell.* **2017**, *11*, 39–69. [CrossRef]
47. Pang, B.; Song, Y.; Zhang, C.; Wang, H.; Yang, R. A Swarm Robotic Exploration Strategy Based on an Improved Random Walk Method. *J. Robot.* **2019**, *2019*, 6914212. [CrossRef]
48. Dimidov, C.; Oriolo, G.; Trianni, V. Random Walks in Swarm Robotics: An Experiment with Kilobots. In *Swarm Intelligence*; Springer International Publishing: Cham, Switzerland, 2016; pp. 185–196. [CrossRef]
49. Kim, J.Y.; Kashino, Z.; Pineros, L.M.; Bayat, S.; Colaco, T.; Nejat, G.; Benhabib, B. A high-performance millirobot for swarm-behaviour studies: Swarm-topology estimation. *Int. J. Adv. Robot. Syst.* **2019**, *16*, 172988141989212. [CrossRef]
50. Kohlbacher, A.; Eliasson, J.; Acres, K.; Chung, H.; Barca, J.C. A low cost omnidirectional relative localization sensor for swarm applications. In Proceedings of the 2018 IEEE 4th World Forum on Internet of Things (WF-IoT), Singapore, 5–8 February 2018; pp. 694–699. [CrossRef]
51. Liu, L.; Fine, B.; Shell, D.; Klappenecker, A. Approximate characterization of multi-robot swarm “shapes” in sublinear-time. In Proceedings of the 2011 IEEE International Conference on Robotics and Automation, Shanghai, China, 9–13 May 2011; pp. 2886–2891. [CrossRef]
52. Carrillo-Arce, L.C.; Nerurkar, E.D.; Gordillo, J.L.; Roumeliotis, S.I. Decentralized multi-robot cooperative localization using covariance intersection. In Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, 3–7 November 2013; pp. 1412–1417. [CrossRef]
53. de Sá, A.O.; Nedjah, N.; de M Mourelle, L. Distributed and resilient localization algorithm for Swarm Robotic Systems. *Appl. Soft Comput.* **2017**, *57*, 738–750. [CrossRef]
54. Inoue, D.; Murai, D.; Ikuta, Y.; Yoshida, H. Distributed Range-based Localization for Swarm Robot Systems using Sensor-fusion Technique. In Proceedings of the 8th International Conference on Sensor Networks, Prague, Czech Republic, 26–27 February 2019; pp. 13–22. [CrossRef]
55. Cornejo, A.; Nagpal, R. *Distributed Range-Based Relative Localization of Robot Swarms*; Springer International Publishing: Cham, Switzerland, 2015; Volume 107, pp. 91–107.
56. Klingner, J.; Ahmed, N.; Correll, N. Fault-tolerant Covariance Intersection for localizing robot swarms. *Robot. Auton. Syst.* **2019**, *122*, 103306. [CrossRef]
57. de Sá, A.O.; Nedjah, N.; de M Mourelle, L. Multi-hop Collaborative Min-Max localization. In Proceedings of the 2015 IEEE 6th Latin American Symposium on Circuits & Systems (LASCAS), Montevideo, Uruguay, 24–27 February 2015; pp. 1–4. [CrossRef]
58. Ma, D.; Er, M.J.; Wang, B.; Lim, H.B. Range-free wireless sensor networks localization based on hop-count quantization. *Telecommun. Syst.* **2012**, *50*, 199–213. [CrossRef]

59. Loefgren, I.; Ahmed, N.; Frew, E.; Heckman, C.; Humbert, S. Scalable Event-Triggered Data Fusion for Autonomous Cooperative Swarm Localization. In Proceedings of the 2019 22th International Conference on Information Fusion (FUSION), Ottawa, ON, Canada, 2–5 July 2019; pp. 1–8.
60. Fukui, S.; Naruse, K. *Swarm EKF Localization for a Multiple Robot System with Range-Only Measurements*; Springer International Publishing: Cham, Switzerland, 2013; pp. 91–103. [[CrossRef](#)]
61. Elfes, A. Using occupancy grids for mobile robot perception and navigation. *Computer* **1989**, *22*, 46–57. [[CrossRef](#)]
62. González-Baños, H.H.; Latombe, J.-C. Navigation Strategies for Exploring Indoor Environments. *Int. J. Robot. Res.* **2002**, *21*, 829–848. [[CrossRef](#)]
63. Thrun, S. Probabilistic robotics. *Commun. ACM* **2002**, *45*, 52–57. [[CrossRef](#)]
64. Yamauchi, B. A frontier-based approach for autonomous exploration. In Proceedings of the 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation CIRA'97. "Towards New Computational Principles for Robotics and Automation", Monterey, CA, USA, 10–11 July 1997; pp. 146–151. [[CrossRef](#)]
65. Katoch, S.; Chauhan, S.S.; Kumar, V. A review on genetic algorithm: Past, present, and future. *Multimed. Tools Appl.* **2021**, *80*, 8091–8126. [[CrossRef](#)] [[PubMed](#)]
66. Hart, P.E.; Nilsson, N.J.; Raphael, B. A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Trans. Syst. Sci. Cybern.* **1968**, *4*, 100–107. [[CrossRef](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.