



HAL
open science

From Utility-Based Action Selection to the Emergence of Communication Chains in a Multi-Robot System

Erwan Martin, Antoine Nongillard, Philippe Mathieu

► To cite this version:

Erwan Martin, Antoine Nongillard, Philippe Mathieu. From Utility-Based Action Selection to the Emergence of Communication Chains in a Multi-Robot System. 18th International Conference on Agents and Artificial Intelligence, Mar 2026, Marbella, France. pp.164-173, <10.5220/0014470900004052>. <hal-05560221>

HAL Id: hal-05560221

<https://hal.science/hal-05560221v1>

Submitted on 20 Mar 2026

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons CC BY-ND 4.0 - Attribution - No Derivative Works - International License

From Utility-Based Action Selection to the Emergence of Communication Chains in a Multi-Robot System

Erwan Martin^a, Antoine Nongaillard^b, Philippe Mathieu^c
Univ. Lille, CNRS, Centrale Lille, UMR 9189, CRISTAL, F-59000 Lille, France
{firstname.surname}@univ-lille.fr

Keywords: collective behavior, action selection, multi-robot system

Abstract: Multi-robot systems (MRS) offer an effective alternative to human operation in hazardous or large-scale environments. However, ensuring efficient coordination when communication is constrained remains a significant challenge. This paper introduces an emergent coordination mechanism that enables each robot to make autonomous decisions based on locally acquired knowledge through perception and proximity communications. Every robot relies on a utility function that balances individual and collective efficiency. Unlike existing approaches, our method does not rely on global consensus or environment modification and demonstrates robustness to communication disturbances. We show that imposing specific constraints leads to the emergence of a communication chain towards a base station, and analyze the influence of various parameters on the dynamics of this phenomenon.

1 INTRODUCTION

In certain operational scenarios such as firefighting, search and rescue in hostile environments, or exploration of contaminated areas, the deployment of robotic systems is becoming an indispensable alternative to human operators for reasons of safety and/or performance. The increased capabilities of robots in recent years are making these deployments possible.

However, in very large environments, a single robot cannot effectively cover the space to be explored or respond to the diversity of local constraints on its own. This is why multi-robot systems (MRS) are favored. Nevertheless, the effectiveness of an MRS depends on its ability to coordinate. Traditional centralized coordination approaches are difficult to apply. They increase required computing resources and communication channel congestion exponentially grows as the number of robots increases (Verma and Ranga, 2021; Wang et al., 2025). In addition, most current distributed methods require a consensus phase to validate task allocation. The real-world environment in which robots are deployed can degrade the communication necessary for this consensus, causing a decrease in MRS performance. How can an MRS

maintain effective coordination under degraded or intermittent communication? Can complex behaviors and constraint management emerge without continuous communication?

In order to improve decision-making in an MRS, we propose a cooperation mechanism under communication constraints for MRS in unknown environments. Our goal is to enable emergent cooperation, where each robot makes autonomous decisions based on local knowledge from its perceptions and proximity communications. Our principal contribution is an action selection method that enables distributed coordination without requiring mandatory communication between robots. This approach aims to maintain good mission performance regardless of the quality of the communication channel. In this work, we aim to observe the emergence of a complex behavior by adding specific high priority tasks or constraints to the MRS. A realistic scenario is a MRS exploring a zone for safe and rescue, or firefighting purpose, the users usually want to keep track of the mission execution. Then, robots information has to be synchronized periodically with the base station. In this work, we introduce this constraint to the MRS.

Section 2 provides an overview of the state of the art in research on task allocation and decision-making methods in robotics, while Section 3 presents the proposed model and describes the action selection mechanism. Simulations are detailed in Section 4 where we

^a <https://orcid.org/0009-0003-3770-786X>

^b <https://orcid.org/0000-0001-8551-0509>

^c <https://orcid.org/0000-0003-2786-1209>

observe the emergent cooperative behaviors among agents before concluding in Section 5.

2 RELATED WORK

In our context, coordinating an MRS means determining for each robot, at a given moment, the action it must take in order to maximize the overall efficiency of the system.

Task allocation methods such as (Kuhn, 1955) perfectly address this problem by optimally allocating tasks to robots. However, centralizing this process creates a bottleneck on the central machine that performs it (increasing the number of robots and tasks exponentially increases the computing time required for allocation). This method also requires knowledge of the tasks in advance before distributing them to the robots, which is not the case in our context. It is therefore necessary to distribute the reasoning across the MRS.

To perform exploration missions, (Yamauchi, 1997) proposes a frontier exploration approach where the robot moves towards the closest unexplored parts of the environment. This approach is adapted to MRS in (Burgard et al., 2005) and improved in (Bautin et al., 2012) to allow for better distribution of robots in the environment, leading to faster exploration.

Other approaches (Choi et al., 2009; Heppner et al., 2024) propose a mechanism whereby robots can bid on different tasks and a global consensus validates an allocation. Some approaches elegantly rely on the robots' local perceptions only, by marking the environment as the mission progresses (Ferranti et al., 2007; Gauville and Charpillat, 2022). However, the real environment is not easy to mark in this way.

Reinforcement learning is also used (Sartoretti et al., 2019; Wang et al., 2024) to teach robots the different actions to perform depending on the state of the environment and the mission. This method, requires intensive simulations when a new constraint appears or another use case is settled. Furthermore, it requires robots with the necessary equipment to perform calculations smoothly. Some robots are limited, particularly by their size or loading capacity, and cannot carry such equipment.

Although all of the approaches cited enable distributed coordination in an MRS, they assume perfect communication conditions in the real world, which are rarely met (Benavides et al., 2016; Verma and Ranga, 2021; Wang et al., 2025).

In order to manage communication disruptions, various methods have been studied. The use of semantic data to reduce the size of the data to be sent

was used in (Gauville, 2022). This method reduces the need for bandwidth, but remains sensitive to disconnections. Stigmergy is a good way to completely avoid direct communication (Ferranti et al., 2007; Gauville and Charpillat, 2022). However, it requires the ability to mark the environment, which is not trivial in the real world. Other approaches seek to extend a global network. In (Agha et al., 2022), robots physically deploy communication nodes in order to extend communication coverage and maintain a centralized architecture as much as possible. In (Petráček et al., 2021), the robots themselves form a communication chain to extend network coverage. In (Fleuriel et al., 2025), robots use light communications while remaining in chains. These approaches maintain a certain level of connectivity but require robots to remain close to each other. This prevents the MRS from dispersing throughout the environment for increased efficiency. "Rendezvous" strategies are introduced in (Tellaroli et al., 2024), where robots calculate rendezvous points in advance in the environment. However, this approach is ineffective in unknown environments.

MRS coordination often requires a compromise: either sacrificing robot autonomy to maintain centralized coordination, or accepting latency or efficiency losses. If communication conditions deteriorate significantly, performance will decline proportionally. Our contribution aims to mitigate this decline in performance.

Centralized approaches face scalability issues and rely on a single entity, while distributed methods require perfect communication or consensus. In contrast, our mechanism enables the MRS to accomplish its mission without mandatory communication. Furthermore, unlike stigmergy-based methods, it does not rely on explicit modification of the environment, which facilitates its deployment in real-world contexts. Finally, unlike behavioral approaches such as ALLIANCE (Parker, 2002) or recent works on distributed allocation under communication constraints (Matignon et al., 2012; Benavides et al., 2016; Chen et al., 2019), our contribution introduces a novel utility function. This function integrates both individual efficiency and team contribution, enabling:

- Autonomous and rapid action selection,
- Robustness to communication disruptions,
- Support for a variety of tasks.

The proposed method is detailed in the following section.

3 PROPOSED METHOD

Our main contributions are:

- A novel decentralized coordination mechanism for multi-robot systems.
- An action selection strategy leveraging a shared utility function that balances individual and collective efficiency.
- Experimental validation showing the emergence of robust communication chains and analysis of system performance under varying parameters.

The goal is to enable each robot to make decisions autonomously based on the local knowledge it has accumulated. It must do so by taking into account not only how efficiently it can perform an action individually, but also the ability of other members of the fleet to perform the same action. In this section, we describe the environment, the robots, and the various tasks they can perform before detailing our action selection mechanism.

3.1 Environmental Modeling

Let E be an environment defined by its size $[w, h]$, divided into cells $c \in E$. Each cell is associated with a value from the set B of possible terrain types (e.g. $\{0 : \text{free}; 1 : \text{obstacle}; 2 : \text{water}; \dots\}$).

3.2 Tasks Modeling

Let η be a set of different types of tasks that a robot can perform (e.g., explore, act as a communication relay, clean up waste, etc.). Let η_i denote a specific type of task where one or more actions a can be identified. For example, in the specific case of exploration, which we denote η_E , each action $a \in \eta_E$ can correspond to a specific frontier between explored and unexplored area of the environment.

3.3 Robots Modeling

In this environment, there is a set of robots \mathcal{R} , equipped with sensors that allow them to perceive their environment in order to build a map of it. We do not study SLAM (Simultaneous Localization and Mapping) because we consider that each robot has access to its own position and that the measurements it takes of the environment are accurate.

In this section, we detail the intrinsic characteristics of robots, which don't change during the mission, and then we model the changing values of robots.

Characteristics: Depending on its equipment, each robot is characterized by: a maximum speed

$v_{max} = (x'_{max}, y'_{max}, \omega'_{max})$ containing respectively the maximum linear speeds in x and y and angular speed on the z axis. A perception range per , a communication range com , a communication period τ , and an ease capacity $\mu : \eta \rightarrow [0, 1]$, allowing the robot to know its effectiveness in performing a specific task in η . We also define the robot's mobility $\phi : B \rightarrow [0, 1]$, which determines how easily a robot can move on a specific type of terrain included in B .

At each moment t , the state of each robot is defined by $r^t = (\theta^t, v^t, I^t, O^t)$, where $\theta^t = (x, y, \omega)$ corresponds to the transformation of the robot, grouping its position in x and y and its rotation in z . $v^t = (x', y', \omega')$ represents its linear and angular velocity in the same dimensions. $I^t = \{\cup_{r_i \in \mathcal{R}} (\theta_{r_i}, \mu_{r_i}, \gamma_{r_i}^t)\}$ represents the knowledge of a robot belonging to \mathcal{R} at time t about its transformation θ , its capabilities μ and a timestamp γ^t representing the obtaining time of the information. $O^t : (x, y)^{h \times w} \rightarrow (B, \mathbb{N})^{h \times w}$ is the local and partial representation of the environment E of robot r , where each cell is associated with a pair containing an environment type value in B and a timestamp γ^O .

Each robot continuously updates its own information in I_r^t and updates O_r^t based on its perceptions. When robot r_1 encounters robot r_2 and they are able to communicate, they exchange their knowledge. Formally, when r_1 receives knowledge from r_2 , it performs the following:

$$O_1^t(x, y) = \begin{cases} O_1^t(x, y) & \text{if } \gamma_1^O(x, y) > \gamma_2^O(x, y) \\ O_2^t(x, y) & \text{else} \end{cases} \quad (1)$$

$$I_1^t(r_i) = \begin{cases} I_1^t(r_i) & \text{if } \gamma_1^I(r_i) > \gamma_2^I(r_i) \\ I_2^t(r_i) & \text{else} \end{cases} \quad (2)$$

With $I_1^t(r_i)$ being the information from robot r_1 about robot r_i and $\gamma_1^I(r_i)$ being the timestamp associated with the information $I_1^t(r_i)$.

Let $\mathbf{M} \in \mathbb{N}^{n \times n}$ be the matrix where each entry \mathbf{M}_{ij} represents the most recent timestamp at which robot i received information from robot j . By construction, \mathbf{M} is symmetric, and its diagonal entries \mathbf{M}_{ii} are close to the current system time t , reflecting the fact that each robot has up-to-date information about itself.

When two robots r_1 and r_2 communicate, they merge their respective matrices \mathbf{M}^{r_1} and \mathbf{M}^{r_2} by taking the element-wise maximum of the timestamps, thereby retaining only the most recent information for each pair of robots. This merging process ensures that each robot can infer which team members possess up-to-date information about any other member of the team.

3.4 Action Selection

All robots in \mathcal{R} perform the same action selection mechanism in parallel and individually. Behavior is homogeneous in the MRS.

For each action $a \in \eta_i$, we compute a utility $u(a)$ representing the estimated impact on the mission if the robot performs the action \mathbf{a} . The objective is then to choose the action a^* associated with the greatest utility, considering all actions for all types of available tasks. Formally:

$$a^* = \operatorname{argmax}_{\forall a \in \eta_i, \forall \eta_i \in \eta} (\alpha_i u(a)) \quad (3)$$

Where α_i is a parameter of importance relative to task type η_i . This parameter can be fixed or dynamic. For example, a communication task may become more important during a mission as time goes, in order to encourage the agent to chose a communication action. This unified representation of different types of tasks allows us to consider all actions that a robot can perform.

For a robot r_x performing an action selection, the calculation of the utility u of an action a is formalized as follows:

$$u(a) = \frac{i(a)}{s(a)} \quad (4)$$

Where $u(a)$ represents the estimate of the agent's contribution to the group by performing action a , taking two elements into account:

- The individual utility i represents how comfortable robot r_x is individually with performing action a . It is denoted as:

$$i(a) = \frac{\mu_r(a)}{C_r(a)} \quad (5)$$

With $\mu_r(a) \rightarrow \mathbb{R}$ the robot's ability to perform the action of type η and $C : (\mathcal{A}, \theta, \phi) \rightarrow \mathbb{R}$ a cost function (which could be, for example, the distance of the robot from the action) that takes into account the robot's mobility ϕ in different types of environments, as well as its transformation θ .

- The collective sufficiency s represents the ability of the rest of the fleet \mathcal{R} to perform the task under study without the intervention of robot r_x . It is expressed as:

$$s(a) = \max_{r \in \mathcal{R} \setminus \{r_x\}} \left(\frac{\mu_r(a)}{C_r(a)} \right) \quad (6)$$

Each element μ_r and C_r , associated with the different robots in the fleet, comes from the knowledge I'_x about the other robots and O'_x about the known environment of robot r_x that performs the action selection calculation.

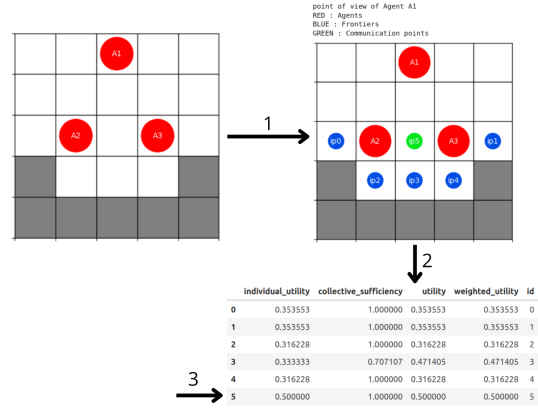


Figure 1: Example of the action selection process: Consider three homogeneous robots capable of performing two types of tasks η : exploration and communication. From the perspective of robot "A1," all types of tasks η have an importance α equal to 1 for the example. Step 1 consists of identifying points of interest: the exploration frontiers are shown in blue and the movements for communication purposes are shown in green. Step 2 consists of calculating the utility of each of these points in relation to current knowledge. Finally, step 3 consists of choosing the point with the highest utility. In our example, robot "A1" is going to move within communication range in order to exchange knowledge.

Such a utility function allows us to take into account the individual efficiency of the robot and the capacity of other members of the fleet to perform an action. This allows for assigning an estimated utility value to each action. Depending on its position, the position of other members of the fleet, and the different efficiencies of each, the robot won't necessarily choose the same action in two different configurations. It is important to note that this calculation is based on the local knowledge I'_x and O'_x of robot r_x at a given time t , which means that the information available to the robot may be incomplete or obsolete. The utility calculated by a robot is only an estimate of the impact of an action based on the knowledge it possesses.

In summary, a robot chooses the action where it is most useful to the mission while avoiding redundancy with its peers. Figure 1 displays an example illustration of the action selection process. More implementation details can be found in (Martin et al., 2026).

In the next section, we propose a series of experiments to validate the performance of our method and the possibility of obtaining a complex behavior by adding constraints on it.

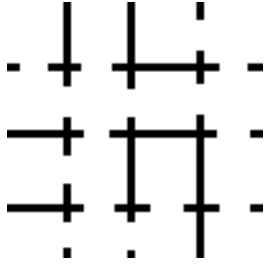


Figure 2: environment: labyrinth

4 EXPERIMENTS

In this section, we begin by detailing the experimental protocol used to implement and constrain the emergent coordination mechanism. We then present experiments in which agents must periodically synchronize with a base station, demonstrating the emergence of a communication chain within the MRS. We analyze the impact of this constraint on both the mission execution and the base station’s knowledge updates, considering various team compositions within the MRS. Finally, we investigate the influence of key parameters on the emergent behavior of the MRS, providing information on their effects on system performance.

4.1 Implementation

An exploration mission is carried out in an empty environment and in a “labyrinth” environment illustrated in Figure 2. As we work with different kinds of robots, some aerial robots can pass above the walls. The labyrinth environment permits to showcase the differences of the robots in terms of capabilities.

In this context, we introduce two different types of tasks η for the action selection method that are settled by default in any mission in an unknown environment:

- **Exploration** : Robots must discover their environment, with each identified frontier between the explored and unknown environment corresponding to a possible action. The frontiers are grouped into clusters, matching the size of the robots’ field of perception, in order to reduce the number of actions to be processed as in (Bautin et al., 2012).
- **Communication** : In order to update its knowledge or retrieve new information, a robot may choose to intentionally position itself within communication range of another robot (which is currently outside its communication range). Several conditions must be met for the robot to identify one or more of its peers as a point of interest for communication: (i) The robot must not be within the current communication range. (ii) The known position of the robot must not be too

obsolete. Formally, to identify a robot r_x as a point of interest for communication, the condition $\gamma^l(r_x) < t - l$ must be met (l being the obsolescence time, a constant dependent on the problem, $\gamma^l(r_x)$ the last timestamp associated with current knowledge about r_x , and t the current time in the mission).

Robots therefore have a choice between two different types of tasks and must coordinate with each other to carry out the overall mission efficiently. The importance of the communication task is dynamic for a robot, and increases as long as it does not communicate with any other robot.

To introduce an additional task, an artifact is placed in the environment. Each artifact is characterized by:

- a position (x, y) ,
- a type, which defines the associated task,
- an *interact()* function, enabling robots to engage with the artifact upon reaching it.

Robots are assumed capable of detecting and localizing artifacts within their sensory range. The task type associated with an artifact must be implemented in the robot’s control architecture, with a predefined capacity μ and an initial importance α . However, robots have no *a priori* knowledge of the number or locations of artifacts in the environment. They can only interact with detected artifacts, as long as their capacity μ is non-zero for the corresponding task.

Additionally, artifacts implement a *check_importance()* method, which dynamically adjusts the task importance α based on contextual factors.

The *BaseStation* is modeled as a fixed member of the multi-robot system (MRS), acting as a purely communicating agent (Ferber, 1995). An artifact of type *base_station_com* is associated with the *BaseStation* to define a dedicated communication task.

The importance α_{bs} of this task for each robot is dynamically updated using the *check_importance()* function of the *BaseStation* artifact, according to the following rule:

$$\alpha_{bs} \leftarrow \frac{(\Delta_{com\ max} - env_size)^3}{t^2} \quad (7)$$

where $\Delta_{com\ max}$ represents the maximum communication delay perceived by the *BaseStation*:

$$\Delta_{com\ max} = t - \min(\mathbf{M}[r_{bs}, :]) \quad (8)$$

Here, $\mathbf{M}[r_{bs}, :]$ denotes the row of the matrix \mathbf{M} corresponding to the *BaseStation* agent r_{bs} , and $\min(\mathbf{M}[r_{bs}, :])$ identifies the oldest timestamp in this row. The importance α_{bs} thus depends on the robot’s

local knowledge of the longest synchronization delay in the system, as encoded in its local matrix \mathbf{M} of last information.

To enforce the constraint, the importance α_{bs} of the communication task is artificially amplified in the robot’s local perception. This importance is further divided by the current time t (measured in simulation steps in our experiments implementation) to incentivize robots to explore the environment over time. Without this adjustment, robots could remain indefinitely constrained near the *BaseStation*, limiting their ability to fulfill other mission objectives. The variation of our parameter α_{bs} is ruled by an understandable formula that shape our robots behavior regarding to the task.

The different robot models used in the experiment are described in detail in Table 1. Ground and Aerial robots have similar maximum speeds, perception, and communication characteristics. The differences between those two robots appear in their capabilities. As the BaseStation agent is fixed, all of its maximum speed is set to zero. Its perception range is extremely low, and the range of communication is lower than that of other agents.

Table 1: Robot model specifications

Name	V_{max}	per	com
Ground	(1, 1, 0.5)	10	20
Aerial	(1, 1, 0.3)	10	20
BaseStation	(0, 0, 0)	2	10

We first evaluate the performance of our system by measuring the time spent to make the full exploration with the mean value at each time of $\min(\mathbf{M}^{r_{bs}}[r_{bs}, :])$ (the local perception at each simulation step of the oldest communication among every other agents of the MRS). The MRS has to explore the environment while periodically synchronizing information with the base station. We analyze how varying the number of robots in the MRS and the proportion of robot models in a heterogeneous fleet impacts both performance and action selection. Those experiments are made in the “empty” and “labyrinth” environments. The robots’ capabilities with regard to the different types of tasks and their ease of movement on different types of terrain are detailed in Table 2.

4.2 Base Station Communication Constraint Impact

4.2.1 Homogeneous Teams

We first ran the simulations with homogeneous teams of different numbers of Ground robots. Figure 3

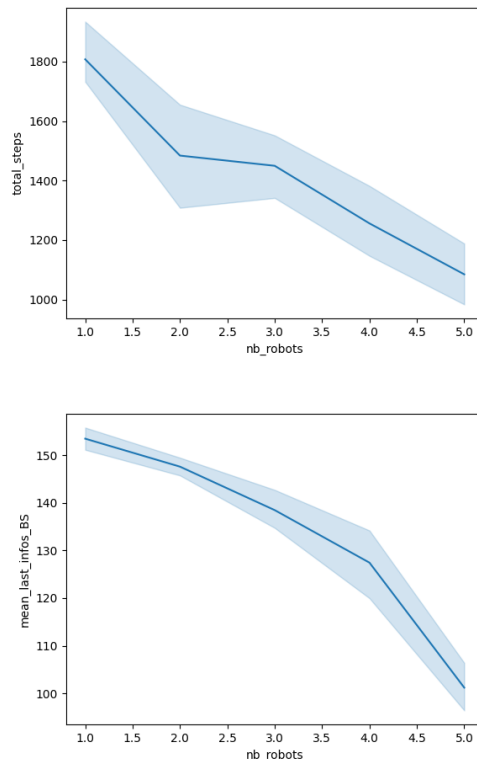


Figure 3: Performance of the system for a team of different robot numbers in an empty environment: In the top graph, the y-axis represents the total number of simulation steps required to complete the mission. In the bottom graph, the y-axis shows the mean of the oldest communication timestamps between the base station and all agents at each simulation step, reflecting the maximum synchronization delay in the system. Ten simulations had been made for each configuration, resulting in the observable standard deviation.

shows the performance of the system depending on the number of robots in an empty environment, while Figure 4 exhibits its performance in the “labyrinth” environment. As expected, the mission completion time decreases when there are more agents involved.

The decrease of the maximum synchronization delay in the system as the amount of robots increase shows the capacity of the members of the fleet to organize in order to better communicate and synchronize information with the base station. What’s observable directly on the simulation is the creation of a communication chain, propagating the agents information to the Base station, before breaking, letting the agents explore again. The synchronization delay being weak, the importance α of the base station communication task decreases, leading the agents to prefer exploration again. A screenshot of our custom-made simulator is displayed in Figure 5, exhibiting a communication chain.

Table 2: Robot capabilities in terms of terrain types and task types

Nom	ϕ		μ		
	floor	obstacle	Exploration	Communication	base_station_com
Ground	1	0	1	1	1
Aerial	1	1	1	1	1

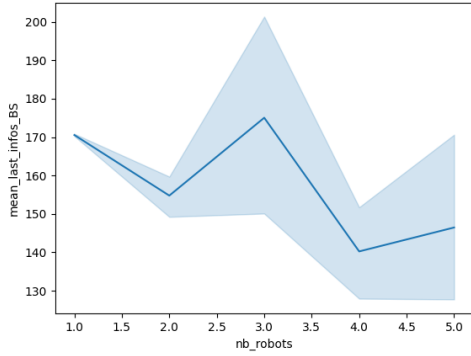
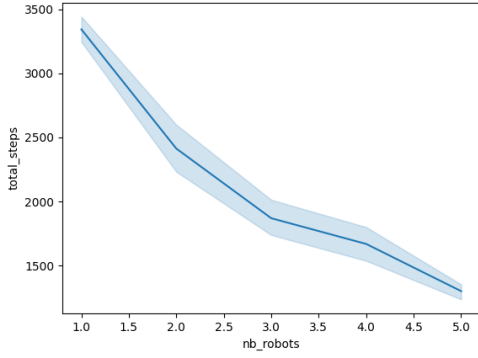


Figure 4: Performance of the system for a team of different robot numbers in an Labyrinth environment: metrics are identical to those in Figure 3.

4.2.2 Heterogeneous Teams

The composition of the fleet, composed of one base station and four other robots of type either Ground or Aerial, can vary. It changes the robots behavior, and hence, the performance of the system. As the differences in terms of capabilities of the robots mainly depend on the obstacles, experiments are only conducted in the "labyrinth" environment in this Section. In Figure 6, we can observe the performance of the MRS with the mission completion depending on the team combination. As Aerial robots are moving more easily in the environment, we could expect the best performance from a team exclusively composed of aerial robots (d). However, we can see that a team composed of three Aerial robots and one Ground robot (b) has the best performance on the system.

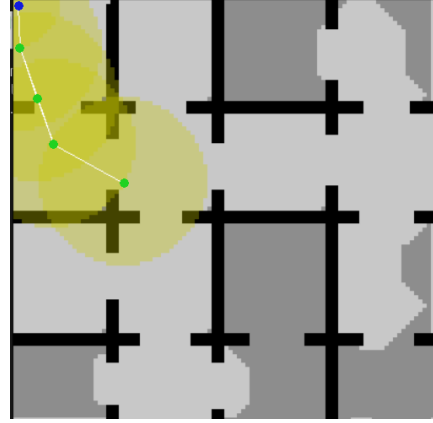


Figure 5: Screenshot of emerging communication chain: In the labyrinth environment, we can see the Ground agents in green, the BaseStation in blue, their communication range in yellow, and their communication links are in white. The dark gray represents the unexplored area of the map.

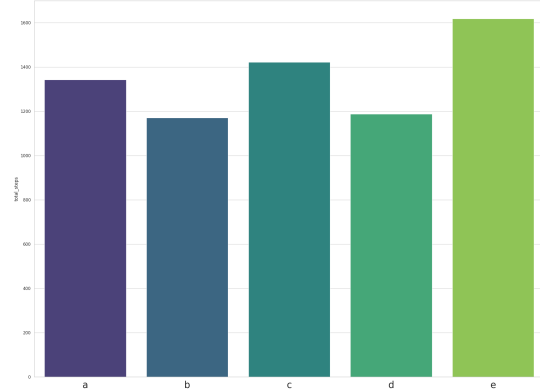


Figure 6: Performance of the system depending on the team composition: On the x axis, we can see letters associated with team compositions as follows : (a) 2 Ground and 2 Aerial, (b) 1 Ground and 3 Aerial, (c) 4 Ground and 0 Aerial, (d) 0 Ground and 4 Aerial, (e) 3 Ground and 1 Aerial. The mean number of steps (as several simulations were conducted per team combinations) is displayed on the y axis.

This can be explained in Figure 7, displaying the proportion of activity for each agent in the team for every composition. In homogeneous teams, composed exclusively of either aerial or ground robots (b and c), tasks are evenly distributed among all team members. However, with heterogeneous teams, as agents take into account locally the capabilities of

the other team members, they chose the most useful task for the MRS. We see that an Aerial agent alone (e) among three ground robots is doing 80% of exploration, while Ground agents balance their exploration and base station communication tasks. When the ground agent is alone (b), it dedicates its time more easily to being part of the communication chain, leading to an easier information synchronization for the Aerial agents that make a faster exploration.

4.3 Influence of Parameters

The configuration and design of the multi-robot system significantly affect the overall mission outcomes. In this section, we analyze the impact of several key parameters on system performance. Specifically, we focus on the *return priority parameter* λ , which is integrated into the base station communication task as follows:

$$\alpha_{bs} \leftarrow \frac{(\lambda \Delta_{commax} - env_size)^3}{t^2}.$$

Two additional parameters are introduced:

- the *replanning delay* Δ_{replan} , defining the time between receiving new information from a teammate and the subsequent replanning;
- the *communication period* τ , representing the interval between two consecutive communications for a given robot.

These parameters collectively influence the speed at which information propagates through the communication chain and the responsiveness of agents to newly acquired information. The simulations were conducted using a team of four "Ground" robots in the "labyrinth" environment.

Figure 8 illustrates the mean maximum synchronization delay of the base station as a function of the return priority parameter λ . The results demonstrate that the synchronization delay can be effectively controlled by adjusting λ . This implies that modulating the importance α of a task enables precise shaping of robotic behavior. Specifically, increasing the return priority parameter λ reduces the mean synchronization delay. While λ is expressed here as an integer priority, it could alternatively be defined in time units to represent explicit temporal constraints.

Figure 9 illustrates the effects of the parameters Δ_{replan} and τ on system performance, measured in terms of both the number of steps required to complete the simulation (Figure 9b) and the mean maximum synchronization delay of the base station (Figure 9a). In Figure 9a, we observe that lower values of τ consistently result in reduced synchronization delays with the base station. Conversely, Δ_{replan} exhibits

a limited impact on this metric. While higher values of Δ_{replan} cause agents to remain near the base station for longer periods before replanning, they continue to communicate with it, thereby maintaining a lower mean maximum synchronization delay. In contrast, when Δ_{replan} is low, agents respond more rapidly to communications from other fleet members, reducing their presence near the base station.

Figure 9b reveals that, in general, lower values of τ and Δ_{replan} accelerate the overall mission completion time. However, if both parameters are excessively low, agents engage in constant replanning, which impedes their ability to organize and disperse efficiently within the environment. These observations suggest that introducing modest delays in communication or replanning can enhance overall system performance.

4.4 Discussion

The proposed mechanism demonstrates how explicit rules for action selection can lead to the emergence of complex phenomena, such as communication chains, in response to the introduction of system constraints. By integrating individual task efficiency with its collective benefit to the fleet, our approach enables the design of self-adaptive robots capable of dynamically assuming distinct roles throughout a mission, depending on contextual demands.

Incorporating the capabilities (μ) and mobility constraints (ϕ) of other robots further fosters the emergence of cooperative behaviors, particularly within heterogeneous teams. Our framework also allows for explicit control over the perceived importance of additional tasks, thereby influencing agent behavior without directly altering the constraints themselves. In essence, we modulate not the constraints, but rather the multi-robot system (MRS) agents' perception of task priority.

However, our experimental validation presents certain limitations. Specifically, we have not evaluated the approach under extreme communication perturbation scenarios (e.g., imperfect or intermittent information transmission within the nominal communication range), nor have we deployed it on physical robots. These aspects represent critical avenues for future investigation.

5 CONCLUSION AND FUTURE WORKS

This study introduces an emergent coordination mechanism for multi-robot systems, where autonomous agents select actions based on local percep-

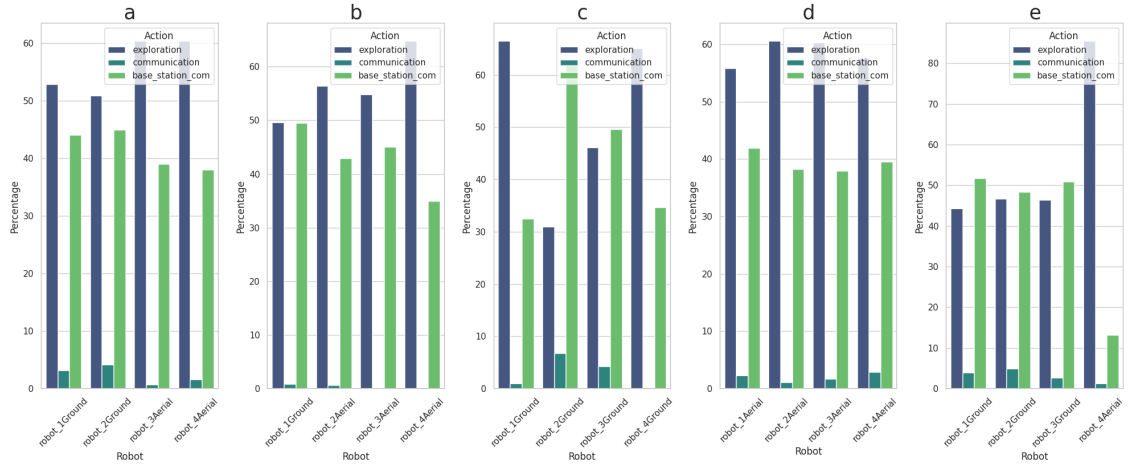


Figure 7: Robots activity depending on the team composition: On the x-axis, we can see letters associated with team compositions as follows : (a) 2 Ground and 2 Aerial, (b) 1 Ground and 3 Aerial, (c) 4 Ground and 0 Aerial, (d) 0 Ground and 4 Aerial, (e) 3 Ground and 1 Aerial. The y-axis represents the percentage of different task types in the total agents' activity.

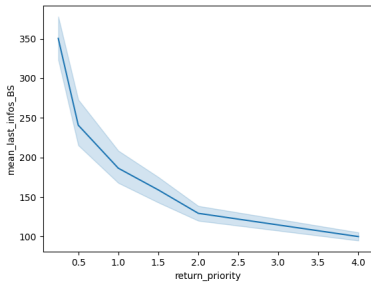
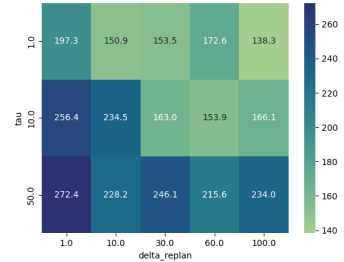


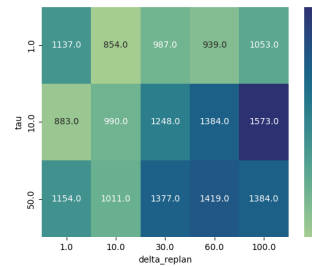
Figure 8: Mean maximum synchronization delay of the base station, depending on the return priority parameter λ .

tions, individual capabilities, and partial fleet knowledge, without requiring global consensus. Our results demonstrate that this approach enables self-adaptive behavior: robots dynamically adjust their decisions in response to varying fleet compositions and contextual constraints. By modulating task importance, we show that complex collective behaviors, such as communication chains, can emerge as a response to system-level constraints. This mechanism highlights the potential of task prioritization to shape global outcomes while preserving agent autonomy. However, the current work is limited to simulated environments with idealized communication conditions, leaving open questions about robustness in real-world scenarios.

Future research will explore the integration of additional constraints, such as energy management and task interdependencies, to investigate the emergence of more sophisticated collaborative strategies. A particularly promising direction involves enabling robots to autonomously learn and adapt task importance pa-



(a) Mean maximum synchronization delay of the base station



(b) steps needed to end the simulation

Figure 9: Heatmap of the effect of Δ_{replan} and τ parameters variation on the system performance.

rameters (α), thereby enhancing system resilience in unpredictable environments. Validating these mechanisms on physical robots and studying their robustness to failures such as agent loss or communication breakdowns will be critical steps toward operational deployment.

ACKNOWLEDGEMENTS

The authors acknowledge the ANR – FRANCE (French National Research Agency) for its financial support of the SOS project (ANR-23-IAS2-0003).

REFERENCES

- Agha, A., Otsu, K., Morrell, B., Fan, D. D., Thakker, R., Santamaria-Navarro, A., Kim, S.-K., Bouman, A., Lei, X., Edlund, J., et al. (2022). Nebula: Team costar’s robotic autonomy solution that won phase ii of darpa subterranean challenge. *Field robotics*, 2:1432–1506.
- Bautin, A., Simonin, O., and Charpillet, F. (2012). Minpos: A novel frontier allocation algorithm for multi-robot exploration. In *Intelligent Robotics and Applications: 5th International Conference, ICIRA 2012, Montreal, Canada, October 3-5, 2012, Proceedings, Part II 5*, pages 496–508. Springer.
- Benavides, F., Monzon, P., Chanel, C. P. C., and Grampin, E. (2016). Multi-robot cooperative systems for exploration: Advances in dealing with constrained communication environments. In *2016 XIII Latin American Robotics Symposium and IV Brazilian Robotics Symposium (LARS/SBR)*, pages 181–186. IEEE.
- Burgard, W., Moors, M., Stachniss, C., and Schneider, F. (2005). Coordinated multi-robot exploration. *IEEE Transactions on Robotics*, 21(3):376–386.
- Chen, X., Zhang, P., Du, G., and Li, F. (2019). A distributed method for dynamic multi-robot task allocation problems with critical time constraints. *Robotics and Autonomous Systems*, 118:31–46.
- Choi, H.-L., Brunet, L., and How, J. P. (2009). Consensus-based decentralized auctions for robust task allocation. *IEEE transactions on robotics*, 25(4):912–926.
- Ferber, J. (1995). *Les sma: vers une intelligence collective*. Inter Editions, Paris.
- Ferranti, E., Trigonì, N., and Levene, M. (2007). Brick& mortar: an on-line multi-agent exploration algorithm. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pages 761–767. IEEE.
- Fleuriel, M., Vanneaux, E., Filliat, D., and Simonin, O. (2025). Exploration d’environnements souterrains par déploiement et réorganisation d’une chaîne de drones. *JFSMA*, page 109.
- Gauville, N. (2022). *Coordination of autonomous robotic platforms in unknown environments for search and rescue*. PhD thesis, Université de Lorraine, France.
- Gauville, N. and Charpillet, F. (2022). Approche locale pour l’exploration autonome d’environnements inconnus par une flottille de robots. *Revue Ouverte d’Intelligence Artificielle*, 3(5-6):625–643.
- Heppner, G., Oberacker, D., Roennau, A., and Dillmann, R. (2024). Behavior tree capabilities for dynamic multi-robot task allocation with heterogeneous robot teams. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4826–4833. IEEE.
- Kuhn, H. W. (1955). The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97.
- Martin, E., Nongillard, A., and Mathieu, P. (2026). Emerging multi-robot cooperation through action selection robust to communication disruptions. In *Proceedings of the International Conference on Robotics, Computer Vision and Intelligent Systems (ROBOVIS)*.
- Matignon, L., Jeanpierre, L., and Mouaddib, A.-I. (2012). Coordinated multi-robot exploration under communication constraints using decentralized markov decision processes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 26, pages 2017–2023.
- Parker, L. E. (2002). Alliance: An architecture for fault tolerant multirobot cooperation. *IEEE transactions on robotics and automation*, 14(2):220–240.
- Petráček, P., Krátky, V., Petrlik, M., Băca, T., Kratochvíl, R., and Saska, M. (2021). Large-scale exploration of cave environments by unmanned aerial vehicles. *IEEE Robotics and Automation Letters*, 6(4):7596–7603.
- Sartoretti, G., Wu, Y., Paivine, W., Kumar, T. S., Koenig, S., and Choset, H. (2019). Distributed reinforcement learning for multi-robot decentralized collective construction. In *Distributed Autonomous Robotic Systems: The 14th International Symposium*, pages 35–49. Springer.
- Tellaroli, M., Luperto, M., Antonazzi, M., and Basilio, N. (2024). Frontier-based exploration for multi-robot rendezvous in communication-restricted unknown environments. In *2024 IEEE/RSSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5807–5812. IEEE.
- Verma, J. K. and Ranga, V. (2021). Multi-robot coordination analysis, taxonomy, challenges and future scope. *Journal of intelligent & robotic systems*, 102:1–36.
- Wang, C., Yu, C., Xu, X., Gao, Y., Yang, X., Tang, W., Yu, S., Chen, Y., Gao, F., Jian, Z., et al. (2025). Multi-robot system for cooperative exploration in unknown environments: A survey. *arXiv preprint arXiv:2503.07278*.
- Wang, R., Zhang, J., Lyu, M., Yan, C., and Chen, Y. (2024). An improved frontier-based robot exploration strategy combined with deep reinforcement learning. *Robotics and Autonomous Systems*, 181:104783.
- Yamauchi, B. (1997). A frontier-based approach for autonomous exploration. In *Proceedings 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation CIRA’97: Towards New Computational Principles for Robotics and Automation*, pages 146–151. IEEE.