Modeling Priorities in Multi-agent Multi-objective Systems

Nitin Kulkarni, Jake Sanders, Sargur Srihari, Chunming Qiao, and Alina Vereshchaka

Department of Computer Science and Engineering,
State University of New York at Buffalo, Buffalo, USA
{nitinvis, jakesand, srihari, qiao, avereshc}@buffalo.edu

Abstract. Reinforcement learning methods can successfully solve many real-world problems. Multi-agent reinforcement learning involves cooperation among multiple agents and has seen successful applications in diverse domains such as robotics, distributed control, and economics. However, some of the remaining challenges that need further discussion are that of fairness and the formulation of a problem that is free of bias and discrimination. In this paper, we consider what be identified as fair behavior in multi-agent systems. Multi-agent reinforcement learning is a framework that has many applications which can mimic and solve real-world optimization problems that consider multi-agent interactions. In this paper, we aim to discuss possible approaches that incorporate fair behaviors and pay attention in particular to the priority of agents in the systems. We aim to maximize the overall utility across all agents while incorporating agent priorities based on their types.

Keywords: Modeling · Multi-agent systems · Reinforcement learning · Fairness · Ethical behavior · Agent priorities · Resource allocation

1 Introduction

Reinforcement learning (RL) methods have been successfully applied to various domains including robotics [1, 2], transportation [3], autonomous driving, industry automation [4] and epidemic mitigation [5] resulting in new and better solutions [6]. Deep RL has proven to be a powerful tool for solving a diverse range of complex problems [7]. RL can be used to solve problems with a high degree of stochasticity such as problems in supply chain management, IoT, and networking [8].

Many real-world problems require multiple RL agents to cooperate with each other [9]. These problems fall into the realm of multi-agent reinforcement learning (MARL). MARL has been successful in many applications such as controlling UAVs [10,11], efficient utilization of resources [12–15], transportation [16,17], and social sciences [18, 19].

One significant challenge that remains unresolved in the field of RL is that of fairness, which in the context of decision-making protocols this means free of bias and discrimination. As is common with any ethical consideration, it is necessary to consider different conditional contexts for fairness in a given decision-making process. In the case of an RL-program's model and workflow, it is the responsibility of the RL practitioner to limit the possibility of discrimination, to explain the model's fairness considerations to the user, and to ensure that the fair decision-making process is reproducible throughout the entirety of the RL framework.

As in human-human interactions, in RL, eliminating bias entirely is not realistic but it is possible to weed bias out of the program through the integration of certain rules into the problem formulation stage. In order to ensure fairness in RL, it is necessary to guarantee that predictions are calibrated for the group of agents in the RL model that abide by certain decision boundaries. However, in addition to "group fairness" calibrated to a selection of agents, it is also necessary to account for individual fairness, such that each agent understands it is being treated fairly by the RL system, or at the very least understands the decision-making process that causes separate agents to reach their individual/common goals differently.

The objective of any artificial intelligence (AI) system is set by humans to benefit us and the planet. AI systems must incorporate our goals and preferences. While formulating the task or the objective, we need to consider the sense of fairness that agents learn and their ethical principles and behaviors. This is extremely important, as the policies they learn will be applied to real-world applications or interactions.

Agents optimize the process for accomplishing the objective and while doing so may incorporate bias given the objective, the task formulation or the data used to train them.

In this paper, we consider fair behaviour in multi-agent systems. MARL is a framework that has many real-world applications which can mimic and solve real-world optimization problems that consider multi-agent interactions. However, utilizing certain common algorithms, like deep q-networks (DQN) or actor-critic, leads to the learning of an optimal policy that may result in the unfair behavior of agents; agents can learn to maximize their utility at the expense of others and take advantage of a free ride. In this paper, we aim to discuss possible approaches that incorporate fairness behaviors and pay attention in particular to the priority of agents in the systems. We aim to maximize the overall utility across the agents, while incorporating agents prioritised based on their types. Our contribution can be summarized as follows:

1. Propose possible approaches regarding agents' priority assignments in multi-objective problems using various approaches based on the task specification.
2. Define three case studies that represent real-world scenarios based on multi-agent interactions using the partial observable Markov decision-making process (POMDP) framework.
3. Show empirical results of case studies by applying various priority assignment strategies, where RL agents training was based on the offline reinforcement learning method.

2 Background

In this section, we formalize a multi-agent interaction in a partially observable Markov decision-making process (POMDP) framework and describe the offline reinforcement learning method that we are using to train our agents. We also review the ethical background behind the multi-agent interactions.

2.1 Partially Observable Markov Decision Process

We model the problem in the framework of a partially observable Markov decision problem (POMDP) which can be formalized by a tuple $(\mathcal{S}, O, \mathcal{A}, r, p, \gamma)$, where $\mathcal{S}$ denotes the set of states in the environment $s \in \mathcal{S}$, O represents the observation i.e., what the agent can see (agent does not have access to the entire state representation and thus holds only partial observability), and $\mathcal{A}$ refers to the set of actions that the agent can take $a \in \mathcal{A}$. The reward function $r : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$ determines the immediate reward. The transition probability $p : S \times A \times S$ characterizes the stochastic evolution of states in time $P(s_{t+1}|s_t, a_t)$. The constant $\gamma \in [0, 1)$ is a scalar discount factor. At each time step t, the agent takes action a. The agent learns a policy $\pi : O \rightarrow \mathcal{A}$ that maps observations to actions.

The RL objective, $J(\pi)$ can be written as an expectation under the trajectory distribution:

$$J(\pi) = \mathbb{E}_{\tau \sim p_\pi(\tau)} \left[\sum_{t=0}^{T} \gamma^t r(o_t, a_t) \right] \tag{1}$$

In multi-agent reinforcement learning (MARL), the objective of each agent is to learn an optimal policy to maximize its value function. Optimizing the v_π^j for agent j depends on the joint policy π of all agents, the concept of Nash equilibrium in stochastic games is therefore of great importance [20]. It is represented by a particular joint policy π_*, $[\pi_*^1, \ldots, \pi_*^N]$ such that for all $s \in \mathcal{S}$, $j \in \{1, ..., N\}$ and for all valid policies π^j of the j's agent it satisfies

$$v^j(s; \boldsymbol{\pi}_*) = v^j(s; \pi_*^j, \boldsymbol{\pi}_*^{-j}) \geq v^j(s; \pi^j, \boldsymbol{\pi}_*^{-j}).$$

Here we adopt a compact notation for the joint policy of all agents except j as $\boldsymbol{\pi}_*^{-j} = [\pi_*^1, \ldots, \pi_*^{j-1}, \pi_*^{j+1}, \ldots, \pi_*^N]$.

Given a Nash policy π_*, the Nash value function

$$\boldsymbol{v}^{Nash}(s) = [v_{\pi_*}^1(s), \ldots, v_{\pi_*}^N(s)]$$

is calculated with all agents following π_* from the initial state s onward.

It can be proved that under certain assumptions, the Nash operator H^{Nash} defined by the following expression

$$H^{\text{Nash}} \boldsymbol{Q}(s, \boldsymbol{a}) = \mathbb{E}_{s' \sim p}[\boldsymbol{r}(s, \boldsymbol{a}) + \gamma \boldsymbol{v}^{\text{Nash}}(s')] \quad [21] \tag{2}$$

forms a contraction mapping, where $\boldsymbol{Q} = [Q^1, \ldots, Q^N]$ and $\boldsymbol{r}(s, \boldsymbol{a}) = [r_1(s, \boldsymbol{a}), \ldots, r^N(s, \boldsymbol{a})]$.

2.2 Offline Reinforcement Learning

For our experiments we use offline reinforcement learning algorithm advantage-weighted regression (AWR) [22] to train the agents. Offline reinforcement learning is a method whose learning policy is based on the previously collected dataset which allows us to limit the number of interactions that the agent may have with the environment. This approach helps to apply this method to more complex real-world scenarios, where interacting with the environment can be challenging or time-consuming. In offline RL, the agent is provided with a static dataset of transitions $\mathcal{D} = \{(o_t, a_t, o_{t+1}, r_t)\}$ and it aims to learn the optimal policy using this dataset.

We use the dataset $\mathcal{D}$ to fit a value function $V_k^{\mathcal{D}}(s)$ to the trajectories by computing the Monte Carlo returns $R_{s,a}^{\mathcal{D}} = \sum_{t=0}^{T} \gamma^t r_t$ [21];

$$V_k^D(s) \leftarrow \arg\min_V \mathbb{E}_{s,a\sim\mathcal{D}} \left[||\mathcal{R}_{s,a}^{\mathcal{D}} - V(s)||^2 \right]$$

$$\pi_{k+1} \leftarrow \arg\max_\pi \mathbb{E}_{s,a\sim D} \left[\log \pi(a|s) \exp(\tfrac{1}{\beta}(\mathcal{R}_{s,a}^{\mathcal{D}} - V^{\mathcal{D}}(s))) \right]$$

2.3 Psycho-cultural Background

Previous research [23] has highlighted the primary relevant feature of an ethical theory as the ability to identify and order actions and their immediate outcomes across states of environments.

Some research [24] has introduced a function $C(u)$ that indicates whether the chosen reward function is the preferred ethical utility function for the agent to follow.

The cake or death example represents a number of possible unethical decisions that can result from an agent choosing actions following this rule or a variant of this rule [24]. In their work the authors suggest that an agent predicts its meta-utility function, represented as the linear combination of possible ethical utility functions. This prediction is based on changes from information gathering actions, which results in future sub-optimal decisions given its current meta-utility function. They suggest keeping the model for the probabilities of ethical utility functions independent from the model that predicts the world. This facilitates the possibility for the agent to predict observations that would inform what constitutes the correct ethical utility function, without simultaneously predicting that same ethical utility function.

The authors highlight that it is unclear how such an agent may be designed and whether satisfying those properties would allow for effective tradeoffs between learning about what is ethical and making ethical decisions.

Previous studies also include discussions on ethical principals in multi-agent settings applied to particular domains, e.g. social factors affecting climate change mitigation [25]. In their work, the authors highlight that different reward mechanisms that target different social factors could result in different emergent behaviors.

3 Methodology

In this section we introduce our priority assignment architecture, based on different prior information about the agents.

Proposed architecture is summarized in Figure 1.

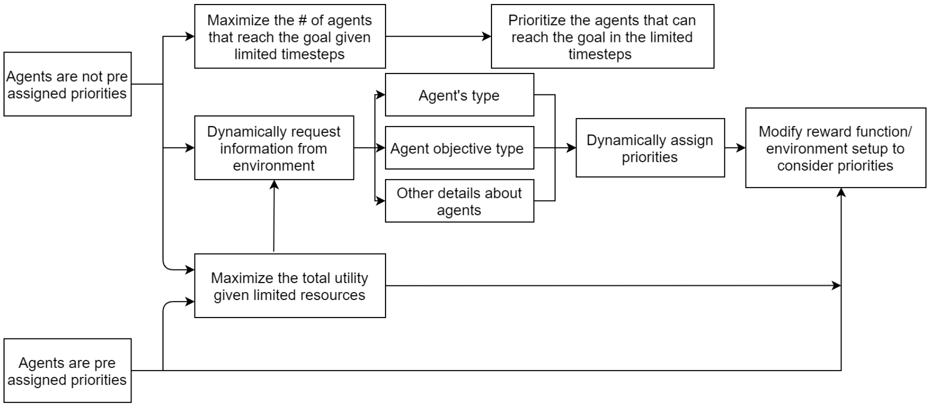

Agents are not pre assigned priorities
Maximize the # of agents that reach the goal given limited timesteps
Prioritize the agents that can reach the goal in the limited timesteps
Dynamically request information from environment
Agent's type
Agent objective type
Other details about agents
Dynamically assign priorities
Modify reward function/ environment setup to consider priorities
Maximize the total utility given limited resources
Agents are pre assigned priorities

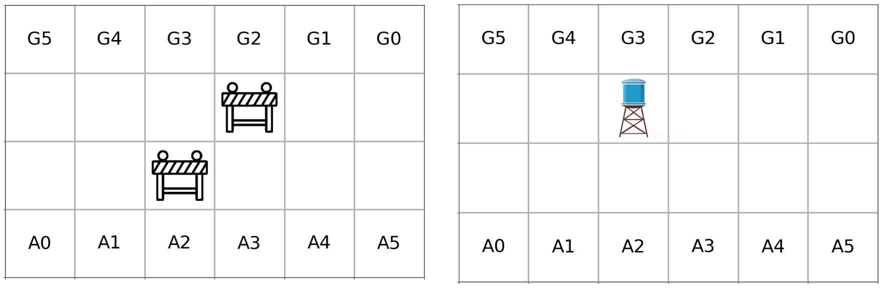
Fig. 1: Architecture for ensuring fairness. We consider scenarios in which the agents are pre assigned priorities and also scenarios in which they are not. We consider scenarios in which the agents maximize their utility given: limited timesteps, limited resources, and dynamically requested information from the environment.

We formulate a problem in a form of multi-agent multi-objective grid-world. The observation is discrete and represented as (x, y) coordinates of the grid. Action space represented in a discrete form where $A \in \{N, E, S, W\}$.

The environment consists of n-agents (A_0, A_1, ..., A_n), where each agent has its own goal $(G_0, G_1, ..., G_n)$. We consider a collaborative environment with multi-objective setup. In other words, A_i has to reach goal G_i. Agents cannot share the same grid location at any time-step t, such that $Li{}_t^{A_i} \neq Li{}_t^{A_j}$ (where Li represents grid location i), and where A_i, A_j are any two agents. Additionally, we introduce two blocks around which the agents must navigate. For the base problem setup we consider the same reward function for every agent as $R_0 = R_1, ..., = R_n$. More details can be viewed on Figure 2a.

4 Case Studies

We consider three case studies, each with various initial problem setups, propose solutions and show experimental results. Our agents were trained using a modified version of the AWR method.



(a) Multi-agent multi-objective problem setup. Here, (A_0, A_1, ..., A_5) represent the 6 agents and $(G_0, G_1, ..., G_5)$ represent their respective goals. There are two road blocks around which the agents must navigate.

(b) Here, (A_0, A_1, ..., A_5) represent 6 firetrucks and $(G_0, G_1, ..., G_5)$ represent their respective goals. There is a main water tank from which the agents must fill their tanks to extinguish the fires at their goal locations.

Fig. 2: Multi-agent multi-objective problem setup.

4.1 Case 1 – Priority based on the agent's type

Problem formulation For this case study, we consider a scenario in which agents belong to different types. The priorities are assigned based on the type of the agent. This type of setup is modeled after real-world scenarios in which certain participants are deemed more essential, e.g. in the domain of transportation. Certain transportation vehicles can carry a type of 'emergency designation' such as ambulances, police cars, fire trucks, as opposed to 'regular vehicles'. We can assume the priority of an agent based on its expected utility. For example, in the case of a large fire, a fire truck might have a more essential task to complete and thus will hold a higher priority over a delivery truck dropping off a package. We adjust our problem formulation in such a way that each agent is provided with a cumulative reward distributed across all agents. Thus, the agents' objectives are not limited to maximizing their individual rewards but rather are administered in order to maximize the cumulative reward. Thus, the agents will learn the overall optimal policy in such a way that the total utility across all agents is maximized.

Experimental Results In the typical formulation in which all agents have the same priority, we observe that some agents (e.g. $A3$ and $A4$) spend twice the amount of time to reach their goal, compared to their optimal path in the case of a single-agent setup, as shown in Figure 3. When we assign different priorities to the agents, we see that they reach their goal in a shorter amount of time, as other agents learn to give way to the agents with a higher priority in order to maximize the total utility across all agents.

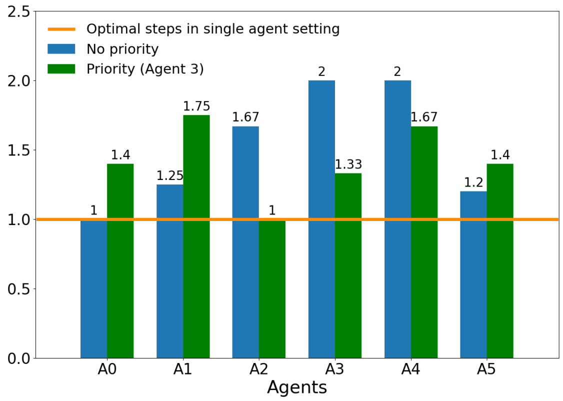


Fig. 3: Comparison of the number of steps taken by an agent in a prioritized vs non-prioritized setting. Values are represented as a fraction of an optimal number of steps taken in a single agent setting.

4.2 Case 2 – Priority based on the shortest path

Problem formulation In this scenario, we consider a problem setup with a limited time frame, so that agents with the shortest path to the goal will have priority over others. Agents must learn a strategy that maximizes the number of agents that achieve their goal within a limited time frame, even if it means that they themselves do not achieve their goal. We keep the rewards for all agents the same. This application may be somewhat analogous to airplane seat assignments based on the priority level of the purchased ticket. Thus, a passenger with a higher priority will reach their seat faster, while the seat location also typically is reached through a shorter path than that of a passenger with a lower priority ticket. So passengers with the seats located further down the plane cabin will have to give way and wait a certain amount of time before boarding.

Experimental Results Our environmental setup requires at least 6 time steps for all agents to achieve their goals. In the case in which the maximum number of time steps is 4, only three agents can achieve their goals (Figure 5a). Thus, some agents learn to give way to other agents even if it means they themselves won't achieve their goal. The path followed by the agents is shown in Figure 4

4.3 Case 3 – Priority based on limited resource availability

Problem formulation For this scenario, we consider a case in which agents' goals are designed to hold different levels of importance. In this scenario, we design the agents to represent firetrucks which have to fill up their water supply from a main water tank (Figure 2). However, there is only enough water for 4 firetrucks in the water tank. Priority is assigned based on how large of a fire each is designed to extinguish and the number of people that could be saved by reaching each goal. We assign priorities to agents A0, A1, A4, and A5 simulating

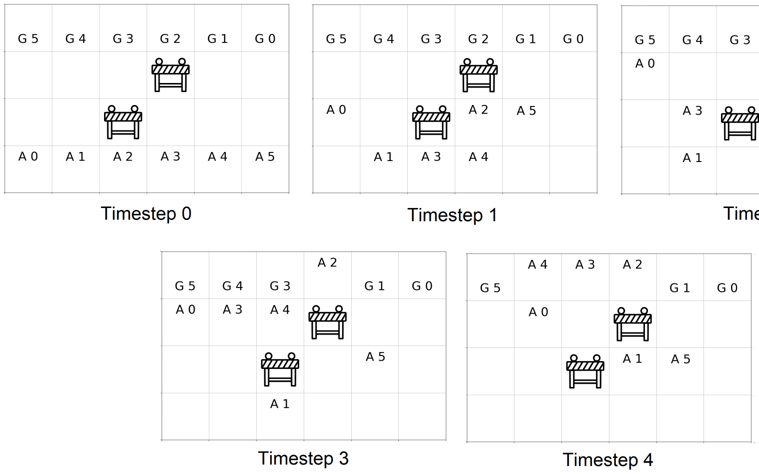


Fig. 4: Path followed by the agents. Agents A0, A1, and A5 learn to wait to maximize the overall number of agents reaching their goal even if they do not.

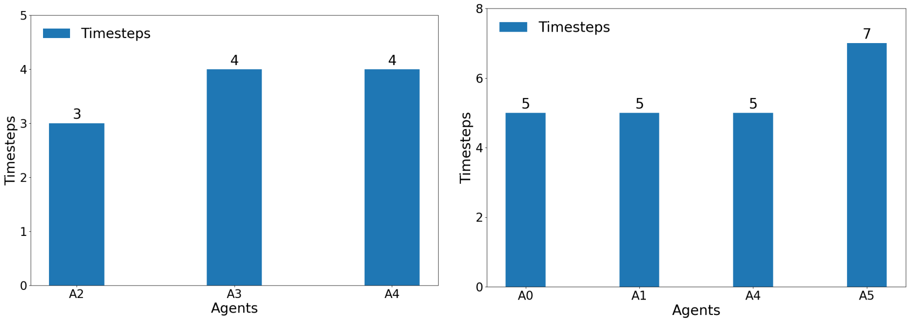
a scenario in which the agents farthest from their goals have the highest importance. Without assignment of these priorities the agents A1, A2, and A3 would achieve their goals quicker as per their distance to the tank and their goal.

Experimental Results As shown on Figure 5b agents A2 and A3 learn to let the other firetrucks fill up their tanks, as this action maximizes the number of people who can be saved.

5 Conclusion

In this paper, we consider the problem of fairness in the decision-making process within the field of reinforcement learning. By integrating approaches that incorporate fairness behaviors and investigating the priority of agents in the systems, this paper maximized the over-all utility across all agents. Our contribution first administered priorities based on agents' deference to the hierarchy of the groups' essential tasks, showing that agents were able to give way to higher-priority agent tasks. We then administered goal-seeking priorities based on the length of distance to the goal, after which time agents learned to defer to the goal-seeking of another, even if they missed their goal. Finally, we administered priorities based on the agents' ability to accomplish the goal, with results showing that agents that would be less capable of accomplishing the goal defer to others who are more capable.

Our results indicate that fairness is contextually determined in the context of reinforcement learning. To ensure that agents act fairly, the developer must integrate reward functions and task completion for each agent in the group. In other words, each agent must be made aware of the collective goals shared by their fellow agents and should be rewarded based on their deference to the needs of the group.



(a) Priority based on shortest path. (The maximum number of steps an agent can take is limited to 4 in this scenario.)

(b) Priority based on limited resource availability

Fig. 5: Number of steps taken by the agents who reach their goal.

References

1. H. Zhu, J. Yu, A. Gupta, D. Shah, K. Hartikainen, A. Singh, V. Kumar, S. Levine, The ingredients of real-world robotic reinforcement learning, arXiv preprint arXiv:2004.12570 (2020).
2. S. Levine, C. Finn, T. Darrell, P. Abbeel, End-to-end training of deep visuomotor policies, The Journal of Machine Learning Research 17 (1) (2016) 1334–1373.
3. I. Arel, C. Liu, T. Urbanik, A. G. Kohls, Reinforcement learning-based multi-agent system for network traffic signal control, IET Intelligent Transport Systems 4 (2) (2010) 128–135.
4. N. Jones, How to stop data centres from gobbling up the world's electricity, Nature 561 (7722) (2018) 163–167.
5. A. Vereshchaka, N. Kulkarni, Optimization of mitigation strategies during epidemics using offline reinforcement learning, in: International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction and Behavior Representation in Modeling and Simulation, Springer, 2021, pp. 35–45.
6. D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, et al., Mastering the game of go with deep neural networks and tree search, nature 529 (7587) (2016) 484–489.
7. H. Li, T. Wei, A. Ren, Q. Zhu, Y. Wang, Deep reinforcement learning: Framework, applications, and embedded implementations, in: 2017 IEEE/ACM International Conference on Computer-Aided Design (ICCAD), IEEE, 2017, pp. 847–854.
8. N. C. Luong, D. T. Hoang, S. Gong, D. Niyato, P. Wang, Y.-C. Liang, D. I. Kim, Applications of deep reinforcement learning in communications and networking: A survey, IEEE Communications Surveys & Tutorials 21 (4) (2019) 3133–3174.
9. T. Chu, J. Wang, L. Codecà, Z. Li, Multi-agent deep reinforcement learning for large-scale traffic signal control, IEEE Transactions on Intelligent Transportation Systems 21 (3) (2019) 1086–1095.

10. J. Cui, Y. Liu, A. Nallanathan, Multi-agent reinforcement learning-based resource allocation for uav networks, IEEE Transactions on Wireless Communications 19 (2) (2019) 729–743.
11. H. X. Pham, H. M. La, D. Feil-Seifer, A. Nefian, Cooperative and distributed reinforcement learning of drones for field coverage, arXiv preprint arXiv:1803.07250 (2018).
12. C.-H. Yen, Y.-C. Lee, W.-T. Fu, Improving the efficiency of allocating crowd donations with agent-based simulation model, in: International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction and Behavior Representation in Modeling and Simulation, Springer, 2017, pp. 248–253.
13. A. R. Srinivasan, F. S. N. Karan, S. Chakraborty, A study of how opinion sharing affects emergency evacuation, in: International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction and Behavior Representation in Modeling and Simulation, Springer, 2018, pp. 176–182.
14. A. Vereshchaka, W. Dong, Dynamic resource allocation during natural disasters using multi-agent environment, in: International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction and Behavior Representation in Modeling and Simulation, Springer, 2019, pp. 123–132.
15. A. Prasad, I. Dusparic, Multi-agent deep reinforcement learning for zero energy communities, in: 2019 IEEE PES Innovative Smart Grid Technologies Europe (ISGT-Europe), IEEE, 2019, pp. 1–5.
16. F. Yang, A. Vereshchaka, C. Chen, W. Dong, Bayesian multi-type mean field multi-agent imitation learning, Advances in Neural Information Processing Systems 33 (2020).
17. K. Lin, R. Zhao, Z. Xu, J. Zhou, Efficient large-scale fleet management via multi-agent deep reinforcement learning, in: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2018, pp. 1774–1783.
18. J. Z. Leibo, V. Zambaldi, M. Lanctot, J. Marecki, T. Graepel, Multi-agent reinforcement learning in sequential social dilemmas, arXiv preprint arXiv:1702.03037 (2017).
19. J. Perolat, J. Z. Leibo, V. Zambaldi, C. Beattie, K. Tuyls, T. Graepel, A multi-agent reinforcement learning model of common-pool resource appropriation, arXiv preprint arXiv:1707.06600 (2017).
20. J. Hu, M. P. Wellman, Nash q-learning for general-sum stochastic games, Journal of machine learning research 4 (Nov) (2003) 1039–1069.
21. R. S. Sutton, A. G. Barto, Reinforcement learning: An introduction, MIT press, 2018.
22. X. B. Peng, A. Kumar, G. Zhang, S. Levine, Advantage-weighted regression: Simple and scalable off-policy reinforcement learning, arXiv preprint arXiv:1910.00177 (2019).
23. A. Ecoffet, J. Lehman, Reinforcement learning under moral uncertainty, in: International Conference on Machine Learning, PMLR, 2021, pp. 2926–2936.
24. D. Abel, J. MacGlashan, M. L. Littman, Reinforcement learning as a framework for ethical decision making, in: Workshops at the thirtieth AAAI conference on artificial intelligence, 2016.
25. K. Tilbury, J. Hoey, The human effect requires affect: Addressing social-psychological factors of climate change with machine learning, arXiv preprint arXiv:2011.12443 (2020).