# The Next Frontier: Combining Information Gain and Distance Cost for Decentralized Multi-Robot Exploration

Rafael Gonçalves Colares
Computer Science Department
UFMG
Belo Horizonte, Brazil
rcolares@dcc.ufmg.br

Luiz Chaimowicz
Computer Science Department
UFMG
Belo Horizonte, Brazil
chaimo@dcc.ufmg.br

## ABSTRACT

The exploration of unknown environments is an important task for autonomous robots. When multiple robots are able to coordinate themselves to explore different areas of the environment, the exploration efficiency can be greatly improved. In this paper, we present a decentralized approach for multi-robot exploration that leverages the classical frontier based methods. We propose a utility function that takes into consideration the information gain and the distance costs of the frontiers to guide the exploration. Moreover, by exchanging information and merging maps, robots are able to better coordinate and avoid the exploration of redundant areas. Experiments performed with both simulated and real robots demonstrate the effectiveness of this approach.

## CCS Concepts

•**Computing methodologies** → **Robotic planning;** *Multi-agent planning;* Planning under uncertainty;

## Keywords

Integrated Exploration, Coordination, Multi-Robot

## 1. INTRODUCTION

When exploring unknown environments, an autonomous robot can use Simultaneous Localization and Mapping (SLAM) techniques to build a reliable map of the environment. But exploring this environment at random, can reduce the exploration efficiency and the map quality. Thus, the use of a planning stage capable of keeping the balance between map quality and exploration efficiency, in conjunction with the SLAM algorithm, can be an alternative to this. This is normally called *integrated exploration.*

The most common integrated exploration technique used in the literature is the "frontier exploration" [13]. It consists in finding regions on the map that are on the frontier between explored and unexplored spaces. By exploring these frontiers, the robot is able to cover the entire environment. In [13], the exploration strategy is to choose the nearest frontier from the robot's position. For this reason, it is called Near-Frontier Exploration (NFE).

The use of a multi-robot system can improve the efficiency of almost every robotic task, including the exploration of unknown environments. When robots are able to coordinate themselves to explore different areas of the environment, the performance can be greatly improved. But the use of multiple robots can impose new challenges in the planning stage, such as sensor interference, obstructed paths, etc. Also, without any coordination, more than one robot can decide to explore the same area, resulting in a waste of exploration effort and time. Thus, in order to obtain the maximum exploration efficiency on a multi-robot system, the planning stage must be able to coordinate the robots, preferably without the need of a centralized coordination.

In this paper, we propose a multi-robot exploration approach based not only on the frontier's distance, but also on the amount of information that can be added by exploring a frontier. By analyzing the occupancy grid map from the SLAM algorithm, we are able to find frontiers with more information to be explored, and by using this information together with the frontier distance, we manage to compute a suitable utility value for each frontier on the map. We use a decentralized coordination mechanism, by which robots exchange information when they meet and avoid exploring redundant regions. This is done by penalizing the utility of frontiers on regions potentially explored by other robots.

This paper is organized as follows. Section 2 brings a brief literature review on multi-robot exploration. Section 3 gives an overview of our approach, detailing each part of the system. Experimental results are presented in Section 4, while Section 5 brings conclusions and directions for future work.

## 2. RELATED WORK

Integrated exploration strategies can be based on a variety of factors, such as information gain, distance costs, etc. Several works in the literature have tackled these factors in different ways. Some use only the distance of the frontiers [14, 2, 4] to decide the best region to explore. Yamauchi et al. [14], for instance, proposed a multi-robot integrated exploration strategy based on the Near-Frontier Exploration.



*SAC 2016,April 04-08, 2016, Pisa, Italy*



http://dx.doi.org/10.1145/2851613.2851706

They applied the NFE to every robot with no coordination, which could result on more than one robot exploring the same frontier. Burgard et al. [2] noticed this problem and proposed a coordination mechanism that reduces the utility of frontiers that are close to other robots.

Since the method on [2] demands robots to share a common map, it may be difficult to implement on the real world due to need of knowing every robot poses at all times. Thus, Bugard et al. [3] proposed an approach where each robot keeps a list of the last known pose of every robot so that, if it loses the connection with other robots, it continues the exploration assuming the last pose of each robot. It uses a utility function that is based on the distance to the frontier and the presence of other robots near that frontier. Fox et al. [6] avoid the limitation of knowing other robots poses by making every robot capable of mapping individually while constantly trying to meet other robots to exchange information. They use a NFE approach when a robot is exploring alone and centralized coordination when robots meet.

Some approaches try to maximize the information of the system by analyzing the status of the map and predicting the frontier which would increase the knowledge of the environment [5, 11]. Stachniss et al. [12], for example, proposed an integrated exploration strategy based on information gain. For them, information gain means reducing the entropy of the system. So, the robot always takes the action that would reduce localization errors and map entropy.

Another approach for the integrated exploration problem was proposed by Juliá et al.[9]. On that work, the robots use a hybrid controller that is active when the robot is close to a obstacle and is based on behaviors to decide where to explore next. On [8] the authors compared this approach with other strategies, like classic NFE [13], the method proposed in [3], among others, and realized that the best strategy depends directly on the application.

The work of Faigl et al. [4] proposed an approach for the multi-robot exploration task where each robot is assigned to a frontier based on a heuristic that solves the travel salesman problem. They compare their approach using different distance calculation methods for the exploration of the nearest frontier.

Instead of analyzing the whole map, Holz et al. [7] claimed that segmenting the environment and assigning robots to frontiers inside the segments can improve the exploration. Following this idea, Bautin et al. [1] proposed a multi-robot exploration approach, called MinPos, in which a robot only explores a frontier if it is the closest one to that frontier. This is done by using wavefronts that start on each frontier until they reach the robot. Then, if the robot is the closest to that frontier it is assigned to it. This results on spreading the robots on the environment. Also, Kaperski et al. [10] proposed a novel approach where a wavefront is started from the robot towards the frontiers. They compared it with the MinPos approach and the near-frontier exploration, showing similar results to both.

Our main contribution is to leverage and combine some of these approaches in order to improve the quality and efficiency of multi-robot exploration. Similarly to [12], we use the concept of information gain but augmented with functions that take into consideration the distance to the frontiers and the multi-robot coordination. Also, inspired by the work of Fox et al. [6], robots map the environment individually and only exchange information when they meet each other. For this, a map merging technique is developed. Thus, by combining these ideas, we are able to coordinate a team of robots in an exploration task, where each robot is individually controlled using a utility function based on information gain and distance cost and only exchanges information when it meets other robots.

# 3. METHODOLOGY

As mentioned on Section 1, our objective is to develop a multi-robot exploration system based on the information and the distance to a frontier. In our approach, each robot explores the environment building its own map, initially without information about other robots. When two robots meet, they exchange information, merge their maps together and use this joint information to coordinate.

The exploration is based on a utility function that analyzes the information and the distance of every frontier on the map. We also coordinate the robots so they do not explore the same frontier by adding a penalizing factor to the utility function on regions that other robots are already exploring. The utility function is described as:

$$\mathrm{U}(f) = \mathrm{Inf}(f) + \mathrm{Dist}(f) - \mathrm{Coord}(f) \tag{1}$$

where, $f$ is a frontier in the estimated map, $U$ is the utility value, $Inf$ is the information factor, $Dist$ is the distance factor and $Coord$ is the coordination factor of the utility function. The next sections will explain these three factors in details.

## 3.1 Information Factor

Usually, SLAM algorithms represent the environment map using an occupancy grid. Each cell on an occupancy grid has a value that represents the probability of that particular cell being occupied. If this value is 1, then the cell is occupied, if it is 0, then the cell is free, if it is between 0 and 1, then the cell has some uncertainty that needs to be resolved. Our approach considers that cells with uncertainty and cells that have not been mapped yet contain valuable information about the environment since we can not be sure if they are occupied or free. In this context, information gain means exploring regions where there are a large concentration of cells with uncertainty and cells that have not been mapped yet.

In order to compute the information factor on each frontier, a window of $n \times n$ is scanned across the occupancy grid, calculating the importance of each cell and the cell interactions in a frontier. Firstly, the concept of **Information Potential** (IP) must be introduced. The IP of a cell gives an estimate of the exploration potential that cell carries, instead of how much information that cell carries as calculated by Entropy. Basically, the idea is that cells with probability of being occupied close to 0.5 have higher exploration potential, which decreases as cell probabilities approach 0 or 1. Thus, IP is modeled as a non-normalized Gaussian function centered on 0.5. Although they have different semantic values, for the IP computation unknown cells are treated as a cell with 0.5 probability.

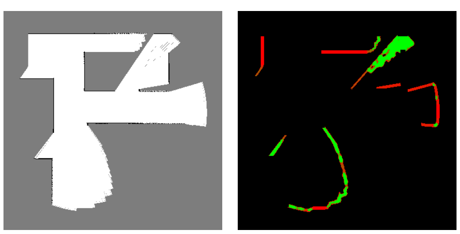
In the next step, we evaluate the interaction between a cell $c$ and its neighbors. We consider that the IP of a neighbor cell gives us information about $c$, for example, if a neighbor cell is free, $c$ have a high probability of being free as well. By analyzing how each cell interacts with its neighbors, we are able to find the frontier that can provide the most information gain to the system. Thus, to have an estimation of the information potential considering the cell interaction, the IP of each cell is summed with the IP of its direct neighbors. We call this value $IP(c)$, for each cell $c$ of the frontier.

The final step is to discover the total information of a frontier $f$. This is done by adding all interactions calculated on the previous step, as can be seen on Equation 3. Basically, this operation informs how much information can be gained by exploring that particular cell. By scanning the window through the whole occupancy grid, calculating the information potential of the frontiers, it is possible to build an *Information Map*, where each cell value consists of the information evaluated on the window for that cell. Fig. 1 shows an estimated map on (a) and the respective *Information Map* on (b). Green areas indicate regions with higher information a good place to explore.

$$\mathrm{IP}(c) = \mathrm{G}(c) + \sum_{n_i \in N} \mathrm{G}(n_i) \tag{2}$$

where, $N$ is the set of all $c$'s neighbors, $n_i$ is the $i^{th}$ neighbor an $\mathbf{G}(n_i)$ is the Gaussian function value for the cell probability.

$$\mathrm{Inf}(f) = \sum_{c \in f} \mathrm{IP}(c) \tag{3}$$



(a) (b)

**Figure 1: (a) Estimated Map (b) Information Map.**

## 3.2 Distance Factor

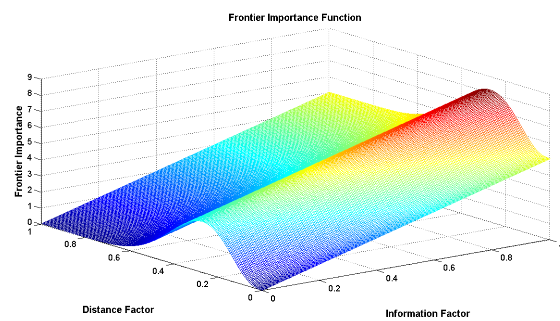
The distance of a frontier to the robot can influence the outcome of the exploration. For instance, if the robot explores only nearby frontiers, it can explore while keeping a good estimation of its localization, but the amount of map exploration percentage gain has the tendency of being low, since the robot has already explored other frontiers close to that one. On the other hand, if the robot chooses to explore a very far frontier, the amount of map exploration percentage tends to be higher, but the robot tends to spend a considerable amount of time navigating through already mapped areas and the localization estimation can be lowered. In order to improve the exploration efficiency, it is interesting to have the possibility to weight the importance of a frontier based on its distance.

To do so, we propose an approach that uses a function that computes a utility based on the frontier's distance. To compute this, a wavefront is started on the map, from the robot towards the frontiers. This computes the distance from the robot to each frontier cell. Then, all frontier cells distances are normalized between [0,1], and each one is given an importance based on a function, described as:

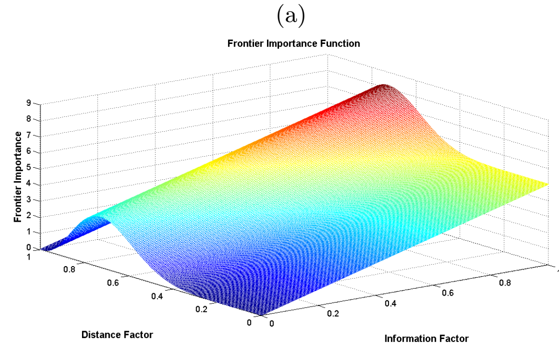
$$\mathrm{Dist}(f) = \mathrm{wavefront}(f)^{\alpha-1} \times (1 - \mathrm{wavefront}(f))^{\beta-1} \tag{4}$$

where, $f$ is a frontier, $Dist$ is the distance factor value for that frontier, $wavefront(f)$ is the normalized distance of the frontier and $\alpha, \beta$ are variables that allow to change the function form. By changing the value of these two variables, we give more importance to a frontier close to the robot or to one that is far away. It is also possible to spread the importance for more distances or to focus on only one.

The use of the distance function together with the information function allows the robot to explore the environment based on both the distance and the information potential of the frontier. When examining both functions together, it is possible to observe the best combination of distance $\times$ information potential. Fig. 2 illustrates two $(\alpha, \beta)$ configurations: on Fig. 2a the robot will prefer to explore closer frontiers and on Fig. 2b more distant frontiers. The information factor is always crescent, since we aim to maximize the information gain.




(a)




(b)

**Figure 2: Utility of a single frontier for (a)** $\alpha = 4, \beta = 9$ **(b)** $\alpha = 9, \beta = 4$**.**

## 3.3 Coordination Factor

So far, the Utility Function chooses the best frontier considering a single robot, but ignoring other robots can reduce the exploration efficiency. Without any coordination, robots can end up exploring the same frontier, which could result on a waste of resources. Since we do not have a centralized coordination, each robot must be able to explore and coordinate itself autonomously. To do so, we propose a coordination strategy that adds a new factor to the Utility Function.

If a robot knows where another robot is, it should choose to explore regions far from the other. To do so, when another robot is spotted, firstly both estimated maps are merged and the Utility Function is calculated for the distance and the information factor, resulting on discovering the frontier with the highest utility. Then, a wavefront, here called *negative wavefront*, is started on the merged map, from the other robot's pose, towards the frontiers.

The main objective of this negative wavefront is to decrease the importance of a frontier that is close to the other robot. The cell where the other robot was detected receives the highest value and as the negative wavefront moves the values start to decrease. Then, the negative wavefront is normalized and its value is subtracted from the utility function of each frontier.

Fig. 3 illustrates the negative wavefront. On Fig. 3a, there is an estimated map with the pose of another robot detected in it. On Fig. 3b, there is the negative wavefront, calculated for that situation, where the brighter the red, the higher the negative value is. And on Fig. 3c there is the utility of each cell, where the closer to green a frontier cell is, the better for exploration it is.

(a) (b) (c)

**Figure 3: (a) The estimated map with the other robot pose (b) The negative wave (c) The resulting importance for each cell.**

## 3.4 Map Merge

When two robots detect each other, they must merge their respective estimated maps. Each robot is equipped with a camera that detects markers located over the robots. This way, when a robot spots a marker, it assumes that another robot was found. There are four markers indicating the front, back and sides of the robots. Thus, when a robot spots another, it can infer the other robot relative position and orientation.

When robot r1 detects robot r2, first, they exchange two pieces of information: their pose $({}^{r1}x_{r1}, {}^{r1}y_{r1}, {}^{r1}\theta_{r1})$ [1] and $({}^{r2}x_{r2}, {}^{r2}y_{r2}, {}^{r2}\theta_{r2})$, and estimated maps map$_{r1}$ and map$_{r2}$. Robot r1 also calculates the pose of r2 $({}^{r1}x_{r2}, {}^{r1}y_{r2}, {}^{r1}\theta_{r2})$ on $map_{r1}$.

Since we have no information about the other robot starting pose, we need to calculate it in order to transform map$_{r2}$ to the same frame of map$_{r1}$. To do so, first we rotate map$_{r2}$ to the same orientation as map$_{r1}$. Basically, we are trying to find an angle $\phi$ that aligns the maps into the same orientation. To calculate $\phi$ we use both robots' orientations (${}^{r1}\theta_{r1}$ and ${}^{r2}\theta_{r2}$) and the relative orientation between robots (${}^{r1}\theta_{r2}$).

$$\phi = {}^{r1}\theta_{r1} + {}^{r2}\theta_{r2} + {}^{r1}\theta_{r2}. \quad (5)$$

Every robot's starting pose is defined at the center of the map, so, after rotating map$_{r2}$ by $\phi$, we need to translate it, so that both maps are on the same origin frame. To do so, we use the pose where we detected r2 $({}^{r1}x_{r2}, {}^{r1}y_{r2})$ and r2's pose $({}^{r2}x_{r2}, {}^{r2}y_{r2})$ to calculate a translation matrix that is applied to map$_{r2}$.

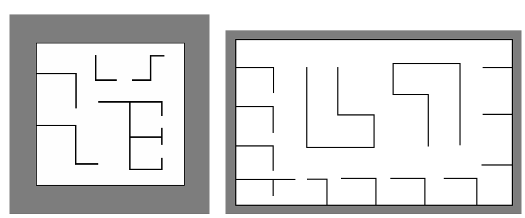
Now that we have the map$_{r2}$ translated and rotated, every cell on map$_{r2}$ corresponds to the same cell on map$_{r1}$. Thus, to merge the maps, all that we need to do is to stitch map$_{r2}$ to map$_{r1}$. Since r1 have no other information about r2, but its map and pose, and these informations can be highly uncentain, r1 have higher trust on its own map. Being so, the map stitching is done by replacing only unknown cells on map$_{r1}$ by correspondent non-unknown cells on map$_{r2}$.

# 4. EXPERIMENTS AND RESULTS

In order to evaluate our approach, we performed a series of experiments using simulated and real robots.

## 4.1 Simulated Experiments

To run the simulated experiments we used ROS/Gazebo, simulating two Pioneer-3AT with a Sick laser range finder. We used two different environments shown on Fig. 4. On Fig. 4a the environment has $394m^2$ and will be called **simple environment** and on Fig. 4b the environment has $812m^2$ and will be called **complex environment**. Our



(a) (b)

**Figure 4: (a) Simple Environment (b) Complex Environment.**

approach was compared with three other. The first one is the classical Near-Frontier Exploration (NFE) adapted to a multi-robot system with a simple coordination strategy:

[1] For instance, ${}^{r1}x_{r1}$ means the $x$ pose of robot 1 on robot's 1 map frame and ${}^{r2}x_{r1}$ means the $x$ pose of robot 1 on robot's 2 map frame.

when robots met, they would avoid exploring frontiers near each other. The second one uses a utility function that calculates the information gain of a frontier using the entropy of the occupancy grid cell probability of being occupied, similar as [2, 12] and the coordination proposed on Section 3.3. And, the third approach is MinPos [1], that guides the robots to a frontier, only if it is the closest one to that frontier. This approach uses a shared map and every robot knows each other positions.

In order to compare the approaches, we used three metrics: i) the exploration time, ii) the quality of the estimated map and iii) the map exploration percentage by time. The exploration time measures how long it takes for the robots to explore the whole environment. The exploration is finished when one robot can not identify any valid frontier to explore. The map quality metric is based on the work of Juliá et al. [8]. First, the estimated map is aligned with a ground truth map of the environment. Then, the maps are subtracted from each other and their differences are counted. This metric indicates the number of cells that have errors on the estimated map, so, the smaller the value, the better is the map quality. The map exploration percentage by time metric evaluates the map exploration percentage by the exploration time. We only consider that a cell is fully explored if its value is either 0, which means that it is a free cell, or if its value is 1, which means a occupied cell. Cells with uncertainty are not considered for the exploration percentage. We use the size of the ground truth map to determine the exploration percentage.

In the experiments, we varied the distance factor to reflect different importance of frontier's distance, by altering the values of $\alpha$ and $\beta$. Both values were varied from 1 to 9 in order to prioritize different distances. Since this generates too much information and it would be difficult to show all results, only the three best $(\alpha, \beta)$ configurations are shown here. The best configurations are the ones with the best pair (map quality x exploration time) indicator. Every configuration was tested 20 times, with random start locations for the robots, and the results presented here are the average of these 20 runs.

### *4.1.1 Simple Environment Results*

In the simple environment, the best $(\alpha, \beta)$ configurations were: $(5, 9)$, $(3, 9)$ and $(8, 8)$. Fig. 5 shows where these configurations stand against the other approaches when comparing the pair (map quality x exploration time). The figure shows that the configuration $(5, 9)$ produces better maps and is faster than the NFE, while $(3, 9)$ is even faster than $(5, 9)$, but has a slightly worse map quality. Configuration $(8, 8)$ produces the better quality maps, but is also slower than the NFE. MinPos and Entropy were both slower than the others.

Regarding the information gain of the system, Fig. 6 shows how each approach behaves during the exploration. The curves presented on this figure are the average of both robots information. It is interesting to observe that the configuration $(8, 8)$ increases the information gain on an approximately constant rate. It can be explained by the fact that this configuration balances equally the distance factor and the information factor, resulting in a smoother exploration.

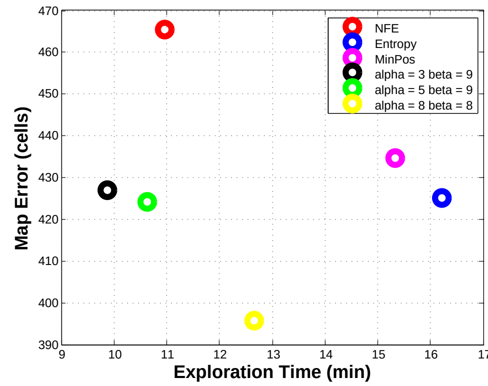



**Figure 5: Exploration time $\times$ Quality for the simple environment.**

The other two configurations, the NFE and MinPos prioritize exploration over map quality, which causes the estimated map to have a lot of cells with uncertainty. Since we only count as mapped cells that have probability either 0 or 1, this results on the map exploration percentage increasing slowly. When the robots merge their maps, cells with uncertainty are resolved, which causes big jumps on the exploration percentage. By using only Entropy in the utility function, we are performing a greedy exploration based only on the information gain, disregarding the distance to a frontier. This can lead the robot to continuously navigate through already visited areas, increasing the exploration time.

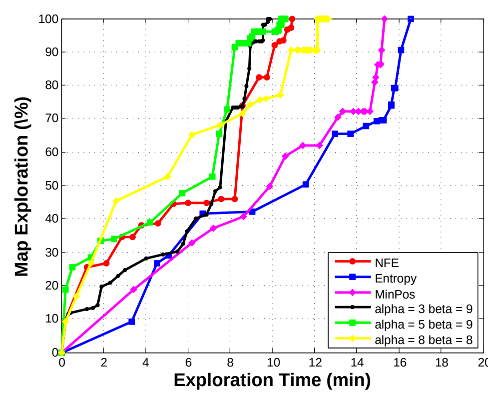



**Figure 6: Information Gain by the exploration time on the simple environment.**

These results indicate that exploring frontiers that are not too close, nor too far from the robot is better than the closest frontier, if the priority is exploration time. If the priority is map quality, it is better to explore frontiers on the medium distance. They also show that it is better to focus on one distance than spreading the importance to various distances.

### *4.1.2 Complex Environment Results*

For the complex environment, the best $(\alpha, \beta)$ configurations were $(5, 9)$, $(5, 5)$ and $(8, 8)$. Fig. 7 demonstrates where these configurations stand against the other approaches when comparing the pair (map quality x exploration time). The figure shows that the configuration $(5, 9)$ produces better maps, but configuration $(8, 8)$ is faster.

As this environment is larger, the robots encounters are reduced. This explains why the other configuration, $(5, 9)$, and the NFE are slower than the $(8, 8)$ configuration. Since the robots encounters are more sparse they have less chances to resolve the cells with uncertainty, which causes them to have a longer exploration time. This also explains why MinPos was the second fastest. Since it has a global map, the number of encounters does not influence the exploration time. Regarding the information gain of the system, Fig. 8 shows

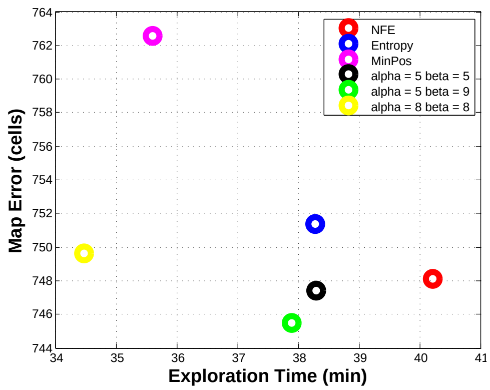



**Figure 7: Exploration time $\times$ Quality for the complex environment.**

how each configuration and the other approaches behave during the exploration. The curves presented on this figure are the average of both robots information. It is interesting to observe that configurations $(8, 8)$ and $(5, 5)$ have the same behavior as the $(8, 8)$ configuration had on the simple environment, balancing the distance factor and the information factor. But, the $(5, 5)$ configuration is slower because it spreads the importance for more distances instead of focusing on one, which can make the robot go to closer or farther frontiers also. These results indicate that, for large environments, exploring the medium distance frontier is better than closer or farther frontiers. It also informs that it is better to focus on one distance than spreading the importance to various distances.

## 4.2 Real Experiments

We also performed experiments with real robots to test the effectiveness of our system. These tests were performed inside our laboratory using two Pioneers 3AT equipped with sick laser range finder and a webcam. We used ROS framework to the control and communication of the robots.

The main goal of the real experiments was to test the performance of the map merging together with our exploration approach. Since the environment is small and the results on Sec. 4.1.1 indicates that exploring farther frontiers is recommended for small environments, we used $\alpha = 9$ and $\beta = 4$ for the distance cost function. Fig. 10 shows one experiment where two robots started on opposite points of the laboratory, not seeing each other. Fig 10a depicts the map of robot 1 and its starting position. Fig. 10b illustrates the same as Fig. 10a, but for robot 2. Fig. 11 illustrates

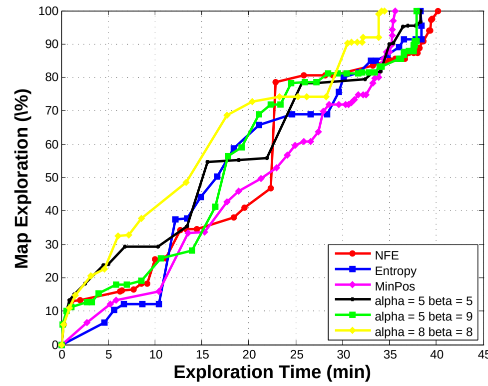



**Figure 8: Information Gain by the exploration time on the complex environment.**

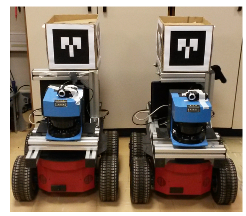
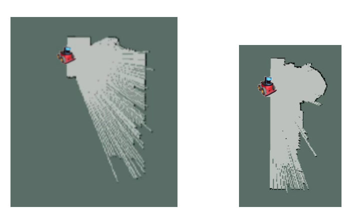


**Figure 9: Real robots set up.**

progress of the exploration for robot 1 point of view. On Fig. 11a and 11b robot 1 is starting to explore. Its trajectory is represented by the blue arrows. In Fig. 11c robot 1 detects robot 2. Since both maps are on the robots' frame, we rotate and translate robot's 2 map to merge with robot's 1 map. Fig. 11d depicts the resulting map. At this point, robot 1 evaluates the utility of the frontiers on the new merged map and decides to explore the frontier marked with a red circle on Fig. 11e. It is interesting to note that all three factors (information, distance and coordination) conducts the robot to the chosen frontier. Fig. 11f is the final map, when robot 1 decides there is no other frontiers to explore. Robot 1 did not had to explore the whole environment to successfully map it. The information added by merging robot's 2 map into its own was enough to complete the bottom part of the map.

# 5. CONCLUSIONS AND FUTURE WORK

This paper presented a novel approach to the multi-robot



(a) (b)

**Figure 10: Real robots experiments: (a) robot 1 initial map and (b) robot 2 initial map.**

(a) (b) (c) (d) (e) (f)

**Figure 11: Real robots experiments. Blue arrows indicates robot's poses and orientations through time. In (a) and (b) robot 1 is starting to explore. On (c) it detects robot 2 and on (d) merges its own map with robot's 2 map. In (e) robot 1 decides to explore a frontier on the top left part of the map, marked by a red circle. And in (f) the final map of the environment.**

exploration task. By combining both planning and coordination of robots on a single Utility Function, robots are able to explore an environment individually, with low information exchange, and still be able to coordinate themselves. The utility function combines a factor for measuring information gain, a factor that takes into consideration the distance cost and a factor for coordination of robots.

We presented a novel approach to measure information gain on exploration based on the information potential of the occupancy grid cells. We also proposed a novel function for measuring distance cost to frontier exploration that can be modified to be used on different scenarios. And we proposed a coordination strategy that is completely decentralized, resulting on a fast coordination and low information exchange.

We compared our approach with three other approaches. Results show that our approach can improve the multi-robot exploration in terms of map quality, exploration time and information gain. We also tested our system on real robots to ensure it can be used on a real scenario.

For future work, we intend to test our approach on different environment sizes, with more than two robots. We intend to analyze the possibility of varying the distance cost dynamically to reflect the estimated map state. Also, we intend to analyze the information factor window and the possibility of expanding or reducing its size during the exploration.

## 6. ACKNOWLEDGMENTS

This work was supported by CNPq, Capes and Fapemig.

## 7. REFERENCES


[1] A. Bautin, O. Simonin, and F. Charpillet. Minpos : A novel frontier allocation algorithm for multi-robot exploration. In *Intelligent Robotics and Applications*, volume 7507 of *Lecture Notes in Computer Science*, pages 496–508. 2012.

[2] W. Burgard, M. Moors, D. Fox, R. Simmons, and S. Thrun. Collaborative multi-robot exploration. In *IEEE Int. Conf. Robotics Automation (ICRA)*, pages 476–481, 2000.

[3] W. Burgard, M. Moors, C. Stachniss, and F. Schneider. Coordinated multi-robot exploration. *Robotics, IEEE Transactions on*, 21(3):376–386, 2005.

[4] J. Faigl, M. Kulich, and L. Preucil. Goal assignment using distance cost in multi-robot exploration. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 3741–3746, 2012.

[5] H. J. S. Feder, J. J. Leonard, and C. M. Smith. Adaptive mobile robot navigation and mapping. *International Journal of Robotics Research*, 18(7):650 – 668, 1999.

[6] D. Fox, J. Ko, K. Konolige, B. Limketkai, D. Schulz, and B. Stewart. Distributed multirobot exploration and mapping. *Proceedings of the IEEE*, 94(7):1325–1339, 2006.

[7] D. Holz, N. Basilico, F. Amigoni, and S. Behnke. Evaluating the efficiency of frontier-based exploration strategies. In *ISR/ROBOTIK*, pages 1–8, 2010.

[8] M. Juliá, Óscar Reinoso, A. Gil, M. Ballesta, and L. Payá. A hybrid solution to the multi-robot integrated exploration problem. *Engineering Applications of Artificial Intelligence*, 23(4):473–486, 2010.

[9] M. Juliá, A. Gil, and O. Reinoso. A comparison of path planning strategies for autonomous exploration and mapping of unknown environments. *Autonomous Robots*, 33(4):427–444, 2012.

[10] S. Kasperski, M. Eich, and J. Wollenberg. Evaluation of cooperative exploration strategies using full system simulation. In *ICAR '13*, 2013.

[11] A. A. Makarenko, S. Williams, F. Bourgault, and H. Durrant-Whyte. An experiment in integrated exploration. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, volume 1, 2002.

[12] C. Stachniss, G. Grisetti, and W. Burgard. Information gain-based exploration using rao-blackwellized particle filters. In *In RSS*, pages 65–72, 2005.

[13] B. Yamauchi. A frontier-based approach for autonomous exploration. In *Computational Intelligence in Robotics and Automation, 1997. CIRA'97., Proceedings., 1997 IEEE International Symposium on*, pages 146–151, Jul 1997.

[14] B. Yamauchi. Frontier-based exploration using multiple robots. In *Second Int. Conf. Autonomous Agents*, 1998.