Cultivating Divergent Multi-Objective Expertise in Multiagent Systems via Expert Ensembles

Raghav Thakar*
Kagan Tumer
Collaboriative Robotics and Intelligent Systems (CoRIS) Institute, Oregon State University
Corvallis, Oregon, USA
thakarr@oregonstate.edu

ABSTRACT

Multiagent reinforcement learning (MARL) traditionally focuses on training autonomous agents to coordinate on a singular, well-defined goal. However, real-world applications often comprise multiple, often conflicting, objectives that require balancing a wide range of Pareto-optimal trade-offs. While recent multi-objective MARL (MOMARL) frameworks approximate this Pareto front by conditioning a multi-head actor network on a preference weight vector, this approach presents a key limitation. Burdening a monolithic actor network with representing both coordinated joint-policies and a continuum of topologically distinct objective trade-offs creates a severe representational bottleneck, leaving the network susceptible to destructive gradient interference. To mitigate this, we introduce Ensemble of Experts for Multi-Objective Multiagent Reinforcement Learning (E2M2). E2M2 replaces each agent's monolithic head with an ensemble of independent expert networks, governed by a preference-conditioned routing mechanism that outputs blending weights. This architecture allows individual experts to specialise in divergent behaviours, while the router learns to smoothly interpolate between them to consolidate the agent's action. Preliminary experiments in a continuous multi-objective MAMuJoCo task demonstrate that E2M2 discovers a more dominant and diverse Pareto front, on average achieving a 22% improvement in hypervolume compared to a parameter-matched baseline.

KEYWORDS

multi-objective multiagent reinforcement learning, mixture of experts, continuous control

1 INTRODUCTION

Multiagent learning addresses the fundamental challenges of training a system of autonomous agents to coordinate and perform complex tasks. In the past decade, multiagent reinforcement learning (MARL) has received significant attention from the broader agents community. While often motivated by real-world problems, most MARL methods assume a singular, well-defined goal to train for. On the contrary, most real-world problems are *multi-objective* in nature, comprising multiple, even conflicting, parallel objectives [14]. For instance, a fleet of autonomous cabs must minimise ride times while actively maximising passenger comfort and safety. Without explicit multi-objective considerations, traditional MARL approaches fall short in such settings.

MARL methods have accounted for multiple objectives via scalarising utility functions [10, 13]. Such utilities collapse vector-valued multi-objective rewards into a single term, which can then be optimised using traditional MARL methods to learn the optimal joint-policy. However, multi-objective problems are typically characterised by not a single, but a range of *Pareto-optimal* solutions, representing the breadth of unique and equally optimal trade-offs among the objectives. Learning joint-policies that provide this range of team behaviours is especially challenging, and requires dedicated multi-objective MARL (MOMARL) approaches.

MOMA-AC [2], a notable recent work, is an actor–critic MOMARL framework that elegantly addresses these challenges. MOMA-AC streamlines training by encoding all agents' policy in a single multi-head actor network, with each head corresponding to a distinct agent. Importantly, it conditions this network on a preference weight vector ω, which acts as a clear signal for the team behaviour. A preference-conditioned vector value critic evaluates the team's global state. Its vector value estimates are scalarised using the same preference vector ω to compute the signal used to update the actor.

While MOMA-AC has successfully trained agents on several multi-objective multiagent continuous control tasks, its actor network is burdened with representing not only multiple agents, but also multiple objectives. Such monolithic architectures may struggle to balance conflicting objectives that require topologically distinct policy representations, and thus become susceptible to destructive gradient interference.

To mitigate this interference, we present **Ensemble of Experts for Multi-Objective Multiagent Reinforcement Learning (E2M2)**. E2M2 decouples the representational burden by replacing each agent's monolithic actor head with an ensemble of independent 'expert' networks. These experts are governed by a routing mechanism that is conditioned on a preference weight vector and outputs blending weights.

E2M2 decouples and allows individual experts to specialise in divergent behaviours—such as maximising speed or conserving energy—while the router learns to smoothly interpolate between them. This work-in-progress paper outlines the E2M2 architecture and presents some preliminary findings. Our experiments show significant improvements in Pareto front hypervolume and coverage compared to the MOMA-AC baseline in a bespoke multi-objective MAMuJoCo environment.

*Corresponding author.

Proc. of the Adaptive and Learning Agents Workshop (ALA 2026), Aydeniz, Delgrange, Mohammedalamen, Yang (eds.), May 25 – 26, 2026, Paphos, Cyprus, https://alaworkshop2026.github.io/. 2026.

2 BACKGROUND

We first introduce the key ideas underlying multi-objective optimisation and cooperative MARL. We then combine these perspectives to formalise MOMARL, which serves as the basis for the setting considered in this paper.

2.1 Multi-Objective Optimisation

Many decision-making problems involve optimising k objectives in parallel. In such settings, the performance of a policy π is naturally captured by a vector-valued expected return $V^\pi \in \mathbb{R}^k$, rather than a single scalar. A common instantiation is the discounted return,

$$V^\pi \;=\; \mathbb{E}_\pi\left[\sum_{t=0}^{\infty} \gamma^t \mathbf{r}_t\right],$$

where $\mathbf{r}_t \in \mathbb{R}^k$ is the vector reward at time t and $\gamma \in [0, 1)$ is the discount factor.

As objectives may conflict, it is generally impossible to identify a single policy that maximises every component of V^π. Instead, solutions are assessed via *Pareto dominance*. A policy π (strictly) dominates another policy π' if it is no worse on all objectives and strictly better on at least one:

$$\forall i \in \{1, \dots, k\},\ V_i^\pi \geq V_i^{\pi'} \quad \text{and} \quad \exists j,\ V_j^\pi > V_j^{\pi'}.$$

The set of non-dominated policies constitutes the *Pareto-optimal set*, and their corresponding return vectors form the *Pareto front*.

A standard alternative is to reduce the vector return to a scalar by means of a *utility function* $u : \mathbb{R}^k \to \mathbb{R}$, which encodes stakeholder preferences over objectives [13]. A widely used choice is linear scalarisation with a weight vector ω,

$$u(V^\pi, \omega) \;=\; \omega^\top V^\pi.$$

However, scalarisation collapses the multi-objective problem into a single-objective one and typically presumes that preferences are fixed and known *a priori*. This can be restrictive when preferences are uncertain, context-dependent, or change over time [15]. Such approaches may still learn an approximation of the Pareto front by re-running their single-objective approach for various scalarisations. These approaches are termed *outer loop* approaches [5]. However, without significant information reuse and representation sharing, these methods may be too inefficient to produce a Pareto front with acceptable density [5].

Motivated by these considerations, we focus on methods that learn the Pareto front directly, also termed *inner loop* methods.

2.2 Multiagent Reinforcement Learning

We consider cooperative MARL in which a team of n agents $\mathcal{N} = \{1, \dots, n\}$ interacts with a shared environment. A convenient formalism is the decentralised partially observable Markov decision process (Dec-POMDP), in which the environment evolves over a global state space $\mathcal{S}$, while each agent i receives only a local observation $o_i \in \mathcal{O}$ and selects an action $a_i \in \mathcal{A}_i$. Joint behaviour is described by the joint action $\mathbf{a} = (a_1, \dots, a_n)$, and the goal in the cooperative case is to learn decentralised policies that coordinate effectively despite partial observability.

A central practical challenge in MARL is *non-stationarity*. As each agent updates its policy, the effective environment faced by the others changes. Contemporary methods commonly address this via *centralised training with decentralised execution* (CTDE) [3, 12]. Under CTDE, training leverages additional information—typically the global state s and joint action $\mathbf{a}$—within a centralised critic to stabilise learning and assign credit across agents. At execution time, however, each agent acts using only its own observation o_i, ensuring decentralised operation.

To improve parameter efficiency and encourage generalisation across agents, it is also common to share parameters across the team. One typical design uses a shared feature-extraction trunk with agent-specific output heads, producing distinct actions (or action distributions) for each agent [1, 2, 7].

2.3 Multi-Objective Multiagent Reinforcement Learning

MOMARL combines these two perspectives by requiring a cooperative multiagent team to optimise multiple objectives simultaneously. Formally, MOMARL can be expressed as a multi-objective Dec-POMDP (MO-Dec-POMDP) [2], specified by the tuple

$$\langle \mathcal{N}, \mathcal{S}, \mathcal{A}, \mathcal{O}, \mathcal{T}, \Omega, \mathcal{R}, \gamma \rangle.$$

The defining feature is the vector-valued reward function $\mathcal{R} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^k$, which induces a vector return and hence a vector value $V^\pi \in \mathbb{R}^k$ for any joint policy π. As a result, the learning objective is no longer to find a single optimal joint policy, but rather to characterise (or approximate) the set of Pareto-optimal joint policies and the associated Pareto front.

3 RELATED WORK

While intersection of multi-objective optimisation and MARL remains relatively under-explored, dedicated MOMARL algorithms have emerged recently to approximate the Pareto front directly.

3.1 MO-MIX

For discrete action spaces, MO-MIX [6] leverages the CTDE framework by conditioning decentralised agent networks on a preference weight vector. It estimates the joint action-value function by leveraging a parallel-track mixing network (inspired by Q-MIX [12]) that aggregates per-agent values into a joint multi-objective Q_{tot}. MO-MIX further proposes an exploration-guidance mechanism to improve the uniformity of the learned non-dominated set.

3.2 MOMA-AC

For continuous state and action spaces, MOMA-AC [2] introduced the first dedicated inner-loop actor–critic framework. MOMA-AC builds upon multiagent variants of TD3 [4, 16] and DDPG [8, 9], combining a multi-headed actor network with a centralised, vector-valued critic. MOMA-AC conditions both networks on a preference weight vector ω which acts as a unique, static utility in each episode. The centralised critic estimates the multi-objective value of a joint state and joint action (for a given preference vector), and this estimate vector of the critic is scalarised with ω to update the actor network. MOMA-AC samples the preference vector ω randomly at the start of each episode, pushing the critic to accurately estimate values in each region of the objective space, and the team to alter

its behaviour per the given preference. Importantly, with this formulation, MOMA-AC preserves the overestimation-mitigation of multiagent TD3.

While MOMA-AC demonstrates strong performance in continuous multiagent domains, its reliance on a single, shared actor to represent both, a coordinated joint-policy and a continuum of conflicting objective trade-offs creates a natural representation challenge. Concretely, a preference-conditioned multi-head actor must learn a mapping from (o_i, ω) to a_i that is simultaneously valid across agents and across the entire preference simplex. Gradients induced by different preferences may point in competing directions: updates that improve performance for one region of the Pareto front can degrade performance elsewhere, particularly when optimal behaviours require qualitatively different coordination behaviours (e.g., aggressive high-reward strategies versus conservative, safe strategies). Moreover, because the actor is trained through a scalarised objective $\omega^\top Q(s, \mathbf{a}, \omega)$, the magnitude and direction of the policy gradient may change with ω across a batch of samples.

These factors make the monolithic actor susceptible to destructive gradient interference, motivating our use of a routed ensemble of experts in E2M2 to decouple incompatible behaviours while still permitting smooth interpolation across preferences.

4 ENSEMBLE OF EXPERTS FOR MULTI-OBJECTIVE MULTIAGENT REINFORCEMENT LEARNING (E2M2)

We start with an identical learning framework to MOMA-AC. To prevent gradients associated with divergent objectives from destructively interfering within the actor's parameters, we introduce a Mixture of Experts (MoE) routing mechanism at the head of each agent. We condition the agent-specific router on the preference vector. Finally, during training, we introduce an entropy regularisation term to the actor's loss function that promotes a more thorough use of each agent's expert networks.

4.1 Preference-Conditioned Routing

Rather than mapping the shared observation features h_i directly to the agent's action a_i using a single agent-specific head, we equip each agent with an ensemble of M expert networks $\{E_{i,1}, E_{i,2}, \ldots, E_{i,M}\}$. Each expert maps the agent's features to a candidate continuous action, $E_{i,m}(h_i)$, while a lightweight routing network produces preference-dependent mixing weights that determine how these candidates are combined.

Concretely, the router for agent i takes only the preference vector ω as input and outputs a simplex-valued weight vector $\alpha_i \in \Delta^{M-1}$ via a softmax:

$$\alpha_i = \mathrm{softmax}(W_{g,i}\,\omega + b_{g,i}). \tag{1}$$

Conditioning on ω serves two purposes. First, it makes the selection of behavioural modes an explicit function of the desired trade-off, ensuring that changes in objective preference induce *predictable* changes in the policy. Secondly, by generating mixture weights (rather than directly emitting actions), the router can express a continuum of intermediate behaviours by smoothly interpolating between a small set of specialised experts.

The final action for agent i is then formed as a convex combination of the experts' outputs:

$$a_i = \sum_{m=1}^{M} \alpha_{i,m}\, E_{i,m}(h_i). \tag{2}$$

This encourages individual experts to specialise in distinct regions of the Pareto front (e.g., one expert capturing high-speed, high-energy behaviour, and another favouring conservative, energy-efficient control), whilst the preference-conditioned router learns to blend these modes as ω varies. In doing so, E2M2 decouples topologically distinct behaviours and mitigates potential destructive gradient interference that arises when a single monolithic head must represent all trade-offs.

4.2 Entropy Regularisation

A common failure mode in mixture-of-experts (MoE) policies is when the router quickly becomes near-deterministic and assigns almost all probability mass to a single expert. This defeats the purpose of maintaining multiple experts and can reduce both coverage and robustness. To discourage premature collapse, we add an entropy-based regulariser to the actor objective that encourages high-entropy routing distributions.

In implementation, we compute (for each agent) the negative entropy term $\sum_{m=1}^{M} \alpha_{i,m} \log(\alpha_{i,m})$, averaged over the batch, and add it to the actor loss with coefficient $\lambda > 0$. Since

$$\mathcal{H}(\alpha_i) \;=\; -\sum_{m=1}^{M} \alpha_{i,m} \log(\alpha_{i,m}),$$

minimising $\sum_m \alpha_{i,m} \log(\alpha_{i,m})$ is equivalent to maximising the entropy $\mathcal{H}(\alpha_i)$. The resulting actor loss is:

$$\mathcal{L}_{\text{actor}} = -\mathbb{E}\big[\omega^\top Q_\phi(s, \mathbf{a}, \omega)\big] \;+\; \lambda \sum_{i \in N} \mathbb{E}\left[\sum_{m=1}^{M} \alpha_{i,m} \log(\alpha_{i,m})\right], \tag{3}$$

where $Q_\phi(s, \mathbf{a}, \omega)$ is the multi-objective value estimate of the centralised critic for the joint state s, joint action $\mathbf{a}$, and preference vector ω. Finally, α_i denotes agent i's router output (Equation 1) and λ controls the strength of the regularisation. In practice, we use a small constant (e.g., 10^{-8}) inside the logarithm for numerical stability when $\alpha_{i,m}$ is close to zero.

5 PRELIMINARY EXPERIMENTS

5.1 Experimental Setup

We evaluate E2M2 on a bespoke multi-objective variant of the MA-MuJoCo (Multiagent MuJoCo) Ant-2x4 environment [11]. The task requires cooperative locomotion: the Ant has four legs controlled by two agents, each actuating two legs. Agent i observes the positions and velocities of the joints it controls, together with the position and velocity of the Ant's torso. Each agent outputs continuous joint torques for its assigned joints, and both agents receive the same team-wide reward.

We decompose the standard scalar reward into a two-objective return:

- **Objective 1 (Speed):** encourage forward locomotion and survival.

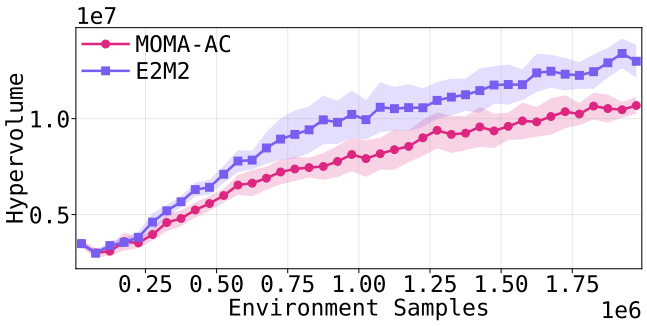

Figure 1: Evaluation hypervolume (Mean ± SEM) over training across 5 independent (reference point [0.0, -3000.0]). E2M2 converges faster and attains a higher final hypervolume than the parameter-matched MOMA-AC baseline.

- **Objective 2 (Efficiency):** encourage low actuation effort by maximising the negative control cost.

We compare E2M2 (with M=4 experts per agent) against the TD3-based variant of MOMA-AC [2]. To ensure a fair comparison, we enlarge the hidden-layer width of the MOMA-AC actor so that its total parameter count matches that of E2M2's mixture-of-experts policy. Both methods are trained for 2 million environment steps over 5 matched random seeds.

5.2 Results and Discussion

Hypervolume progression. Figure 1 reports the two-dimensional hypervolume of the Pareto front throughout training. Every 50,000 environment steps, we evaluate the current policy across 100 episodes, each conditioned on a different preference vector. Preferences are swept linearly along the 2D simplex from $[0.01,\ 0.99]$ to $[0.99,\ 0.01]$, and the resulting set of non-dominated returns is used to compute hypervolume.

Across training, E2M2 achieves higher hypervolume than MOMA-AC, with the gap emerging early and persisting to convergence, at which point it is 22% higher. We attribute the improved hypervolume to the expert ensemble: by allowing different experts to specialise in qualitatively distinct behaviours (e.g., fast but costly locomotion versus slower, more efficient control) and blending them via preference-conditioned routing, E2M2 demonstrates a stronger representational capability compared to MOMA-AC.

Pareto Front Coverage: Examining the final Pareto fronts (Figure 2), we observe that E2M2 discovers a front that is both, more dominant and more diverse than the parameter-matched MOMA-AC baseline. In particular, E2M2 produces many solutions that achieve substantially higher speeds (extending beyond the baseline's range) whilst maintaining comparable, and often better, energy efficiency. This qualitative difference aligns with our hypothesis: routing over multiple experts enables the policy to represent distinct locomotion modes and interpolate between them as the preference varies, rather than forcing a single monolithic head to compromise across conflicting gradients. As a result, E2M2 can express a richer family of Pareto-optimal coordination strategies, improving both hypervolume (Figure 1) and the practical range of attainable trade-offs.

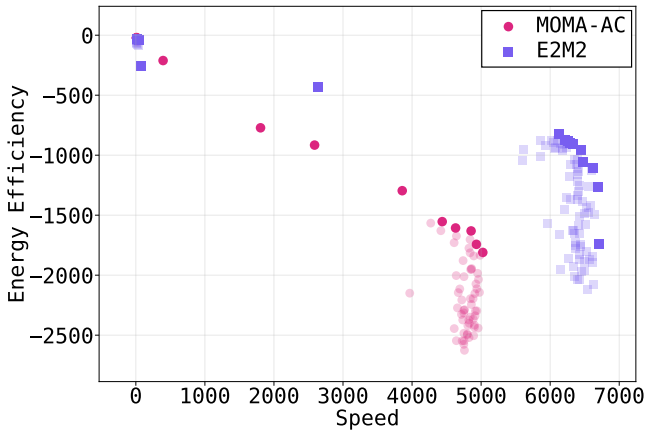

Figure 2: Comparison of the final discovered Pareto fronts at 2 million steps for the same seed. E2M2 finds a more dominant front with a wider spread of points.

6 CONCLUSION AND FUTURE WORK

This work-in-progress paper presented **E2M2**, a preference-conditioned mixture-of-experts architecture for continuous multi-objective multiagent reinforcement learning. Building on the CTDE actor–critic foundation of MOMA-AC [2], E2M2 introduces expert specialisation at each agent head and a preference-conditioned router that blends experts to produce a continuum of trade-offs. In a bespoke two-objective MAMuJoCo Ant task [11], E2M2 attains higher hypervolume and learns a more dominant, better-covered Pareto front than a parameter-matched MOMA-AC baseline, supporting the claim that structural decoupling can reduce destructive gradient interference in preference-conditioned multiagent policies.

Future work. The results presented in this paper are preliminary, and there are several clear next steps:

- **Handling distinct objectives.** Study settings where objectives demand markedly different (potentially topologically distinct) behaviours, and test whether the MoE approach enables a single preference-conditioned actor to optimise for each objective without collapsing. This includes constructing tasks with intentionally dissimilar objective optima and evaluating whether experts specialise into separable behavioural modes that remain high-performing across the full preference range.
- **Broader evaluation.** Test E2M2 across a wider suite of continuous multiagent control domains and MAMuJoCo configurations with varying coordination difficulty to assess generality.
- **Routing analysis and interpretability.** Analyse expert utilisation over training, and test whether routers consistently allocate different physical roles (e.g., front vs. back legs) to different experts under different preferences.
- **Robustness to changing preferences.** Evaluate settings where preferences shift within an episode, probing whether routing can adapt online without destabilising coordination.

Taken together, these directions aim to clarify when the MoE approach is most beneficial in MOMARL, and how expert specialisation can be exploited to learn richer and more reliable Pareto-optimal coordination strategies.

REFERENCES

[1] Ayhan Alp Aydeniz, Robert Loftin, and Kagan Tumer. 2023. Novelty Seeking Multiagent Evolutionary Reinforcement Learning. In *Proceedings of the Genetic and Evolutionary Computation Conference* (Lisbon, Portugal) *(GECCO '23)*. Association for Computing Machinery, New York, NY, USA, 402–410. https://doi.org/10.1145/3583131.3590428

[2] Adam Callaghan, Karl Mason, and Patrick Mannion. 2026. MOMA-AC: A preference-driven actor-critic framework for continuous multi-objective multi-agent reinforcement learning. *Neurocomputing* 664 (2026), 132032. https://doi.org/10.1016/j.neucom.2025.132032

[3] Jakob N. Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. 2018. Counterfactual multi-agent policy gradients. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence* (New Orleans, Louisiana, USA) *(AAAI'18/IAAI'18/EAAI'18)*. AAAI Press, Article 363, 9 pages.

[4] Scott Fujimoto, Herke van Hoof, and David Meger. 2018. Addressing Function Approximation Error in Actor-Critic Methods. In *International Conference on Machine Learning*. https://api.semanticscholar.org/CorpusID:3544558

[5] Conor F. Hayes, Roxana Rădulescu, Eugenio Bargiacchi, Johan Källström, Matthew Macfarlane, Mathieu Reymond, Timothy Verstraeten, Luisa M. Zintgraf, Richard Dazeley, Fredrik Heintz, Enda Howley, Athirai A. Irissappane, Patrick Mannion, Ann Nowé, Gabriel Ramos, Marcello Restelli, Peter Vamplew, and Diederik M. Roijers. 2022. A practical guide to multi-objective reinforcement learning and planning. *Autonomous Agents and Multi-Agent Systems* 36, 1 (April 2022), 59. https://doi.org/10.1007/s10458-022-09552-y

[6] Tianmeng Hu, Biao Luo, Chunhua Yang, and Tingwen Huang. 2023. MO-MIX: Multi-Objective Multi-Agent Cooperative Decision-Making With Deep Reinforcement Learning. *IEEE Trans. Pattern Anal. Mach. Intell.* 45, 10 (Oct. 2023), 12098–12112. https://doi.org/10.1109/TPAMI.2023.3283537

[7] Shauharda Khadka, Somdeb Majumdar, Santiago Miret, Stephen McAleer, and Kagan Tumer. 2020. Evolutionary reinforcement learning for sample-efficient multiagent coordination. In *Proceedings of the 37th International Conference on Machine Learning (ICML'20)*. JMLR.org, Article 617, 10 pages.

[8] Timothy P. Lillicrap, Jonathan J. Hunt, Alexander Pritzel, Nicolas Manfred Otto Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. 2015. Continuous control with deep reinforcement learning. *arXiv: Learning* (2015). https://api.semanticscholar.org/CorpusID:16326763

[9] Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. 2017. Multi-agent actor-critic for mixed cooperative-competitive environments. In *Proceedings of the 31st International Conference on Neural Information Processing Systems* (Long Beach, California, USA) *(NIPS'17)*. Curran Associates Inc., Red Hook, NY, USA, 6382–6393.

[10] Junlin Lu, Patrick Mannion, and Karl Mason. 2022. A multi-objective multi-agent deep reinforcement learning approach to residential appliance scheduling. *IET Smart Grid* 5 (05 2022), n/a–n/a. https://doi.org/10.1049/stg2.12068

[11] Bei Peng, Tabish Rashid, Christian A. Schroeder de Witt, Pierre-Alexandre Kamienny, Philip H. S. Torr, Wendelin Böhmer, and Shimon Whiteson. 2021. FACMAC: factored multi-agent centralised policy gradients. In *Proceedings of the 35th International Conference on Neural Information Processing Systems (NIPS '21)*. Curran Associates Inc., Red Hook, NY, USA, Article 934, 14 pages.

[12] Tabish Rashid, Mikayel Samvelyan, Christian Schroeder De Witt, Gregory Farquhar, Jakob Foerster, and Shimon Whiteson. 2020. Monotonic value function factorisation for deep multi-agent reinforcement learning. *J. Mach. Learn. Res.* 21, 1, Article 178 (jan 2020), 51 pages.

[13] Roxana Rădulescu, Patrick Mannion, Diederik M. Roijers, and Ann Nowé. 2019. Multi-objective multi-agent decision making: a utility-based analysis and survey. *Autonomous Agents and Multi-Agent Systems* 34, 1 (Dec. 2019), 52. https://doi.org/10.1007/s10458-019-09433-x

[14] Peter Vamplew, Benjamin J. Smith, Johan Källström, Gabriel Ramos, Roxana Rădulescu, Diederik M. Roijers, Conor F. Hayes, Fredrik Heintz, Patrick Mannion, Pieter J. K. Libin, Richard Dazeley, and Cameron Foale. 2022. Scalar reward is not enough: a response to Silver, Singh, Precup and Sutton (2021). *Autonomous Agents and Multi-Agent Systems* 36, 2 (Oct. 2022), 19. https://doi.org/10.1007/s10458-022-09575-5

[15] Peter Vamplew, John Yearwood, Richard Dazeley, and Adam Berry. 2008. On the Limitations of Scalarisation for Multi-objective Reinforcement Learning of Pareto Fronts. In *Proceedings of the 21st Australasian Joint Conference on Artificial Intelligence: Advances in Artificial Intelligence* (Auckland, New Zealand) *(AI '08)*. Springer-Verlag, Berlin, Heidelberg, 372–378. https://doi.org/10.1007/978-3-540-89378-3_37

[16] Fengjiao Zhang, Jie Li, and Zhi Li. 2020. A TD3-based multi-agent deep reinforcement learning method in mixed cooperation-competition environment. *Neurocomputing* 411 (2020), 206–215. https://doi.org/10.1016/j.neucom.2020.05.097