2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

D-MARL: A Dynamic Communication-Based Action Space Enhancement for Multi Agent Reinforcement Learning Exploration of Large Scale Unknown Environments

Gabriele Calzolari, Vidya Sumathy, Christoforos Kanellakis, George Nikolakopoulos

Abstract— **In this article, we propose a novel communication-based action space enhancement for the D-MARL exploration algorithm to improve the efficiency of mapping an unknown environment, represented by an occupancy grid map. In general, communication between autonomous systems is crucial when exploring large and unstructured environments. In such real-world scenarios, data transmission is limited and relies heavily on inter-agent proximity and the attributes of the autonomous platforms. In the proposed approach, each agent's policy is optimized by utilizing the heterogeneous-agent proximal policy optimization algorithm to autonomously choose whether to communicate or explore the environment. To accomplish this, multiple novel reward functions are formulated by integrating inter-agent communication and exploration. The investigated approach aims to increase efficiency and robustness in the mapping process, minimize exploration overlap, and prevent agent collisions. The D-MARL policies trained on different reward functions have been compared to understand the effect of different reward terms on the collaborative attitude of the homogeneous agents. Finally, multiple simulation results are provided to prove the efficacy of the proposed scheme.**

I. INTRODUCTION

Autonomous systems are widely used in the collaborative exploration of large-scale environments, where approaches to address specific situations, concerning public safety and Search and Rescue (SAR) operations, should thoroughly examine the agents' ability to operate independently of human supervision, their overall centralized or decentralized orchestration, as well as their adaptability and reactivity towards different and time-varying operating scenarios. Tackling this problem, through a centralized strategy, is extremely challenging and unreliable, as the mission's success relies on the central node's ability to communicate with other agents. Conversely, using a decentralized framework is intriguing as it allows the agents to make independent decisions and share their knowledge of the environment. When dealing with real-life missions, the assumption that communication is always feasible and for any distance between the agents is unrealistic, while additionally it should be considered that the demand for good communication links could further impair the battery life of the exploring robots. For these reasons, it is essential to have an event-based (when needed and possible) communication strategy so that the information sharing between the agents can significantly improve the execution of the exploration task.

This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation, and by the European Union's Horizon Europe Research and Innovation Program, under the Grant Agreement No. 101119774 SPEAR.

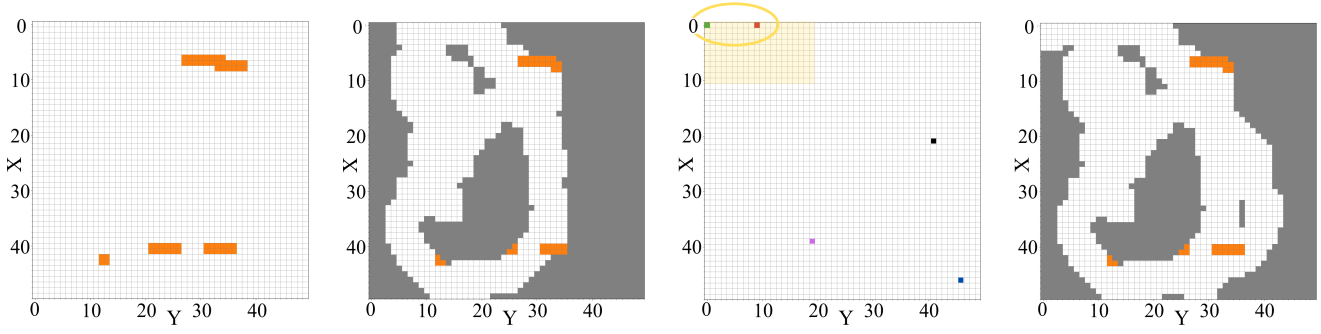
G. Calzolari, V. Sumathy, C. Kanellakis, G. Nikolakopoulos are with the Robotics and AI Group, Department of Computer Science, Electrical and Space Engineering, Luleå University of Technology, Sweden. E-mails: gabriele.calzolari@ltu.se, vidya.sumathy@ltu.se, christoforos.kanellakis@ltu.se, george.nikolakopoulos@ltu.se

A. Related works

Many approaches for implementing multi-agent exploration have been proposed over the years [1]. Among the classical strategies, frontier-based algorithms are ubiquitous [2]–[6]. Nevertheless, most of these methods rely on a central node for assigning agents to various goals, assuming satisfactory communication among these elements. Other papers focus on different non-learning approaches, such as [7] which proposes a novel online coverage algorithm that exploits a new Back Tracking Points (BTP) detection scheme and Boustrophedon motion.

In contrast, the main novelty introduced by learning-based approaches lies in their ability to adapt obtained policies to similar environments, without requiring parameter tuning or modifying the algorithm structure. The method proposed in [8] individuates frontier points and assigns them a cost according to the history of the exploration process. However, some articles have focused on applying reinforcement learning (RL) to multi-agent exploration as in [9]–[11]. In this case, the approach uses RL to implement the policy needed to choose the relevant locations that should be reached through policy search by dynamic programming [12]. Furthermore, [13] exploits dynamic Voronoi partitions to assign different areas to the various robots to avoid overlapping, while [14] proposes a frontier-based method that relies on a centralized training and decentralized architecture (CTDE) and the policies are trained using the multi-agent deep deterministic policy gradient (MADDPG) algorithm. [15] suggests a technique similar to the one mentioned in [14], but it models the environment as a topological graph and proposes H2GNN, which is a cooperative decision-making framework. Finally, [16] explores an approach blending reinforcement learning with target neural networks and the prioritized experience replay to plan the movements and multi-agent collaboration.

Classical approaches are easy to implement but often fail to meet performance constraints, while learning-based methods can enhance performance through experiential data but typically rely on continuous or regular inter-agent communication [17]–[20].



(a) Arena with obstacles (b) Agent's local map (c) Agent's transmission map (d) Agent's global map

Fig. 1: The environment is represented as a 2D grid map, wherein the free, occupied (with obstacles), and undiscovered grids are denoted by white, orange, and grey cells, respectively. The agents are denoted by □ markers with colors red, green, blue, black, and purple. Fig. 1a depicts the completely known arena showing the free and occupied cells. Each agent has its own local and global maps as shown for the red agent in the following sub-figures. Indeed, Fig. 1b shows the red agent's local map. While, the communication network and communication covered area of the red agent are shown in Fig. 1c by the yellow ellipse and shaded area, respectively. Fig. 1d shows the red agent's global map, obtained after blending the local maps of the communicating agents, namely the green and red ones.

In this research and for the first time, to the authors' best knowledge, we are adding communication as an element in the action space so that the homogeneous agents can choose whether to communicate according to a cluster-based approach or to explore. That is accomplished by training the agents' policies through reward functions that consider inter-agent communication and efficient exploration. As such, the novel contributions of this article can be stated as:

- We propose a novel decentralized multi-agent reinforcement learning (D-MARL) architecture with inter-agent communication integrated into the action space and a unique map-based observation space.
- We propose a communication strategy, based on communication clusters, to choose the nearby agents for efficient transfer of knowledge of the environment.
- We propose and investigate the effectiveness of different reward functions to study the efficacy of communication in action space.

II. Methodology

We formulate the multi-agent exploration problem as a Partially Observable Stochastic Game (POSG), defined by $<\mathcal{I}, \mathcal{S}, \{\mu\}, \{\mathcal{A}_k\}, \{\mathcal{O}_k\}, \{r_k\}, \{\mathcal{T}\}>$, since the agents cannot sense the actions chosen by the other agents, where:

- $\mathcal{I} = \{i_k, k \in \mathbb{N} \mid 1 \leq k \leq n\}$ represents the finite set of the n homogeneous agents interacting within the environment. At each time step, an agent occupies a grid cell, detects cells within a rectangular neighborhood of padding r_d, and communicates directly with agents within a communication range r_c.
- $\mathcal{S}$ represents the finite set of states, where each state comprises a global map that includes all agents' discoveries and their current positions.
- $\{\mu\}$ represents the initial state distribution and the probability of one agent being initialized in an occupied cell is zero, otherwise, it follows a uniform distribution.
- $\{\mathcal{A}_k\}$ and $\{\mathcal{O}_k\}$ represent respectively the finite set of actions and observations associated with agent i_k.
- $\{r_k\}$ is the reward function associated with the agent i_k
- $\mathcal{T}$ denotes the probability that taking joint action $\mathbf{a}_t$ in state s_t results in a transition to a new state s_{t+1} and joint observation $\mathbf{o}_{t+1}$.

Fig. 2 shows the investigated architecture, featuring four agents in the exploration arena.

A. Exploration arena and the agent's representation of the surroundings

The exploration arena, as depicted in Fig. 1a, is an $n \times n$ occupancy grid $\mathcal{M}$ with cells of dimension $l \times l$. During exploration, every cell $m_{i,j}$ is assigned a state value from $\mathcal{H} = \{0, 1, 2\}$, indicating whether it is free, occupied, or unexplored. For complete environmental mapping, all free cells must be connected. To achieve the optimal course of action, each agent uses two environment maps and a third map including the agents' positions with which it can directly communicate as illustrated by Fig. 1 and defined as follows:

- *Local map* $\mathcal{M}_{i_k,local}$ is an occupancy grid with a global reference frame that contains only the information gathered from the agent's exploration.
- *Global map* $\mathcal{M}_{i_k,global}$ collects the local observations and the information gathered by the other agents whenever a communication link is established.
- *Transmission map* $\mathcal{M}_{i_k,trans}$ includes the positions of agents reachable via direct communication.

B. Communication strategy

Agents within the same communication cluster can exchange global maps through direct communication if they are within each other's range, or indirectly through a chain of directly communicating agents. Let $\mathcal{C} \subseteq \mathcal{I}$ be a communication cluster, then the merged maps are assigned to the agents' global maps according to Eq. (1).

$$\mathcal{M}_{i_k,global} = \bigcup_{i_j \in \mathcal{C}} \mathcal{M}_{i_j,global} \quad \forall\, i_k \in \mathcal{C} \tag{1}$$

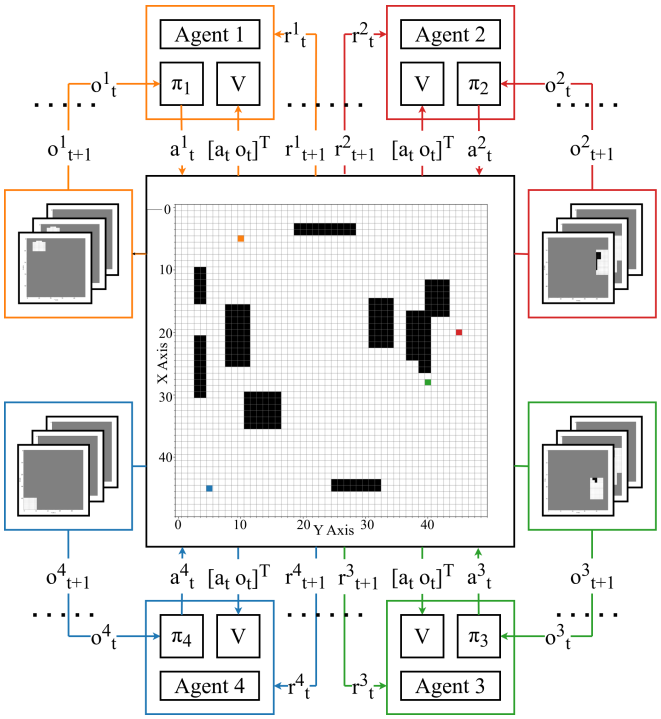


Fig. 2: Proposed decentralized multi-agent reinforcement learning architecture for exploration. The occupancy grid depicts the environment with obstacles (black) and free cells (white), while the positions of the agents are outlined by the colored cells. Furthermore, the components of each agent are shown, namely the agent-based policy (π_k) and the shared critic (V). To simplify the notation, in this figure the agent i_k's action $a_t^{i_k}$, observation $o_t^{i_k}$ and reward $r_t^{i_k}$ at time t have been abbreviated as a_t^k, o_t^k, r_t^k, respectively.

C. Agents' learning algorithm

The policy optimization algorithm Heterogeneous-Agent Proximal Policy Optimisation (HAPPO) algorithm [21] is based on a multi-agent sequential update method. In this algorithm, there are two essential elements for each agent: a shared critic (V) and a non-shared policy (π_k). As indicated by [21], HAPPO works by progressively updating both the parameters of the agents' policies and of the shared critic function for $z \in [1..Z]$ steps, where Z is the number of episodes, each one consisting in T steps per episode. The algorithm iterates over each episode and stores in the replay buffer some samples $\{(s_t, o_t^{i_k}, a_t^{i_k}, r_t, s_{t+1}, o_{t+1}^{i_k}), i_k \in \mathcal{I}, t \in [1..T]\}$ from the trajectories that are generated by the joint policy $\boldsymbol{\pi_{\theta_k}} = (\pi^1_{\theta^1_k}, \ldots, \pi^n_{\theta^n_k})$. Subsequently, a batch of size B is extracted from the replay buffer and the advantage function $\hat{A}(s, \boldsymbol{a})$ is computed using the critic network with GAE and setting $M^{i_1}(s, \boldsymbol{a}) = \hat{A}(s, \boldsymbol{a})$. Then each policy parameters are updated according to Eq. (2).

$$\theta^{i_k}_{z+1} = \underset{\theta}{\operatorname{argmax}} \left(\frac{1}{BT} \sum_{\substack{b=1 \\ t=0}}^{B,T} \min \left(\frac{\pi^{i_k}_{\theta^{i_k}}(a_t^{i_k}|o_t^{i_k})}{\pi^{i_k}_{\theta^{i_k}_z}(a_t^{i_k}|o_t^{i_k})} M^{i_{1:k}}(s_t, \boldsymbol{a}_t), \right. \right.$$
$$\left. \left. clip\left(\frac{\pi^{i_k}_{\theta^{i_k}}(a_t^{i_k}|o_t^{i_k})}{\pi^{i_k}_{\theta^{i_k}_z}(a_t^{i_k}|o_t^{i_k})}, 1 \pm \epsilon \right) M^{i_{1:k}}(s_t, \boldsymbol{a}_t) \right) \right) \tag{2}$$

where, $b \in [1..B]$ and $t \in [0..T]$.

$$M^{i_{1:k+1}}(s, \boldsymbol{a}) = \frac{\pi^{i_k}_{\theta^{i_k}_{z+1}}(a^{i_k}|o^{i_k})}{\pi^{i_k}_{\theta^{i_k}_z}(a^{i_k}|o^{i_k})} M^{i_{1:k}}(s, \boldsymbol{a}) \tag{3}$$

After all agents' policy parameters are improved, the value network parameters are updated according to Eq. (4).

$$\phi_{z+1} = \underset{\phi}{\operatorname{argmin}} \frac{1}{BT} \sum_{\substack{b=1 \\ t=0}}^{B,T} \left(V_\phi(s_t) - \hat{R}_t \right)^2 \tag{4}$$

D. Action and observation space

One of the contributions of this work concerns the definition of the action space as given in Eq. (5) and (6). In conventional approaches, the actions associated with a movement could have 8 directions: up (↑), up-right (↗), right (→), down-right (↘), down (↓), down-left (↙), left (←) and up-left (↖). In addition to movement actions, we propose two stationary actions: *stay* for remaining in place, and *comm* for enabling inter-agent communication as described in II-B.

$$\mathbf{a}_t = (a_t^{i_1}, ..., a_t^{i_n}) \tag{5}$$

$$a_t^{i_k} \in \mathcal{A}_k = \{\uparrow, \nearrow, \rightarrow, \searrow, \downarrow, \swarrow, \leftarrow, \nwarrow, stay, comm\} \tag{6}$$

Each agent's observation is a 3D matrix comprising its local, global, and transmission maps, as defined in II-A. Fig. 3 demonstrates how these observations are generated for four agents after various actions.

E. Reward and penalization terminology

To optimize non-repetitive exploration and enhance cluster-based communication among agents, we propose and compare various agent-based reward functions that can incorporate the following terms evaluated at time t:

- The exploration reward $r_t^{i_k,exp}$ is designed to persuade the agents to explore new grids compared to their global map. This term is described by the following expression:

$$r_t^{i_k,exp} = \frac{\Delta \mathcal{M}_{i_k,global}}{e_{max}} \tag{7}$$

where $\Delta \mathcal{M}_{i_k,global}$ denotes the number of grid cells that have been explored by the agent in the given time step and that increases its knowledge about the environment. The parameter e_{max} is computed according to Eq. (8) and denotes the maximum possible increase in environmental knowledge per step, depending on the detection range r_d, and cell size l.

$$e_{max} = 2\left(2\frac{r_d}{l} + 1\right) - 1 = \cdots = 4\frac{r_d}{l} + 1 \tag{8}$$

- The communication reward $r_t^{i_k,com}$, as defined in Eq. (9), incentivizes agents to share exploration knowledge. The parameter w_{i_k} makes this term different from zero only if the agent i_k communicates. In this case, the reward is proportional to the increase in knowledge due to communication within the cluster, $c_{i_k,global}$,

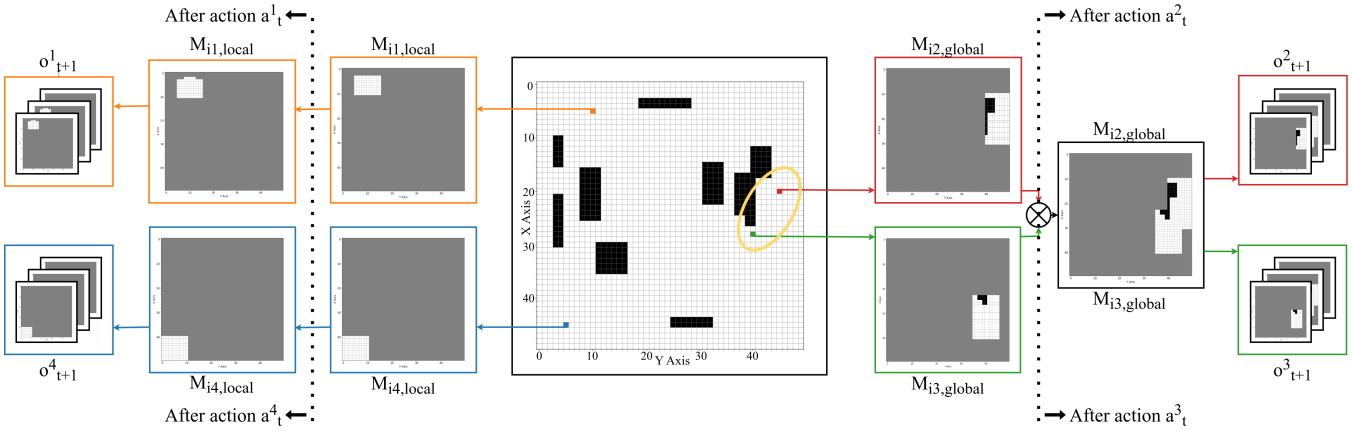


Fig. 3: Creation of the agents' observations resulting from the execution of their actions within the environment. Specifically, agent i_1 (orange) performs one step upwards. agent i_2 (red) and agent i_3 (green) communicate, and agent i_4 (blue) stays in the same position. Agent i_1's local map shows that the known area is enlarged after the action is executed, while agent i_4's local map does not change since it keeps its position. Agents i_2 and i_3 global maps are instead merged since the two agents can communicate. The resulting map is assigned to both agents. The collections of local, global, and transmission maps are the observations given to the agents.

divided by the number of the remaining unknown grids $n_{unknown}$.

$$r_t^{i_k,com} = w_{i_k} \frac{c_{i_k,global}}{n_{unknown}} \tag{9}$$

- Different types of penalizations: $r_t^{i_k,rep} = r_{rep}^*$ discourages agents from remaining stationary unless they are communicating, $r_t^{i_k,bou} = r_{bou}^*$ if the agent i_k occupies a cell near the exploration arena boundary, and $r_t^{i_k,ndi}$ that diminishes by r_{ndi}^* as $r_t^{i_k,exp}$ is lower than r_{exp}^* to emphasize the value of actions that expand environmental knowledge.

F. Synthesized reward functions

To evaluate the impact of including communication in the action space, various reward functions are formulated based on the terms described in II-E.

- *Case 1*: The reward function formulated in Eq. (10) considers only the contribution of exploration and not inter-agent communication.

$$r_t^{i_k} = r_t^{i_k,exp} + r_t^{i_k,rep} + r_t^{i_k,bou} + r_t^{i_k,ndi} \tag{10}$$

- *Case 2*: The objective function presented in Eq. (11) is analogous to *Case 1*, but with the addition of the communication term. As mentioned in II-E, this contribution to the reward prioritizes communication when its effect is to expand the agents' knowledge avoiding useless transmissions of data.

$$r_t^{i_k} = r_t^{i_k,exp} + r_t^{i_k,com} + r_t^{i_k,rep} + r_t^{i_k,bou} + r_t^{i_k,ndi} \tag{11}$$

- *Case 3*: A refined reward function is proposed by multiplying the communication reward in the objective function in *Case 2* by a parameter p_k formulated as per Eq. (12).

$$p_t^{i_k} = \frac{\Delta \mathcal{M}_{i_k,global}(previous\ step)}{e_{max}} + 0.6 \tag{12}$$

This coefficient accounts for the number of cells explored in the previous step. Thus, the reward function is fully expressed by Eq. (13).

$$r_t^{i_k} = r_t^{i_k,exp} + p_t^{i_k} r_t^{i_k,com} + r_t^{i_k,rep} + r_t^{i_k,bou} + r_t^{i_k,ndi} \tag{13}$$

- *Case 4*: In this reward function, the coefficient $p_t^{i_k}$ included in *Case 3* is reformulated as per Eq. (14) with the average number of grids explored by the agent since its last communication with agents in the same communication cluster.

$$p_t^{i_k} = \frac{1}{N-1} \sum_{i_j \in \mathcal{I}, i_j \neq i_k} \frac{w_{i_j} q_{i_j} o_{i_k,i_j}}{n^2} + 0.8 \tag{14}$$

In this formulation, the variable q_{i_j} takes the value 1 if the agent i_j participates in the same communication cluster as agent i_k. The term o_{i_k,i_j} stands for the number of discoveries performed by agent i_k since the last communication with agent i_j. Moreover, in this version, the boundary and negative exploration terms are not present, and the denominator of the communication reward $r_t^{i_k,com}$ is e_{max}.

$$r_t^{i_k} = r_t^{i_k,exp} + p_t^{i_k} r_t^{i_k,com} + r_t^{i_k,rep} \tag{15}$$

However, if an agent selects an action resulting in a collision, the assigned penalization is r_{col}^*.

III. Simulation Studies

A. Experimental setting

The training and testing of the agents' policies are performed in a Ubuntu 20.04.6 LTS Computer with 32 13th Gen Intel® Core™ i9-13900K and an NVIDIA GeForce RTX 4090. In particular, the policies' neural networks training has been done using samples from environments with different obstacles and starting positions for the agents. The learning rate has been set for the actor and critic networks at 5×10^{-4} and the number of epochs for update is set to 5. The policies have been trained on Z = 10,000 episodes, each episode has T = 200 and the batch size is B = 1. Meanwhile, evaluation is performed every 25 training steps. Overall, the

TABLE I: Metrics on the study cases

	n_{steps}		$\mathcal{J}$		*IG*		*robustness*
	Mean	Std	Mean	Std	Mean	Std	Value
Study case 1	588.75	189.79	0.26	0.10	365.50	61.87	0.48
Study case 2	453.25	204.78	0.27	0.05	573.67	149.16	0.64
Study case 3	462.29	187.19	0.24	0.04	701.50	-	0.92
Study case 4	416.39	160.31	0.25	0.04	445.23	216.11	0.92

environment has been modeled using the PettingZoo library and the Parallel API [22]. After multiple evaluations, the following values were chosen for the terms in the reward function: $r^*_{rep} = -1$, $r^*_{bou} = -1$, $r^*_{col} = -10$, $r^*_{ndi} = 0.2$, $r^*_{exp} = 0.3$. Meanwhile, the parameters of the environment and agents have been set as $n = 50$, $l = 0.5$, $r_d = 1$, $r_c = 5$.

B. Metrics

The following metrics have been used to evaluate and compare the different case studies mentioned in III-A:

- The number of steps n_{steps} required by the agents to map the environment considering that exploration is deemed successful when $p = 90\%$ of the obstacle-free grids are covered.
- The efficacy of the agents' collaboration is estimated from the overlap between their local maps using the Jaccard similarity coefficient [23]. This metric is computed at the end of the exploration process as the average Jaccard index across all pairs of agents' local maps as per Eq. (16).

$$\mathcal{J} = 2\frac{\sum_{i_k \in \mathcal{I}} \sum_{i_j \in \mathcal{I},\ j>k} J(\mathcal{M}_{i_k,\text{local}}, \mathcal{M}_{i_j,\text{local}})}{|\mathcal{I}| \times (|\mathcal{I}| - 1)} \tag{16}$$

- The effectiveness of communication is measured by the amount of information shared among agents within the same cluster, denoted as *IG*.
- The robustness of the trained policies when dealing with different environments.

C. Results

The study cases have been evaluated in 25 environments with different obstacles. Fig. 4 (b) shows that the cases that implement a communication term in the reward function manage to achieve the exploration in fewer steps compared to study case 1. Study case 2 lowered the median number of steps by over 100 compared to the no-communication case but had a wider interquartile range due to sparse data. Study case 3 improved data density and reduced the interquartile range. Finally, study case 4 achieved the best performance by further reducing the median number of steps. The episode rewards obtained during the training, as shown in Fig. 4 (a), shows that cases 2 and 4 converge more swiftly than the others, while case 1 appears to be the slowest one in reaching the convergence. Tab. I indicates that the policies from study case 4 exhibit the highest robustness, exceeding 90%, meaning they can effectively explore almost all the testing environments, whereas study case 1 is the least robust. Considering the other metrics, the average Jaccard

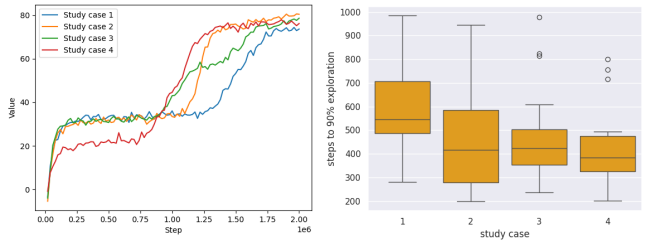


Fig. 4: From left to right. (a) Train episode reward curves for each study case and (b) statistical distribution of the number of steps required to explore the testing environments according to the four study cases.

similarity coefficients $\mathcal{J}$ show that both cases 3 and 4 obtain similar performances. The most significant deviation is the large standard deviation observed in study case 1, despite its arithmetic mean being similar to that of cases 3 and 4. However, significant differences exist in the amount of information shared between communicating agents across the various study cases. Tab. I shows that in case 3 there is just one simulation in which the agents were efficiently communicating with each other and the content of information shared is the largest in all the study cases. On the other hand, the policies obtained by using the reward function proposed in study case 4 achieve more communication instants compared to study case 3, but less information is shared each time. Study case 1 is the one in which the lowest sharing of knowledge happens between the agents and it performs similarly in every communication cluster due to the low variance. Study cases 2 achieve communication with more sharing of information compared to case 4 but comparatively low standard deviation except for case 1 which has the lowest standard deviation.

Eventually, Fig. 5 illustrates the results of the exploration performed by the study cases' policies in the same environment. In study case 1 the agents tend to overlap many times focusing the exploration in the rightmost part of the map. On the contrary, case 4 is the one with less overlap between the agents' trajectories since the agents explore different regions of the map, while cases 2 and 3 tend to have lots of overlays between agents. Concerning the balance in the division of exploration work by agents, it can be stated that in this environment case 1 is more balanced since in the other three cases agent 2 does not contribute as the other agents to the exploration. On the right, some plots illustrate the information shared during cluster-based communication. Case 4 was the one that counted the most communication events in this environment. Moreover, as shown by the collected images, the communication clusters have been very heterogeneous since it can be noted that the following

Fig. 5: Collection of exploration results performed by the study cases in an unknown environment with obstacles. The topmost image illustrates the free cells (black) and obstacles (yellow) that are in the environments. Each row contains the samples collected from the complete exploration done by the policies developed in each study case. In each series, the first image depicts the occupancy map of the environment after each exploration. In particular, they show the free cells (black), the obstacles (purple), and the unknown grids (yellow). Then, the trajectories of agents 1 to 4 are shown from left to right, respectively. These pictures show the cells covered by the trajectories (yellow), the direct discoveries of the agents (black), and the unknown cells (orange). The last images, whenever displayed, show the map that is shared between communicating agents in the same clusters. The yellow areas depict non-overlapping areas between the agents, while the purple, yellow, and orange ones are the sections that overlap between the agents that participate in the communication scheme

clusters have been formed: agent 1 and agent 2, agent 1 and agent 4, agent 3 and agent 4, agent 2 and agent 3, agent 1 and agent 3. The other study cases have not managed to bring satisfactory results concerning communication, except for case 2 in which a relevant communication pattern happens between agent 1 and agent 2.

Overall, considering the evaluated metrics and the simulation results depicted in Fig. 5, it can be stated that study case 4 minimizes the overlap between the agents during the exploration task, reduces the number of steps to reach at least 90% of exploration and proves to be robust. Moreover, Fig. 5 shows that it involves many different combinations of agents in the communication clusters.

IV. CONCLUSIONS

This article proposed a decentralized multi-agent reinforcement learning scheme that allows a set of homogeneous agents to explore an unknown environment with obstacles. Specifically, the main contribution concerns the design of different reward functions that can promote the sharing of knowledge about the environment between the different agents. The results have shown that the reward functions with communication terms reduce both the overlap in the explored area and the time steps needed to explore the environment. Indeed, this methodology aims to avoid the overlap in task execution by merging the knowledge accumulated by the different agents. Future investigations may focus attention on the evaluation of the developed policies in real experiments and the employment of heterogeneous agents.

REFERENCES

[1] A. Adeleye, "Robotic exploration for mapping," *arXiv preprint arXiv:2307.08690*, 2023.

[2] B. Yamauchi, "Frontier-based exploration using multiple robots," in *Proceedings of the second international conference on Autonomous agents*, 1998, pp. 47–53.

[3] S. Sharma and R. Tiwari, "A survey on multi robots area exploration techniques and algorithms," in *2016 International Conference on Computational Techniques in Information and Communication Technologies*. IEEE, 2016, pp. 151–158.

[4] A. Bautin, O. Simonin, and F. Charpillet, "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*. Springer, 2012, pp. 496–508.

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

[6] A. Batinović, J. Oršulić, T. Petrović, and S. Bogdan, "Decentralized strategy for cooperative multi-robot exploration and mapping," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 9682–9687, 2020.

[7] A. Koval, S. S. Mansouri, and G. Nikolakopoulos, "Online multi-agent based cooperative exploration and coverage in complex environment," in *2019 18th European Control Conference (ECC)*. IEEE, 2019, pp. 3964–3969.

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

[9] X. Zhang, Y. Ma, and A. Singla, "Task-agnostic exploration in reinforcement learning," *Advances in Neural Information Processing Systems*, vol. 33, pp. 11 734–11 743, 2020.

[10] A. Gupta, R. Mendonca, Y. Liu, P. Abbeel, and S. Levine, "Meta-reinforcement learning of structured exploration strategies," *Advances in neural information processing systems*, vol. 31, 2018.

[11] A. Nair, B. McGrew, M. Andrychowicz, W. Zaremba, and P. Abbeel, "Overcoming exploration in reinforcement learning with demonstrations," in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018, pp. 6292–6299.

[12] T. Kollar and N. Roy, "Trajectory optimization using reinforcement learning for map exploration," *The International Journal of Robotics Research*, vol. 27, no. 2, pp. 175–196, 2008.

[13] J. Hu, H. Niu, J. Carrasco, B. Lennox, and F. Arvin, "Voronoi-based multi-robot autonomous exploration in unknown environments via deep reinforcement learning," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 12, pp. 14 413–14 423, 2020.

[14] L. Deng, W. Gong, and L. Li, "Multi-robot exploration in unknown environments via multi-agent deep reinforcement learning," in *2022 China Automation Congress (CAC)*. IEEE, 2022, pp. 6898–6902.

[15] H. Zhang, J. Cheng, L. Zhang, Y. Li, and W. Zhang, "H2GNN: Hierarchical-hops graph neural networks for multi-robot exploration in unknown environments," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 3435–3442, 2022.

[16] A. Sygkounas, D. Tsipianitis, G. Nikolakopoulos, and C. P. Bechlioulis, "Multi-agent exploration with reinforcement learning," in *2022 30th Mediterranean Conference on Control and Automation (MED)*. IEEE, 2022, pp. 630–635.

[17] T. Chu, S. Chinchali, and S. Katti, "Multi-agent reinforcement learning for networked system control," *arXiv preprint arXiv:2004.01339*, 2020.

[18] R. Müller, H. Turalic, T. Phan, M. Kölle, J. Nüßlein, and C. Linnhoff-Popien, "ClusterComm: Discrete communication in decentralized marl using internal representation clustering," *arXiv preprint arXiv:2401.03504*, 2024.

[19] R. Pina, V. De Silva, C. Artaud, and X. Liu, "Fully independent communication in multi-agent reinforcement learning," *arXiv preprint arXiv:2401.15059*, 2024.

[20] S. V. Albrecht, F. Christianos, and L. Schäfer, *Multi-agent reinforcement learning: Foundations and modern approaches*. MIT Press, 2024.

[21] Y. Zhong, J. G. Kuba, X. Feng, S. Hu, J. Ji, and Y. Yang, "Heterogeneous-agent reinforcement learning," *Journal of Machine Learning Research*, vol. 25, no. 1-67, p. 1, 2024.

[22] J. Terry, B. Black, N. Grammel, M. Jayakumar, A. Hari, R. Sullivan, L. S. Santos, C. Dieffendahl, C. Horsch, R. Perez-Vicente *et al.*, "Pettingzoo: Gym for multi-agent reinforcement learning," *Advances in Neural Information Processing Systems*, vol. 34, pp. 15 032–15 043, 2021.

[23] L. d. F. Costa, "Further generalizations of the jaccard index," *arXiv preprint arXiv:2110.09619*, 2021.