Towards Cooperation and Fairness in Multi-Agent Reinforcement Learning

Jasmine Jerry Aloor
jjaloor@mit.edu
Massachusetts Institute of Technology

Siddharth Nayak
sidnayak@mit.edu
Massachusetts Institute of Technology

Sydney Dolan
sydneyd@mit.edu
Massachusetts Institute of Technology

Victor Qin
victorqi@mit.edu
Massachusetts Institute of Technology

Hamsa Balakrishnan
hamsa@mit.edu
Massachusetts Institute of Technology

Abstract

We consider the problem of fair multi-agent navigation for a large group of decentralized agents using multi-agent reinforcement learning (MARL). Previous MARL research optimizes for efficiency, safety, or both. We introduce a methodology that considers fairness over the entirety of the simulation period using a specialized reward function and decentralized goal assignment approach. Specifically, our work (1) incorporates fairness into the reward function, which leads to (2) agents learning a fair assignment of goals and (3) achieves almost perfect goal coverage in the navigation scenario using each agent's localized observations in a decentralized manner that allows scaling to any number of agents. We show that our work achieves higher fairness metrics when compared to other centralized goal-assignment fairness methodologies.

1 Introduction

Scalable coordination methods with equitable resource distribution will need to be developed to manage interactions between entities in large multi-agent systems like advanced air mobility (Chin et al., 2023) and space traffic management (Dolan et al., 2023). Traffic management for these systems must actively direct well-actuated but fast-moving and safety-critical autonomous agents to reach their intended destinations without incident.

Multi-agent reinforcement learning (MARL) approaches enable the investigation of a range of interactions (e.g., competitive or cooperative) between agents or between teams of agents. The ability of RL to learn by trial and error makes it well-suited for problems in which optimization-based methods are not effective. In particular, multi-agent reinforcement learning (MARL) approaches are suitable in these situations due to their fast run times, superior performance, and ability to model shared goals between agents using appropriate reward structures.

One common approach to training cooperative multi-agent reinforcement learning models is the centralized training decentralized execution (CTDE) paradigm. In CTDE, decentralized agent policies are trained through a centralized mixing model with global state information, while agents select actions using only their local observations (Zhou et al., 2023). As an example, agents could be required to complete a set of tasks in a collaborative manner. The learning process occurs as the agents seek to find an optimal policy to maximize the shared reward function. While this shared reward encourages collaboration amongst agents, it does not consider the fairness of each agent's task assignment, meaning that certain agents could contribute a disproportionate portion of the overall shared reward. We wish to avoid any assignment that distributes the tasks in an inequitable way that leads to some agents receiving an unfair advantage while others starve for resources.

Our approach focuses on the evolution of fairness throughout the course of an episode in a cooperative, decentralized multi-agent reinforcement learning setting. Specifically, we design a reward function that encourages both the completion of the task and the overall fairness in each agent's task selection and execution. This reward is based on optimal cost distance assignment or minimax fair distance assignment, which allows agents to learn fairness and solve it adaptively.

Our contributions include the following:

- We introduce a reward function that enables the underlying reinforcement learning algorithm to learn fair behavior.
- We demonstrate a decentralized learning-based goal assignment approach that allows agents to adaptively select their goals during execution rather than relying on a centralized goal assignment algorithm.

The remainder of this paper is organized as follows: In Section 2, we review related work in the field. Section 3 introduces our problem formulation and methods. Section 4 presents our results and discusses our findings. Finally, we conclude in Section 5 with future work directions.

2 Related Work

2.1 Fairness

Fairness has been extensively studied in many contexts, including in game theory (Jong et al., 2008), economics (Frey & Pommerehne, 1993), and machine learning (Caton & Haas, 2024). In machine learning, work in fairness often refers to mitigating social biases and the social, legal, and ethical aspects of machine learning discrimination. One of the most common classes of approaches is the alpha fairness method (Altman et al., 2008). Our work is concerned with fairness in the network engineering sense (Huaizhou et al., 2013), where individual users receive a fair share of system resources (Kleinberg et al., 2001).

There are relatively few works in MARL studying this definition of fairness. De Jong et al. (2008) incorporated priority awareness into their fairness modeling. As human notions of fairness often consider it fair if agents receive slightly different rewards in the presence of additional contextual information, this work assumes that each agent knows the true priority of all other agents in the simulation. This approach produces sub-optimal reward solutions. In Grupen et al. (2022), they find that sophisticated coordination behavior only emerges when there is a shared reward but that this emergent behavior does not ensure fairness. They introduce a soft-constraint equivariant policy learning method to dynamically balance the fairness-utility trade-off. By contrast, our work considers fairness in navigation-based settings where global information is not available to each agent.

2.2 Communication Structures in Multi-Agent Reinforcement Learning

Our work focuses on the problem of multi-agent navigation and collision avoidance among a set of decentralized N agents that can only sense the presence of other obstacles and agents within a limited radius r. Communication between agents is vital for them to complete their respective tasks successfully. The study of the role of communication between agents is an active and extensive field within MARL, so we refer the reader to a survey on the topic (Zhu et al., 2024) for a comprehensive description. We highlight several works in this area that address problems similar to ours.

Existing MARL work on this problem often assumes that even if the behavior of the agents is decentralized, communication amongst them is *centralized*. This means that all agents have access to messages from all other agents in the environment. Unfortunately, this means that as the number of agents increases, the computational expense of communication increases superlinearly. Centralized communication also creates privacy concerns, where agents are unable to participate in the MARL framework without sharing data with everyone. To address this expense, ATOC (Jiang & Lu, 2018; Das et al., 2018) relies on attention mechanisms to provide a compact representation of message priority. In EMP (Agarwal et al., 2019), the environment is translated into a shared agent-entity graph representation that allows agents to communicate along connected edges. This formulation provides a compact graph representation of the communication connections between all entities in the environment. However, all agents must know the positions of all entities in the graph at the beginning of an episode, and thus, a similar centralized communication constraint arises. In contrast to these centralized communication approaches, InforMARL (Nayak et al., 2023) differs from these works in that it relies on only locally available information throughout training and during evaluation. Our algorithm relies on InforMARL as the underlying MARL algorithm for training and evaluation, with several adaptations to its reward and buffer structure.

3 Methodology

In this section, we will describe a modified reward function that trains policies with fairer outcomes of agents to goals. We first describe the environment, then discuss centralized efficiency and fairness optimization formulations, and finally discuss modifications to the reward function that lead to fair behavior.

We train an adapted version of InforMARL, an existing CTDE multi-agent reinforcement learning algorithm, on our modified reward functions. The details of this algorithm are beyond the scope of this paper and can be found in Nayak et al. (2023).

3.1 Preliminaries

Following the InforMARL framework, our environment comprises entities categorized into agents, obstacles, and goals. For each agent i at each time-step t, we define an agent-entity graph as $g_t^{(i)} \in \mathcal{G} : (\mathcal{V}, \mathcal{E})$, where each node $v \in \mathcal{V}$ is an entity in the environment. Entities are connected to each other by edges if they are within a certain sensing distance. Agent-agent edges are bi-directional, which is equivalent to a communication channel between them, whereas agent-non-agent edges are unidirectional, with messages being passed from the non-agent entity to the agent.

Our environment is based on the Multi Particle Environments (MPE) (Lowe et al., 2017) collection of tasks. We formulate our environment as a Decentralized Partially Observable Markov Decision Process (Dec-POMDP) defined by the tuple $\langle N, S, O, \mathcal{A}, \mathcal{G}, P, R, \gamma \rangle$, where:

- N is the number of agents
- $s \in S = \mathbb{R}^{N \times D}$ is the state space of the environment, with D as the dimension of the state
- $o^{(i)} = O(s^{(i)}) \in \mathbb{R}^d$ is the local observation for agent i.
- $a^{(i)} \in \mathcal{A}$ is the action space for agent i.The action space for each agent is discretized such that it can control unit acceleration and deceleration in the x- and y- directions.
- $g^{(i)} \in \mathcal{G}(s; i)$ is the graph network formed by the entities in the environment with respect to agent i
- $P(s'|s, A)$ is the transition probability from s to s' given the joint action A
- $R(s, A)$ is the joint reward function
- $\gamma \in [0, 1)$ is the discount factor

Note that our environment has modified the reward structure and the collaborative information sharing setup compared to the environments originally tested for InforMARL. The task is to find a policy $\Pi = \left(\pi^{(1)}, \cdots, \pi^{(N)}\right)$, where $\pi_\theta^{(i)}\left(a^{(i)}|o^{(i)}, g^{(i)}\right)$ for agent i selects action $a^{(i)}$ based on its graph network $g^{(i)}$ and the local observation $o^{(i)}$ which consists of its position and velocity in a global frame of reference and the information about each agent's goals denoted by $\zeta^{(i)} \in G$. Any goal position that is input to the agent's observation is the relative position of the goals with respect to the agent's position. We also add a goal occupancy flag that informs agents of the proximity of any agent to a particular goal. In Section 3.2, we discuss this modification that allows our algorithm to be scalable and more decentralized.

We investigate a navigation problem involving N agents that navigate a 2D space using a double integrator dynamics model (Rao & Bernstein, 2001). Agents are tasked with navigating to goals in the environment such that each agent is at a unique goal while avoiding obstacles and walls. At each timestep, we calculate the preliminary reward for each agent $a^{(i)}$, $\mathcal{R}_d(s_t, a_t^{(i)})$ by taking the negative mean of the minimum Euclidean distances to an assigned goal. When an agent reaches its assigned goal, it receives a goal reward $\mathcal{R}_g(s_t, a_t^i)$. The agents are flagged as "done" and stay locked in at the goal. We also restrict the actions that can be taken by these "done" agents. This prevents the agents from drifting away under the collision force influence of other agents that are still navigating to their respective goals. We call this setting "death-masking" based on Yu et al. (2021).

To demonstrate the impact of our learning-based goal assignment and fairness reward function, we compare our methods against fair *a priori* goal assignment methods. In Section 3.3, we discuss these *a priori* goal assignment techniques used in this work. In Section 3.4, we discuss the structure of our fairness reward function.

3.2 Observations for Decentralized Execution

In typical MARL applications, agents are provided with all goal positions as well as information about the neighboring agents in the environments. This leads to a lower level of privacy and assumes global knowledge of every agent's position. When agents are sparsely distributed in larger environments, they can only observe parts of the environment preventing them from knowing the positions of all goals *a priori*. We model this in our observation function by only using the positions of the closest two goals from each agent. The information of other goals is shared between agents within the sensing distance using graph message passing.

We provide a goal occupancy flag that informs agents how close any agent is to that goal. It allows agents to know if a goal in their sensing range is occupied or will be soon occupied due to the presence of another agent in its proximity. The goal occupancy flag η is created for each goal in the environment. We populate the flags based on the distance of the closest agent to the goal. For each goal $\zeta^{(i)}$ with position $p_g^{(i)}$, we initialize all $\eta^{(i)}$ to 0. As agents move closer to the goals, we calculate the minimum distance any agent is from a particular goal i,

$$\begin{aligned} d_{\text{min}}^{(i)} &= \min_{j \in N} \| x^{(j)} - p_g^{(i)} \|_2 \\ \eta^{(i)} &= 1 - d_{\text{min}}^{(i)} \end{aligned} \tag{1}$$

We restrict the value of η to be within 0 and 1 for ease of computation. For a given goal, the flag value $\eta^{(i)}$ increases from 0 to 1 as an agent tries to reach it. In particular, $\eta^{(i)} = 0$ means the goal $\zeta^{(i)}$ is available for any agent, and 1 indicates that the goal is occupied. The final ego observation $o^{(i)}$ is $o^{(i)} = [p_i, v_i, p^{\text{goal}_1}, \eta_i^{\text{goal}_1}, p^{\text{goal}_2}, \eta_i^{\text{goal}_2}]$. The neighborhood information is also collected into a graph observation vector that is then passed into the GNN. $x_j = [p_i^j, v_i^j, p_i^{\text{goal}_1, j}, \eta_i^{\text{goal}_1}, \texttt{entity\_type(j)}]$ where $p_i^j, v_i^j, p_i^{\text{goal}_1, j}$ are the *relative* position, velocity, and position of the closest goal of the entity at node j with respect to agent i, respectively. The variable $\texttt{entity\_type(j)} \in \{\texttt{agent}, \texttt{obstacle}, \texttt{goal}\}$ determines the type of entity at node j.

3.3 Goal Assignment

Assigning agents to goals is a well-investigated problem (Skaltsis et al., 2021). In this paper, we compare an optimal method of assigning goals with a min-max fair method and describe their implementation here. Note that this assignment is only used to determine the rewards for the agents and is not used in the observations.

3.3.1 Optimal Cost Assignment

For an efficient assignment, we use the linear sum assignment or minimum weight matching in the bipartite graphs algorithm. This matches each agent i to a goal j so that the total cost c_{ij} for all agents is minimized, which here corresponds to minimizing the total distance, or $\min_{x_{ij}} \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij}$. where x_{ij} is 1 if agent i is assigned to goal j and 0 otherwise.

3.3.2 Min-Max Fair Assignment

We compare the optimal cost assignment with the performance of agents when the system is provided with a fair assignment of agent-goal results. We determine a min-max fair assignment by optimizing the objective $\min z$ subject to constraints ensuring each agent and goal is assigned exactly once. z represents the maximum cost assigned to any agent, $c_{ij} x_{ij} \leq z$ where c_{ij} and x_{ij} are defined as above.

3.4 Fairness

Many works prior have tried to enable fairness in multi-agent networked systems. We want to incorporate fairness by having an equitable allocation of resources. We choose the distance traveled and time taken by each agent to reach its goal as the metrics for fairness. We would like to minimize the standard deviation of distance traveled and time to reach goals among all agents. By incorporating these metrics in our reward function, we can add fairness to agent behavior in the system.

The distance traveled by each agent i at each time step is represented by $d_t^{(i)}$, with an overall mean μ_t and standard deviation σ_t for all agents. We compute the coefficient of variation as $CV_t = \sigma_t / \mu_t$. We choose our fairness metric $\mathcal{F} = 1/CV$ to be the inverse of CV so that it is a non-dimensional quantity with higher numerical values indicating greater fairness.

$$\mathcal{F}_t(s_t, a_t^{(i)}) = \frac{1}{CV_t} \tag{2}$$

We then pass $\mathcal{F}$ to every agent policy $\pi^{(i)}$ through the reward function. The fairness metric is transformed through a $tanh$ function whose zero value is shifted appropriately by τ_0 to enable agents to have a higher fairness metric. This value is scaled up by a factor λ to determine the fairness reward $\mathcal{R}_f(s_t, a_t^{(i)})$.

$$f_t = tanh(\mathcal{F}_t(s_t, a_t^{(i)}) - \tau_0) \tag{3a}$$

$$\mathcal{R}_f(s_t, a_t^{(i)}) = \lambda f_t \tag{3b}$$

Another metric could be the standard deviation or its inverse, $1/\sigma_t$. Further investigation found that the trends in fairness are similar between the two metrics; thus, we continue our implementation with $1/CV$, which is a non-dimensional quantity. The agents have a common objective to reach the targets, and the fairness metric does not directly conflict with the goal reward.

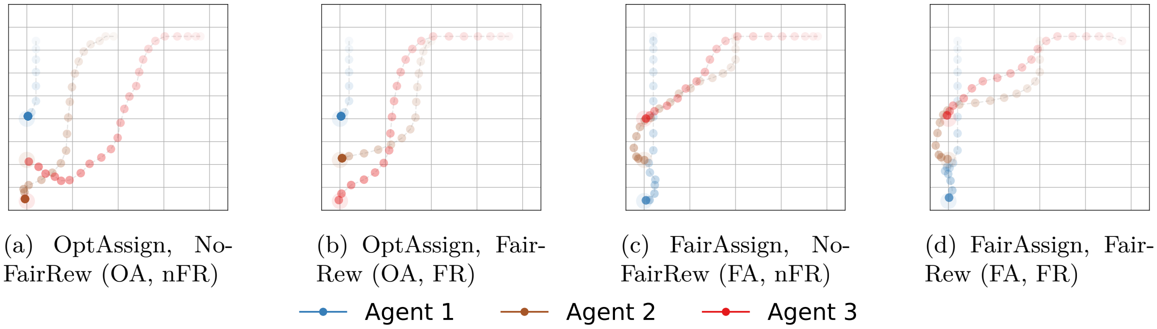
We also have a goal reaching reward $\mathcal{R}_g(s_t, a_t^{(i)})$. This goal-reaching reward occurs when an individual agent reaches their assigned goal, as marked by ρ. ρ is 1 if the agent reaches the goal; otherwise, 0. The overall reward is tuned with the addition of the fairness metric, which we will call the "fair" reward.

$$\mathcal{R}_{total}(s_t, a_t^{(i)}) = \mathcal{R}_d(s_t, a_t^{(i)}) + \mathcal{R}_f(s_t, a_t^{(i)}) + \rho \mathcal{R}_g(s_t, a_t^{(i)}) \tag{4}$$

4 Experiments

4.1 Problem Setting

We formulate our problem as a navigation coverage task (Tokekar et al., 2014; Dames et al., 2017), where each agent must reach a single unique goal. In a navigation task with pre-assigned goals, the



(a) OptAssign, No-FairRew (OA, nFR)

(b) OptAssign, Fair-Rew (OA, FR)

(c) FairAssign, No-FairRew (FA, nFR)

(d) FairAssign, Fair-Rew (FA, FR)

Figure 1: Visualization of behaviors of the four navigation cases with and without fairly assigned goals and fairness metric rewards. The agents start from the upper half of the environment and navigate to goals located on the bottom left part of the environment. The darker shades indicate newer states in the trajectories traveled by each agent, and the lighter circles indicate earlier states.

Metric Type	Fairness metric $\mathcal{F}$			
	OA, nFR	OA, FR	FA, nFR	FA, FR
	2.43	2.77	11.82	**12.44**

Table 1: Fairness metric $\mathcal{F}$ calculated for the four navigation scenarios shown in Fig. 1.

agent's policy is fixated on reaching that particular goal. However, this approach is not feasible for a scalable decentralized system, as it fixes the model's input and outputs.

Based on the local observations at each step, the agent is provided with the two closest goals available for that step. The agents also get additional information about the two closest goals to prevent issues with crowding on any one goal. For each of the two selected goals, the agents also receive a goal occupancy flag, which determines if the goal has been occupied by another agent. This enables the agents to prioritize goals that they need to navigate to based on the level of occupancy of a particular goal.

We use both goal assignment schemes and add the fairness reward to create four cases for the coverage navigation scenario, varying using the optimal (Sec. 3.3.1) or fair (Sec. 3.3.2) assignment and including and not including the fairness reward in Eqn. 4.

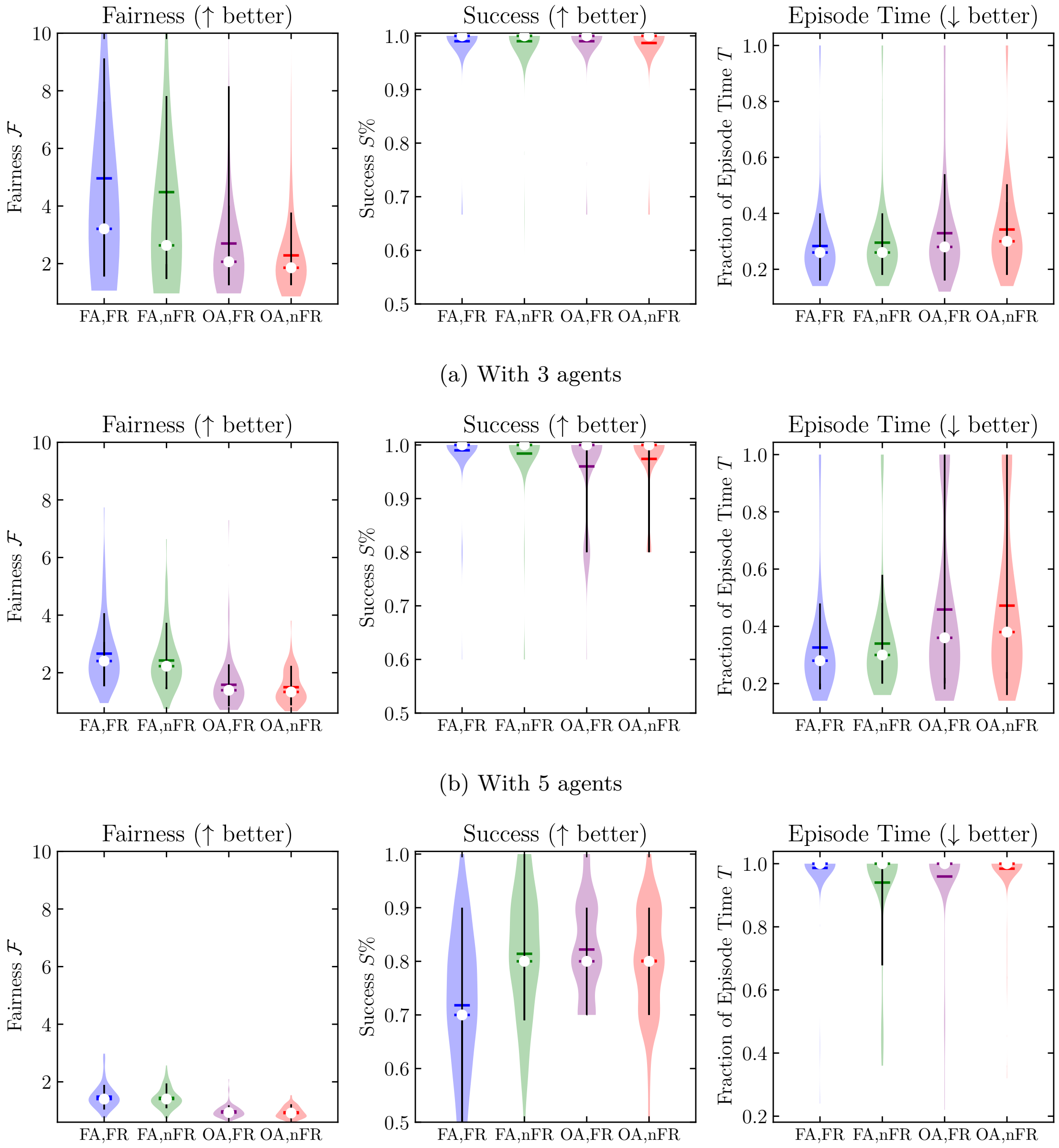
1. Fair goal assignment with fairness reward (FA, FR)
2. Fair goal assignment with no fairness reward (FA, nFR)
3. Optimal distance cost goal assignment with fairness reward (OA, FR)
4. Optimal distance cost goal assignment with no fairness reward (OA, nFR)

An example scenario of these four cases is shown in Fig. 1. Three agents are initialized in the upper portion of the environment, and the three goals are located in the lower left corner. Agents navigate to their goals based on the different policies they are trained on, and the colored dots show their trajectories. Table 1 shows the fairness metric values for each case for this episode.

4.2 Evaluation

We train with three agents in each scenario and evaluate with 3, 5 and 10 agents over 100 episodes. During evaluation, we do not rely on a centralized critic. We also do not require knowing the global positions of all agents and use communication through message passing to sense other agents, obstacles, and goals present in the environment. We calculate the following metrics:

1. Fairness, which is the total fairness value (Eqn. 2) throughout the episode, denoted by $\mathcal{F}$ (higher the better).



(a) With 3 agents

(b) With 5 agents

(c) With 10 agents

Figure 2: The violin plots show the distribution of fairness ($\mathcal{F}$), success rates ($S\%$) and fraction of episode time (T) over 100 episodes for each of the four cases of goal assignment and fairness reward models discussed in Section 4.1: 1) Fair goal assignment with fairness reward (FA, FR), 2) Fair goal assignment with no fairness reward (FA, nFR) 3) Optimal distance cost goal assignment with fairness reward (OA, FR) 4) Optimal distance cost goal assignment with no fairness reward (OA, nFR). The medians are denoted by a white circle and tick, a plain tick represents the means, and the vertical black lines indicate the 90-10 percentile range. The trends show that the case (FA, FR) has the highest fairness values, greater success rates, and lower completion time for 3 and 5-agent scenarios. The results for the scenario with 10 agents are skewed due to congestion in the environment.

2. Success rate as the percent of episodes in which all agents are able to get to unique goals, denoted by $S\%$ (higher is better).
3. The fraction of an episode time taken by all agents to get to their goal, denoted T (lower is better). T is set to 1 if any agent does not reach its goal.

Figure 2 shows the distribution of $\mathcal{F}$, $S\%$ and T along with the medians, means, and 90-10 percentile ranges for the four cases of the coverage navigation task. The results show that our method (FA, FR) has the highest fairness metrics as compared to the other cases. We also see higher success rates $S\%$ and a lower fraction of episode time T. Case (FA, nFR) also performs well, showing that the agents have learned a fair assignment and are utilizing it to maintain a greater fairness metric independent of the fairness reward. The trend is different for the 10 agents scenarios as the agents experience congestion with a greater agent count in the same environment. This leads to more collisions, resulting in a decrease in overall fairness and success rates. Further results are provided in Table 2 in Appendix A.

5 Conclusions and Future Work

In this work, we proposed a method to incorporate fairness into the multi-agent navigation problem. The agents are able to achieve almost complete coverage of the goals in each scenario. Our approach enables a greater level of decentralization with less dependence on a centralized oracle. As an added benefit, we obtain some privacy by not requiring that all agents and goal positions be known before the navigation starts. We utilized goal occupancy values for each goal to indicate to agents if a particular goal is occupied and not worth reaching. We compared the fairness metric and success rates of the coverage task for both goal assignment schemes with and without the inclusion of the fairness reward. We also implemented “death masking” to prevent collision forces from influencing the goal-reaching behaviors. In future work, we would like to extend this algorithm to various formation tasks for a swarm of agents. We will evaluate the level of fairness metrics achieved as agents come into formation and compare the fairness-informed models to our baselines.

References

Akshat Agarwal, Sumit Kumar, and Katia P. Sycara. Learning transferable cooperative behavior in multi-agent teams. *CoRR*, abs/1906.01202, 2019. URL http://arxiv.org/abs/1906.01202.

Eitan Altman, Konstantin Avrachenkov, and Andrey Garnaev. Generalized α-fair resource allocation in wireless networks. In *2008 47th IEEE Conference on Decision and Control*, pp. 2414–2419, 2008. doi: 10.1109/CDC.2008.4738709.

Simon Caton and Christian Haas. Fairness in machine learning: A survey. *ACM Comput. Surv.*, 56 (7), Apr 2024. ISSN 0360-0300. doi: 10.1145/3616865. URL https://doi.org/10.1145/3616865.

Christopher Chin, Victor Qin, Karthik Gopalakrishnan, and Hamsa Balakrishnan. Traffic management protocols for advanced air mobility. *Frontiers in Aerospace Engineering*, 2, May 2023. doi: 10.3389/fpace.2023.1176969.

Philip Dames, Pratap Tokekar, and Vijay Kumar. Detecting, localizing, and tracking an unknown number of moving targets using a team of mobile robots. *The International Journal of Robotics Research*, 36(13-14):1540–1553, 2017. doi: 10.1177/0278364917709507. URL https://doi.org/10.1177/0278364917709507.

Abhishek Das, Théophile Gervet, Joshua Romoff, Dhruv Batra, Devi Parikh, Michael G. Rabbat, and Joelle Pineau. Tarmac: Targeted multi-agent communication. *CoRR*, abs/1810.11187, 2018. URL http://arxiv.org/abs/1810.11187.

Steven De Jong, Karl Tuyls, Katja Verbeeck, and Nico Roos. Priority awareness: towards a computational model of human fairness for multi-agent systems. In *Adaptive Agents and*

Multi-Agent Systems III. Adaptation and Multi-Agent Learning, pp. 117–128. Springer, 2008. doi: 10.1007/978-3-540-77949-0_9. URL https://link.springer.com/chapter/10.1007/978-3-540-77949-0_9.

Sydney Dolan, Siddharth Nayak, and Hamsa Balakrishnan. Satellite navigation and coordination with limited information sharing. In Nikolai Matni, Manfred Morari, and George J. Pappas (eds.), *Proceedings of The 5th Annual Learning for Dynamics and Control Conference*, volume 211 of *Proceedings of Machine Learning Research*, pp. 1058–1071. PMLR, 15–16 Jun 2023. URL https://proceedings.mlr.press/v211/dolan23a.html.

Bruno S. Frey and Werner W. Pommerehne. On the fairness of pricing ‚Äî an empirical survey among the general population. *Journal of Economic Behavior & Organization*, 20(3):295–307, 1993. ISSN 0167-2681. doi: https://doi.org/10.1016/0167-2681(93)90027-M. URL https://www.sciencedirect.com/science/article/pii/016726819390027M.

Niko A. Grupen, Bart Selman, and Daniel D. Lee. Cooperative multi-agent fairness and equivariant policies. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(9):9350–9359, Jun. 2022. doi: 10.1609/aaai.v36i9.21166. URL https://ojs.aaai.org/index.php/AAAI/article/view/21166.

Shi Huaizhou, R Venkatesha Prasad, Ertan Onur, and IGMM Niemegeers. Fairness in wireless networks: Issues, measures and challenges. *IEEE Communications Surveys & Tutorials*, 16(1): 5–24, 2013.

Jiechuan Jiang and Zongqing Lu. Learning attentional communication for multi-agent cooperation. *CoRR*, abs/1805.07733, 2018. URL http://arxiv.org/abs/1805.07733.

Steven de Jong, Karl Tuyls, and Katja Verbeek. Fairness in multi-agent systems. *The Knowledge Engineering Review*, 23(2):153–180, 2008. doi: 10.1017/S026988890800132X.

Jon Kleinberg, Yuval Rabani, and Éva Tardos. Fairness in routing and load balancing. *Journal of Computer and System Sciences*, 63(1):2–20, August 2001. ISSN 0022-0000. doi: 10.1006/jcss.2001.1752. URL https://www.sciencedirect.com/science/article/pii/S0022000001917520.

Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pp. 6382–6393, 2017.

Siddharth Nayak, Kenneth Choi, Wenqi Ding, Sydney Dolan, Karthik Gopalakrishnan, and Hamsa Balakrishnan. Scalable multi-agent reinforcement learning through intelligent information aggregation. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 25817–25833. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/nayak23a.html.

V.G. Rao and D.S. Bernstein. Naive control of the double integrator. *IEEE Control Systems Magazine*, 21(5):86–97, 2001. doi: 10.1109/37.954521.

George Marios Skaltsis, Hyo-Sang Shin, and Antonios Tsourdos. A survey of task allocation techniques in mas. In *2021 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 488–497, 2021. doi: 10.1109/ICUAS51884.2021.9476736.

Pratap Tokekar, Volkan Isler, and Antonio Franchi. Multi-target visual tracking with aerial robots. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3067–3072, 2014. doi: 10.1109/IROS.2014.6942986.

Chao Yu, Akash Velu, Eugene Vinitsky, Yu Wang, Alexandre M. Bayen, and Yi Wu. The surprising effectiveness of MAPPO in cooperative, multi-agent games. *CoRR*, abs/2103.01955, 2021. URL https://arxiv.org/abs/2103.01955.

Yihe Zhou, Shunyu Liu, Yunpeng Qing, Kaixuan Chen, Tongya Zheng, Yanhao Huang, Jie Song, and Mingli Song. Is centralized training with decentralized execution framework centralized enough for marl?, 2023.

Changxi Zhu, Mehdi Dastani, and Shihan Wang. A survey of multi-agent deep reinforcement learning with communication. *Autonomous Agents and Multi-Agent Systems*, 38(1):4, January 2024. ISSN 1573-7454. doi: 10.1007/s10458-023-09633-6. URL https://doi.org/10.1007/s10458-023-09633-6.

A Appendix

Number of Agents	Experiment Type	Fairness $\mathcal{F}_d \uparrow$	Fairness $\mathcal{F}_t \uparrow$	Success $S\% \uparrow$	Episode Fraction Time (T) $\downarrow$
3	FA, FR	**3.21**	**4.26**	99.0	0.28
	FA, nFR	2.63	4.24	99.0	0.30
	OA, FR	2.07	2.79	99.0	0.33
	OA, nFR	1.86	2.53	98.7	0.34
5	FA, F	**2.41**	**3.02**	99.0	0.33
	FA, nFR	2.23	**3.02**	98.4	0.34
	OA, FR	1.39	1.86	96.0	0.46
	OA, nFR	1.33	1.74	97.4	0.47
10	FA, FR	**1.40**	0.99	71.8	0.99
	FA, nFR	**1.40**	**1.18**	81.4	0.94
	OA, FR	0.93	0.72	82.2	0.96
	OA, nFR	0.92	0.68	80.1	0.98

Table 2: Comparison of the fairness informed experiments. The following metrics are compared: (a) fairness based on distance traveled ($\mathcal{F}_d$, higher is better), (b) fairness based on the time taken ($\mathcal{F}_t$, higher is better), (c) success rates ($S\%$, higher is better) and (d) fraction of episode time (T, lower is better) over 100 episodes for each of the four cases of goal assignment and fairness reward models discussed in Section 4.1: 1) Fair goal assignment with fairness reward (FA, FR), 2) Fair goal assignment with no fairness reward (FA, nFR) 3) Optimal distance cost goal assignment with fairness reward (OA, FR) 4) Optimal distance cost goal assignment with no fairness reward (OA, nFR). The medians are presented over 100 episodes for all but the success rate where the mean values are presented. The (FA, FR) method has the highest fairness based on distance traveled and fairness based on the time taken for 3 and 5 agents scenarios.