Cooperative Exploration Strategy for Micro-Aerial Vehicles Fleet

Nesrine Mahdoui, Vincent Frémont, Enrico Natalizio

Cooperative Exploration Strategy for Micro-Aerial Vehicles Fleet

Nesrine Mahdoui[1], Vincent Frémont[1], Enrico Natalizio[1]

Abstract— In this paper, the problem of the exploration of an unknown environment by deploying a fleet of Micro-Aerial Vehicles (MAV) is considered. As a single robot has already proven its efficiency for this task, the challenge is to extend it to a multi-robots system to reduce the exploration time. For this purpose, a cooperative navigation strategy is proposed based on a specific utility function and inter-robots data exchange. The novelty comes from the exchange of the frontiers points instead of maps, which allows to reduce computation and data amount within the network. The proposed system has been implemented and tested under ROS using the Gazebo simulator. The results demonstrate that the proposed navigation strategy efficiently spreads robots over the environment for a faster exploration.

I. INTRODUCTION

In the recent years, researchers have become interested in using MAV fleets to implement new features and execute new tasks over large scale environments such as disaster surveillance, inundation monitoring, environment discovery, infrastructure inspection, etc. Among these, the exploration and mapping of large areas is still a challenging task and one of the fundamental problem in aerial robotics with limited embedded processing capabilities and energy. In this way, many Simultaneous Localization And Mapping algorithms (SLAM) have been proposed for a single robot, and recent research trends are oriented to their extension to MAV fleets. The possibility of using multiple agents is often proposed, due to different advantages such as: increased efficiency, reduced mission time, robustness to agent failure (scalability), etc.

Nevertheless, when it comes to multi-MAV exploration, many challenges arise. When working in team, interferences may occur between the embedded sensors. Also, agents (robots) have to avoid collisions with the obstacles and the other members of the team. Furthermore, a fleet coordination has to be set up to avoid to visit areas that have been already explored, or to visit the same area at the same time. To be able to have a cooperative behaviour, fleet members have to collect information about each others. This is mostly done by wireless communication between MAV.

Compared to a group of robots that does not involve communication, inter-robots data exchange is beneficial even if it is limited. However, sometimes communication may fail, so the MAV have to be able to continue the exploration on their own.

In this paper, an algorithm for coordinating a MAV fleet is presented in order to efficiently explore an unknown environ-

ment using an unknown number of robots. The purpose of this paper is to investigate the following research directions:

- Using multi-robots system for exploration efficiency;
- Coordinating the fleet by using inter-robots communications;
- Improving area coverage using simultaneous exploration and coverage strategies;
- Treating the considered system as a system of system to handle fleet scalability.

Based on these research directions, the main contributions of this work are the following:

- Introducing a decentralized and coordinated exploration algorithm using both limited sensor range and communication range;
- Increasing fleet coordination scalability (adding or removing robots);
- Spreading autonomous MAV within the environment to speed up the exploration and to reduce energy consumption;
- Handling of limited information exchange between MAV.

In multiple robots system coordination, data exchange mainly focuses on maps. In this paper, the proposed idea is to exchange frontiers data instead. To the best of our knowledge, this is first work that proposes a cooperative navigation strategy based on frontiers data exchange.

This paper is organized as follows. In the next Section, related works on the multiple robots exploration and coverage systems is proposed. Then, the proposed strategy for multi-MAV coordination is presented in Section III. The following Section describes the simulations performed to assess the reliability of our technique. And finally, we conclude the paper with Section V.

II. RELATED WORKS

Exploration strategies can be classified in different manners. Among them, random explorations, where a robot explores random points around itself within a given distance, have been proposed. Frontier features exploration where a robot explores unknown areas nearby already explored areas have been also published [1], [2]. Finally, human directed exploration where the human directs the robot for exploration have been presented. In this paper, a focus on frontier-based exploration [1], [2] is proposed. The exploration strategy can also be combined with coverage constraints [3], where both the target and the path to reach it are processed.

In coordinated multi-robots exploration, one robot has to choose its target state while running coordination with other

robots. One way to do that is to have a joint map in a central server/robot. A goal state can then be assigned for each robot [4], [5]. An alternative way to represent the environment is to use occupancy grid map. In [6], the authors introduce coverage maps, where a posterior probability distribution about whether the corresponding cell is covered by an obstacle or not, is stored in each cell of the grid.

The problem of group exploration deals with the challenge of spreading the robots as much as possible for a full coverage while maintaining communications between them. The rolling dispersion approach [7] achieves full exploration with a small number of robots and an environment that are both unknown. The dispersion and coordination of robots are based on data exchange and processing. Mostly, for that purpose, the robots share their maps and merge them to get a joint map [8], [9]. Some cooperative approaches in [10], [11] propose to take into account constraints on communication limits in order to achieve an efficient and distributed exploration.

When it comes to collaborative mapping and navigation problems [12], tasks can be shared between cartographers that look for target and a navigator that has to reach it. In [13], for the assignment of different targets to different robots, the costs of reaching a target point and the utility of target points are taken into account. In [5], the idea is to maximize the overall utility by minimizing the potential for overlap in information gain amongst the various robots. Another approach [14] is proposed where the solution is to switch from cooperative to individual exploration behavior if some conditions are not satisfied.

Related to the exploration and coverage problem, a possible approach is to divide the environment into regions to explore. The art gallery problem [15] or also called museum problem is an example. It is a computational geometry problem that consists of triangulation to optimally place cameras in a gallery (interior of an n-vertex simple polygon) so that the whole gallery is covered. This problem has been improved and different algorithms [16] have been proposed depending on the polygon form, the segment number, the cameras number, etc. Based on the same context of environment subdivision, algorithms based on Voronoi approach have been proposed [17] to cope with intersection of sensing area, redundant data, communication interference, and energy waste sensor and radio ranges. In [18], the environment is divided into segments based on its structure. And then robots are assigned to explore the corresponding areas.

This exploration topic has been also well investigated in swarm robotics and can be used with extended resources [19] to do tasks such as foraging, discovering and coverage. In this context, different methods are used. Some of them [20] are based on a combination of pheromones to keep track of the visited places for navigation.

III. MAV FLEET COORDINATION

This section discusses the concepts underlying the coordination strategy. The implementation details are presented in section IV. In the proposed approach, a decentralized algorithm to coordinate multi-MAV fleet is adopted. The aim is to efficiently explore an unknown environment by reducing the time needed for this mission with limited embedded processing capabilities. The considered fleet may be composed of an unknown number of heterogeneous robots.

A. Clusters and Status Selection

Since the communication range can be limited, only some agents may communicate with others and thus they form a so-called cluster. In each of the clusters, one MAV is responsible for the coordination. This MAV has a *Leader* status. It is a robot that is able to make cooperative decision for others taking into account some specific exchanged information. Other MAV that belong to the cluster have an *Explorer* status.

Since the decision is decentralized and, in order to avoid the mission interruption if the *Leader* is out of order, two principles are adopted. The first is that each MAV is autonomous, so when it does not receive a response from its *Leader*, it can follow its exploration on its own based on its local data. The second is that if the *Leader* goes out of the communication range, a new one is then chosen. Indeed, the status is always updated to cope with cluster changes. A MAV is proclaimed *Leader* when its identification number is the smallest in the cluster [21]. Thus, the algorithm of status selection is distributed among the team and it ensures uniqueness of the *Leader* status.

So at the beginning of the mission, each member of the fleet has a *Leader* status by default, and gradually, when the robots share their identification numbers with each other, each one chooses its appropriate status.

B. The Exploration Process

The exploration process outline is presented in Fig. 1. It is implemented on all MAV in the fleet. Depending on the MAV status, the behavior is adapted for the mission.
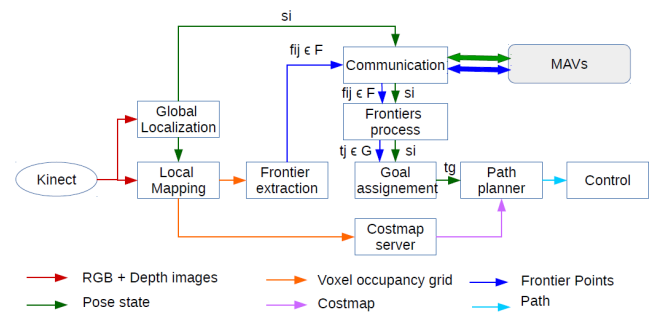

Fig. 1. Outline of the exploration process implementation.

1) The Mapping Process: In order to create a map of the environment, localization information and point cloud reconstruction are used. The mapping process is handled by the Octomap mapping framework [22] based on the octree encoding [23]. Indeed, the Octomap framework is a state estimation and a mapping process that presents several advantages especially for memory consumption. It is also largely adopted by the robotic community. Using only a low

cost RGB-D camera, the Octomap framework can generate, on-line, an accurate 3D map for ground robot as well as flying ones [24], in both challenging scenarios like fast camera motion or feature-less environments [25].

2) Frontiers and Information Gain: As said previously, each robot must have a goal to reach, while avoiding collisions and maximizing exploration in a minimum time. For this purpose a frontier-based approach is proposed. The frontier environment represents all the free voxels, in the explored area, that are adjacent to obstacle or unknown spaces.

For all these frontiers f_{ij} of robot i in *F*, the corresponding information gain is computed as:

- Assigning, at the beginning, the same value of 1 to all candidate states. Then, this value is reduced, using a probability function, for the adjacent candidate states according to their relative distance to it [4].
- Considering the voxels in the map where the number of unknown and not occluded voxels in the view frustum of the target goal t_g are estimated [3].

The first method is general and does not depend on the updated explored area of the map. The second one is more precise because it depends on the real estimation of the amount of information gained when visiting this state, but it requires more computation. In our case, the information gain $I(f_{ij})$ is the number of unknown voxels that are adjacent to this target [26].

These frontiers are then exchanged with the other robots of the fleet and processed to keep only the potential candidate goals. Candidate goals are defined as all target goals $t_g \in G$ located on frontier *F* between free and unknown cells. All frontiers on overlapping explored area are also removed. These candidates are selected such that $dmin(s_i, t_g) < D_{com}$, where $dmin(s_i, t_g)$ is the minimum distance between neighbors' state s_i that was shared by communication and the target goal t_g. D_{com} is a minimum range that is allowed for communication between two robots. This distance guarantees that the robots maintain a certain communication without imposing a rigid distance between them.

3) The Utility Function: An utility function is used to estimate the utility of going to a certain state goal while taking into account the cost and the information gained. The proposed utility function is inspired by [3], where it is used for a single robot exploration. In this paper, it has been extended for multiple MAV case.

$$U(t_j) = I(t_j) \exp(-\lambda.(dmin(s_i, t_j)$$
$$+ n_C / \sum_{k=1, k \neq i}^{n_C} (dmin(s_k, t_j)))) \quad (1)$$

where $I(t_j)$ is the information gain of the target goal t_j, λ is a tuning parameter for trade-off between rapid exploration and filling in details, n_C is the number of neighbors in the cluster *C*, and $dmin(si, t_j)$ is the shortest distance from robot *i*'s state to the candidate target goal.

Eq. 1 is a function of $x = dmin(s_i, t_j)$ that represents the minimum distance between current robot state and the candidate target state t_j. The smaller the distance $dmin(s_i, t_j)$, the more important the utility function. Therefore, the position t_j is more likely to be chosen as a future target.

The parameter $\sum_{k=1, k \neq i}^{n_C} (dmin(s_k, t_j))$ represents the sum of minimum distance between target state t_j and the neighbors' state s_k. So, if the candidate state has neighboring robots that are too far, the utility function will decrease and the candidate target is more likely to be chosen, and vice versa.

Actually two cases where the sum $\sum_{k=1, k \neq i}^{n_C} (dmin(s_k, t_j))$ can be equal to zero. The first one is when there are no neighbors. In this case, $n_C = 0$ and the the equation becomes: $U(t_j) = I(t_j) \exp(-\lambda.(dmin(s_i, t_j))$. The second case is when the candidate state goal and the neighbors state are almost confused (too close to each other). In this case the target goal is less likely to be chosen.

Actually n_C is added to compute the distance average of neighbors. This is because the sum of them could be important either when one far neighbor is considered and it can be choose as t_j; or when several neighbors exist but they are at a relatively short distances.

4) Goal Assignment: Based on the computed utility functions, the robot has to choose its next state. For a single robot exploration, the best way is to go to the closest unexplored area that has the lowest cost path while maximizing the information gained [3].

However, when robots operate in a team, they have to be efficiently spread in the environment while avoiding to go to places already explored. Therefore, a cooperative exploration strategy is needed. Thereby, a goal is assigned to each robot in the team based on collected information about others. This assignment of pairs (robot with state goal) is satisfying a trade-off between distance to this location and the information gained by visiting it. The algorithm 1 proposes a cooperative decision making to assign robot to state.

To assign appropriate target for each MAV, algorithm 1 is used to maximize the utility function. The information gain of adjacent targets [4] is then reduced within the distance of sensor range.

This algorithm is implemented on-board at each member of the fleet, since each one is likely to become a *Leader*. However, it is run only by one robot: the *Leader*. The assigned targets are then sent back to the concerned MAV. When a MAV, for some reason (out of communication range when the states are sent back, communication issues, etc.), does not receive the target assigned after an elapsed time, it will decide autonomously, depending on its collected information, which is the next state to go to.

C. Communication and Collected Data Management

During the mission and to take advantages of the MAV fleet, the robots exchange some information among them.

Input : Eventual target goals t_j with $t_j \in G$, information gain about them $I(t_j)$, states of *Leader*(me) and *Explorer* robots s_i in the cluster C, number of robots in the cluster n_c

Output: $\theta(i,j)$ assignment of robot i with target j

Assigned target $\mathcal{G} = \emptyset$

while *robot i still without target goal* **do**

   /* Assign available states to robots in the cluster respecting a trade-off between distance and the amount of information gained. */

   while $t_j \notin \mathcal{G}$ **do**

      $U(t_j) = I(t_j)\exp(-\lambda.(dmin(s_i,t_j) + n_C/\sum_{k=1,k\neq i}^{n_C}(dmin(s_k,t_j))))$

   end

   $t_{j'} = argmax_{t_j \in G \cap \mathcal{G}} U(t_j)$

   Associate robot i to target state $j' \to \theta(i,j')$

   /* reduce the information gain of the adjacent targets in distance of sensor range */

   /*) */

   while $t_j \in G \cap \mathcal{G}$ **do**

      if $dist(t_{j'}, t_j) <= sensorRange$ **then**

         $I(t_j) = I(t_j) - I(t_j).\exp[-\gamma.\|t_{j'} - t_j\|_2]$

      end

   end

   $\mathcal{G} = \mathcal{G} \cup t_{j'}$

end

Algorithm 1: Goal assignment for coordinated multi-MAV.

This communication is a key component for a cooperative exploration strategy. As introduced in the section III-A, MAV in the same clusters are those that communicate with each others and share the same *Leader*. Data exchange in the system can be classified into two types. The first one is local data where the MAV share its identification number and its state with its neighbors. This is used for status selection. Furthermore, the MAV exchange global data for cooperative goal assignment purpose. Instead of sharing their maps [4], [5], [8], [21], [26], it consists on sending frontiers f_i with respective information gain $I(f_i)$ to the *Leader* of the cluster. This novelty allows the fleet to share less information on their network, to divide the computation of frontiers and information gained over all members, and to save the map merging time and the required memory.

Collecting data from robots allows the achievement of a global view of the explored environment and, thus, it facilitates the task of assigning goals to not visited areas. When it comes to map exchange, the map merging step allows the detection of the overlap between maps and, thus, it avoids redundant target goals. The same process should be done when sharing frontiers to avoid choosing target goals situated in an already mapped environment. Indeed, the received frontiers from the *Explorer* are processed to remove the overlapping areas. The contouring shape of frontiers are

determined. Let us consider f_{ij} frontiers of robot j and f_{ik} frontiers of robot k. If f_{ij} is within the contouring shape of robot k, then f_{ij} is removed. And if f_ik is within the contouring shape of robot j, f_{ik} is then removed. So that all points that belong to their intersection are removed. The candidate targets goal is then obtained and the goal assignment algorithm (see Algo. 1) is applied. This process is done by pairs if there are more than two set of frontiers.

D. Path planning and control

In the path planning and control layer, the MAV plans a path and follows it to reach the assigned target. By retrieving the on-line built occupancy grid map, the MAV constructs a corresponding cost-map. Given this local cost-map, the MAV plans a path from its current state to the target goal following the Dijkstra's algorithm [27].

IV. EXPERIMENTAL EVALUATION

The exploration strategy described above has been implemented and evaluated in simulation.

A. Implementation Details

For the evaluation, we have used ROS [1] and Gazebo simulator [2] on a 2.60GHz i7 Linux machine. In each simulation, we have used the Hector quad-rotor model [28] for the mobile robot. Each MAV is equipped with a single front-looking RGBD camera sensor with outputs of 640×480 pixels. MAV fly in an indoor environment as shown in Fig. 2. To fairly compare simulations, the same parameters and conditions are kept for all runs.
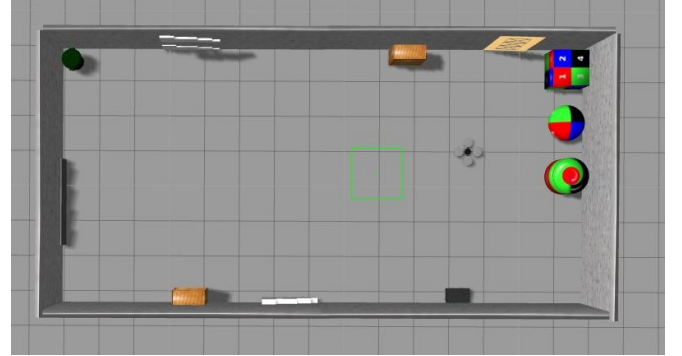

Fig. 2. Simulated indoor environment in Gazebo.

B. Exploration evaluation

Experiments have been performed in order to validate the proposed exploration strategy, the use of multiple MAV system and to demonstrate the benefits of cooperative team. The Fig. 3 shows a mapped environment of Fig. 2 after an autonomous exploration deployment of a single MAV.

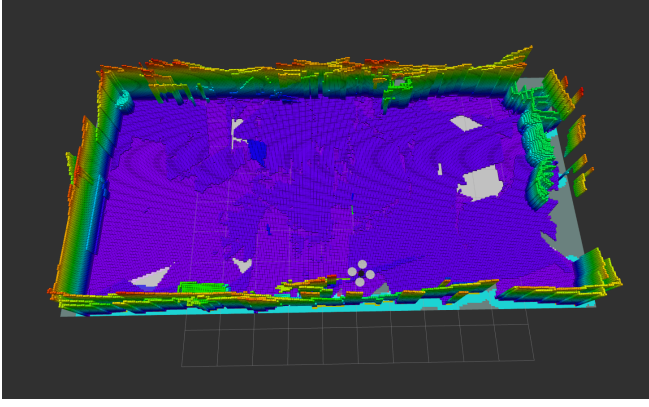[1]http://www.ros.org/
[2]http://gazebosim.org/

Fig. 3. Local map of a single robot autonomous exploration.

1) Exploration with a team of MAV: The goal of the first experiment is to demonstrate the advantage of our approach to cover efficiently an unknown environment using a fleet of MAV instead of a single robot. In all runs execution, it is assumed that the environment is initially unknown aw well as the number of MAV. The Tab. I shows the estimated mapping time for different MAV number using the cooperative navigation strategy. It is important to notice that the more the number of MAV increases, the less the mapping time is important. Indeed, the time is reduced by 7 minutes from using one to two MAV.

Run	# Mapping robots	Mapping time
1	1	23 min
2	2	16 min

TABLE I

EXPLORATION RUNS EVALUATION USING DIFFERENT MAV NUMBERS.

2) Cooperative versus Non Cooperative MAV Exploration: The second simulations are designed to show the capability of the proposed approach to improve the exploration performances when using coordinated MAV compared to uncoordinated ones. For this, the robots are able to build their own maps on-line while exploring the environment. Then, they exchange their identification numbers within their respective clusters to nominate the *Leader*. After the status selection, *Explorers* determine and send to the *Leader* their frontiers features f_i. Furthermore, the *Leader* needs to remove frontiers that belong to the overlapping areas in addition to frontiers that are near obstacles. This process is shown in Fig.4. It allows us to avoid to assign target goals t_j that do not belong to an already explored area.

Additional simulations with MAV fleet that are cooperating compare to others that are not, have been carried out. The results are summarized in Tab. II. One can notice that the cooperative fleet achieves the mission by using less goal targets than the non cooperative one.

V. CONCLUSION AND FUTURE WORKS

In this paper, cooperative MAV exploration strategy that is able to cope with limited sensor and communication range

Run	# Mapping robots	# Targets goal reached/MAV
Cooperative	2	27 for MAV1 and 18 for MAV2
Non cooperative	2	41 for each MAV

TABLE II

COOPERATIVE VERSUS NON COOPERATIVE EXPLORATION RUNS EVALUATION.

has been proposed. The MAV are spread in the environment while avoiding visiting already known areas or exploring the same area at the same time. Also, frontiers data exchange has been proposed instead of usual maps exchange to allow less memory consumption and faster computations.

As future works, we plan to run real world environment experiments with the proposed exploration strategy on-board MAV and also to increase the number of robots within the fleet. We also plan to extend our approach to use relative localization techniques within our exploration strategy. From the networking side, we plan to investigation new communication protocols to improve the data exchange robustness.

REFERENCES

[1] C. Zhu, R. Ding, M. Lin, and Y. Wu, "A 3d frontier-based exploration tool for mavs," in *Tools with Artificial Intelligence (ICTAI), 2015 IEEE 27th International Conference on.* IEEE, 2015, pp. 348–352.

[2] D. Holz, N. Basilico, F. Amigoni, and S. Behnke, "Evaluating the efficiency of frontier-based exploration strategies," in *Robotics (ISR), 2010 41st International Symposium on and 2010 6th German Conference on Robotics (ROBOTIK).* VDE, 2010, pp. 1–8.

[3] L. Heng, A. Gotovos, A. Krause, and M. Pollefeys, "Efficient visual exploration and coverage with a micro aerial vehicle in unknown environments," in *2015 IEEE International Conference on Robotics and Automation (ICRA).* IEEE, 2015, pp. 1071–1078.

[4] W. Burgard, M. Moors, C. Stachniss, and F. E. Schneider, "Coordinated multi-robot exploration," *IEEE Transactions on robotics*, vol. 21, no. 3, pp. 376–386, 2005.

[5] R. Simmons, D. Apfelbaum, W. Burgard, D. Fox, M. Moors, S. Thrun, and H. Younes, "Coordination for multi-robot exploration and mapping," in *AAAI/IAAI*, 2000, pp. 852–858.

[6] C. Stachniss and W. Burgard, "Mapping and exploration with mobile robots using coverage maps," in *Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, vol. 1. IEEE, 2003, pp. 467–472.

[7] E. A. Jensen and M. L. Gini, "Rolling dispersion for robot teams." in *IJCAI.* Citeseer, 2013.

[8] M. J. Schuster, C. Brand, H. Hirschmuller, M. Suppa, and M. Beetz, "Multi-robot 6d graph slam connecting decoupled local reference filters," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on.* IEEE, 2015, pp. 5093–5100.

[9] C. Forster, S. Lynen, L. Kneip, and D. Scaramuzza, "Collaborative monocular slam with multiple micro aerial vehicles," in *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference.* IEEE, 2013.
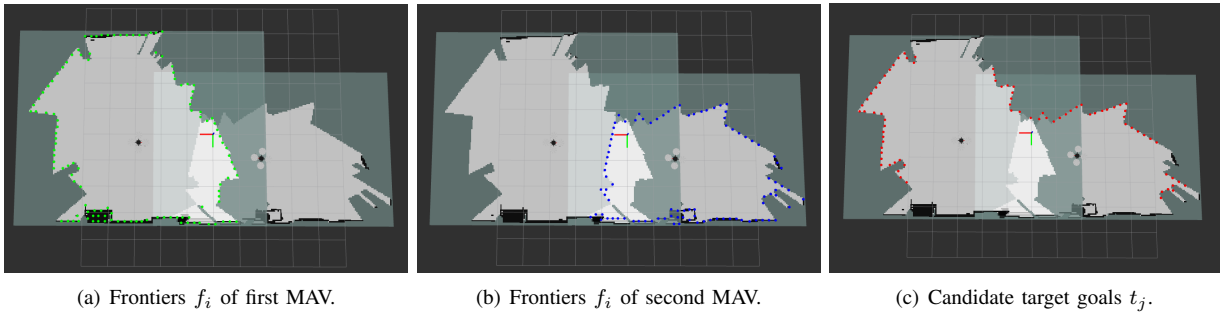
(a) Frontiers f_i of first MAV. (b) Frontiers f_i of second MAV. (c) Candidate target goals t_j.

Fig. 4. Frontiers process: Frontiers f_i of MAV 1 and 2 are respectively shown in images (a) and (b). In image (c), the candidate target goals, that represent frontiers after process, are shown. The process consist on removing both frontiers that belong to overlapped areas; and those that are adjacent to obstacles.

[10] J. Yuan, Y. Huang, T. Tao, and F. Sun, "A cooperative approach for multi-robot area exploration," in *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*. IEEE, 2010, pp. 1390–1395.

[11] W. Sheng, Q. Yang, J. Tan, and N. Xi, "Distributed multi-robot coordination in area exploration," *Robotics and Autonomous Systems*, vol. 54, no. 12, pp. 945–955, 2006.

[12] W. W. Cohen, "Adaptive mapping and navigation by teams of simple robots," *Robotics and autonomous systems*, vol. 18, no. 4, pp. 411–434, 1996.

[13] W. Burgard, M. Moors, D. Fox, R. Simmons, and S. Thrun, "Collaborative multi-robot exploration," in *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*, vol. 1. IEEE, 2000, pp. 476–481.

[14] W. Wu and F. Zhang, "Robust cooperative exploration with a switching strategy," *IEEE Transactions on Robotics*, vol. 28, no. 4, pp. 828–839, 2012.

[15] E. Krohn, "Survey of terrain guarding and art gallery problems," *Unpublished manuscript. November*, 2007.

[16] J. O'rourke, *Art gallery theorems and algorithms*. Oxford University Press Oxford, 1987, vol. 57.

[17] M. Viera, L. Viera, L. B. Ruiz, A. A. F. Loureiro, A. O. Fernandes, and J. M. S. Nogueira, "Scheduling nodes in wireless sensor networks: A voronoi approach," in *Local Computer Networks, 2003. LCN'03. Proceedings. 28th Annual IEEE International Conference on*. IEEE, 2003, pp. 423–429.

[18] K. M. Wurm, C. Stachniss, and W. Burgard, "Coordinated multi-robot exploration using a segmentation of the environment," in *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*. IEEE, 2008, pp. 1160–1165.

[19] S. Alers, K. Tuyls, B. Ranjbar-Sahraei, D. Claes, and G. Weiss,

"Insect-inspired robot coordination: foraging and coverage," *Artificial life*, vol. 14, pp. 761–768, 2014.

[20] E. Ventocilla, "Swarm-based area exploration and coverage based on pheromones and bird flocks," 2013.

[21] D. Fox, J. Ko, K. Konolige, B. Limketkai, D. Schulz, and B. Stewart, "Distributed multirobot exploration and mapping," *Proceedings of the IEEE*, vol. 94, no. 7, pp. 1325–1339, 2006.

[22] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "Octomap: An efficient probabilistic 3d mapping framework based on octrees," *Autonomous Robots*, vol. 34, no. 3, pp. 189–206, 2013.

[23] D. Meagher, "Geometric modeling using octree encoding," *Computer graphics and image processing*, vol. 19, no. 2, pp. 129–147, 1982.

[24] M. Sanfourche, A. Plyer, A. Bernard-Brunel, and G. Le Besnerais, "3dscan: Online ego-localization and environment mapping for micro aerial vehicles," *AerospaceLab*, no. 8, pp. 1–17, 2014.

[25] F. Endres, J. Hess, J. Sturm, D. Cremers, and W. Burgard, "3-d mapping with an rgb-d camera," *Robotics, IEEE Transactions on*, vol. 30, no. 1, pp. 177–187, 2014.

[26] T. Andre, D. Neuhold, and C. Bettstetter, "Coordinated multi-robot exploration: Out of the box packages for ros," in *Globecom Workshops (GC Wkshps), 2014*. IEEE, 2014, pp. 1457–1462.

[27] H. Wang, Y. Yu, and Q. Yuan, "Application of dijkstra algorithm in robot path-planning," in *2011 Second International Conference on Mechanic Automation and Control Engineering*, July 2011, pp. 1067–1069.

[28] J. Meyer, A. Sendobry, S. Kohlbrecher, U. Klingauf, and O. von Stryk, "Comprehensive simulation of quadrotor uavs using ros and gazebo," in *3rd Int. Conf. on Simulation, Modeling and Programming for Autonomous Robots (SIMPAR)*, 2012, p. to appear.