# DECAF: Learning to be Fair in Multi-agent Resource Allocation

Extended Abstract

Ashwin Kumar
Washington University in St. Louis
Saint Louis, MO, USA
ashwinkumar@wustl.edu

William Yeoh
Washington University in St. Louis
Saint Louis, MO, USA
wyeoh@wustl.edu

## ABSTRACT

A wide variety of resource allocation problems operate under resource constraints that are managed by a central arbitrator, with agents who evaluate and communicate preferences over these resources. We formulate this broad class of problems as *Distributed Evaluation, Centralized Allocation (DECA)* problems and propose methods to learn fair and efficient policies in centralized resource allocation. Our methods are applied to learning long-term fairness in a novel and general framework for fairness in multi-agent systems. Our methods outperform existing fair MARL approaches on multiple resource allocation domains, even when evaluated using diverse fairness functions, and allow for flexible online trade-offs between utility and fairness.

## KEYWORDS

Resource Allocation, Fairness, Multi-Agent RL

**ACM Reference Format:**
Ashwin Kumar and William Yeoh. 2025. DECAF: Learning to be Fair in Multi-agent Resource Allocation: Extended Abstract. In *Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Detroit, Michigan, USA, May 19 – 23, 2025*, IFAAMAS, 3 pages.

## 1 INTRODUCTION AND BACKGROUND

We look at a class of problems that we term *Distributed Evaluation, Centralized Allocation (DECA)*, which involves sequential decision making with distributed agents evaluating resources and a central decision maker enforcing constraints and maximizing total utility. To the best of our knowledge, thus far, researchers have investigated the different problems in this class separately and applied domain-specific approaches to solve them [3, 4, 8]. In this paper, we represent them with a unifying DECA formulation and propose fairness approaches that apply broadly to all DECA problems.

In DECA problems, each agent $i$ evaluates the utility of receiving any resources $a$ independently (Distributed Evaluation (DE)). This may also include some predictor of long-term value $Q(o_i, a)$ based on the current agent observation $o_i$, like a value function learned from experiences [7]. The arbitrator uses these valuations to allocate a limited set of indivisible resources to the agents to maximize total utility based on the following ILP (Centralized Allocation (CA)):

$$\max_{x_i(a)\in\{0,1\}} \sum_{i\in\alpha}\sum_{a\in A_i} x_i(a)Q(o_i,a) \tag{1}$$



*Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Y. Vorobeychik, S. Das, A. Nowé (eds.), May 19 – 23, 2025, Detroit, Michigan, USA.* © 2025 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org).

while satisfying resource and allocation constraints:

$$\sum_{a\in A_i, x_i(a)\in\{0,1\}} x_i(a) = 1, \quad \forall i \in \alpha, \quad \text{(Action Constraint)} \tag{2}$$

$$\sum_{a\in\mathcal{A}} c(a)_k \le \mathcal{R}_k, \quad \forall k \in \{1,\dots,K\}, \quad \text{(Resource Constraint)} \tag{3}$$

Here, $\mathcal{R} \in \mathbb{R}^K$ denotes the availability of $K$ different types of resources, and $c(a)_k$ is the resource consumption function denoting how many resources of type $k \in \{1, 2, \dots, K\}$ are consumed by action $a$. This is a general formulation that has seen use in domains ranging from ridesharing [8] to homelessness prevention [3].

Further, we present a framework to also learn to improve fairness in DECA problems. Previous work in fair multi-agent reinforcement learning [2, 5, 9, 11] has attempted to train independent agents to behave fairly in a cooperative setting. However, these approaches do not model the complexity of enforcing resource constraints and do not consider flexible utility-fairness trade-offs. When they do consider constraints (e.g., [5]), the solutions are domain-specific and myopic. *DECAF*, our approach for learning fairness in DECA problems, allows agents to learn long-term fairness, in addition to being able to trade off utility and fairness during execution.

A longer version of this paper is available on ArXiv [6].

## 2 METHODS

Formally, we seek to maximize a combined measure of system utility and fairness, represented as:

$$\max\ (1-\beta)\mathcal{U}_T + \beta\mathcal{F}_T \tag{4}$$

where $\mathcal{U}_T$ denotes the total utility at time $T$ and $\mathcal{F}_T$ represents the fairness measure, weighted by $\beta$. Our approach to account for fairness involves updating the value function associated with agent utilities $Q(o_i, a)$ to also capture fairness.

Value functions are often learned from Bellman updates [10], which capture the error in the predicted value and the actual value observed in the trajectory. We adapt a version of Double-Deep Q-Learning [1] modified to suit the DECA framework to learn using experience replay with centralized training. A transition $\tau = \langle \mathbf{o}, \mathcal{A}, \mathbf{r}_u, \mathbf{r}_f, \mathbf{o}' \rangle$ contains utility rewards $\mathbf{r}_u$ and fair rewards $\mathbf{r}_f$ for all agents in addition to the previous and next observations $(\mathbf{o}, \mathbf{o}')$ and the central allocation $\mathcal{A}$. Given a replay buffer $\mathcal{D}$, we want to minimize the loss function $J_\theta = \mathbb{E}_{\tau\sim\mathcal{D}} L(\delta(\tau))$, where $\delta(\tau)$ is the Bellman error of the transition $\tau$, and $L$ is the MSE loss.

We propose three approaches for integrating fairness:

- **Joint Optimization (JO):** A single estimator optimizes a weighted combination of fairness and utility.

$$\delta(\tau) = (1-\beta)\mathbf{r}_u + \beta\mathbf{r}_f + \gamma Q_\theta(\mathbf{o}') - Q_\theta(\mathbf{o}, \mathcal{A}) \tag{5}$$

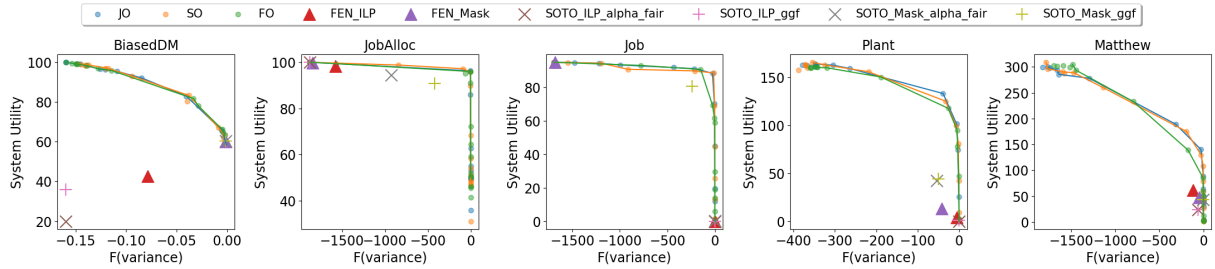



**Figure 1: Change in system utility and fairness as $\beta$ is increased, with $\beta = 0$ at the top left $\beta = 1$ at the bottom-right. For all domains, we can see that split and joint optimization perform similarly, while learning only fairness can sometimes be slightly worse. All our methods Pareto-dominate SOTO and FEN. Each point depicts the average performance over five different models trained at that $\beta$ value, and the lines show the Pareto front for each method.**

- **Split Optimization (SO):** Separate estimators for fairness ($F_\theta(\cdot)$) and utility ($U_\theta(\cdot)$) allow dynamic adjustment of their trade-off during policy execution.

$$\delta^f(\tau) = \mathbf{r}_f + \gamma F_\theta(\mathbf{o}') - F_\theta(\mathbf{o}, \mathcal{A}) \tag{6}$$

$$\delta^u(\tau) = \mathbf{r}_u + \gamma U_\theta(\mathbf{o}') - U_\theta(\mathbf{o}, \mathcal{A}) \tag{7}$$

$$Q(\mathbf{o}, \mathcal{A}) = (1-\beta)U_\theta(\mathbf{o}, \mathcal{A}) + \beta F_\theta(\mathbf{o}, \mathcal{A}) \tag{8}$$

- **Fair-Only Optimization (FO):** A fairness estimator ($F_\theta(\cdot)$) adjusts a pre-existing utility function $U^*(\cdot)$ to incorporate fairness, useful when utility functions are provided externally.

$$\delta^f(\tau) = \mathbf{r}_f(s, a) + \gamma F_\theta(\mathbf{o}') - F_\theta(\mathbf{o}, \mathcal{A}) \tag{9}$$

$$Q(\mathbf{o}, A) = (1-\beta)U^*(\mathbf{o}, \mathcal{A}) + \beta F_\theta(\mathbf{o}, \mathcal{A}) \tag{10}$$

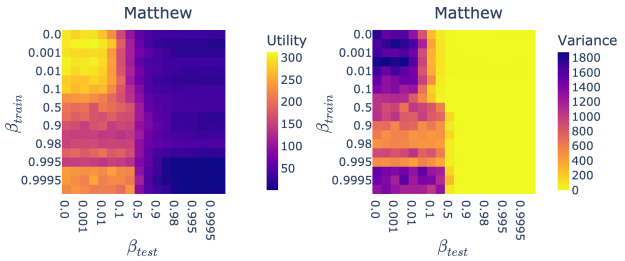
The fair rewards $\mathbf{r}_f$ are computed based on the difference in system fairness (based on agents' accumulated historical utilities) between the current and next state, decomposed to each agent. One simple decomposition is to equally divide the change in fairness across all agents. For our experiments, we use a stronger decomposition for the fairness objective of **minimizing variance in agent utilities**, calculated by attributing each agent's contribution to the change in variance.

## 3 RESULTS

We compare our methods to two baselines, FEN [2] and SOTO [11]. These methods rely on policy optimization, and output action probabilities rather than Q-values, and are not designed for environments with resource constraints. We adapt two variants for a fair comparison (1) Using the action probability as Q-values (**_ILP** suffix), and (2) Sequential allocation without the ILP by masking resources claimed by previous agents (**_Mask** suffix).

We convert some of the environments used by FEN and SOTO to make them DECA environments by adding resource constraints and a central decision maker. Our results are shown in Figure 1. Our methods clearly Pareto-dominate the baselines, in addition to providing diverse trade-offs based on the $\beta$ value selected. All three of our methods provide similar results, with FO being slightly worse at some intermediate $\beta$ values.

SO and FO further provide the additional flexibility of changing $\beta$ online to vary how fair or utilitarian the decisions are. This is possible because they both use two separate models, only combining them during inference to compute the fair-efficient Q-value (Eq. 8,10). This is illustrated for SO for the Matthew domain in Figure 2. We can see that despite the different $\beta_{train}$ used, each model improves fairness as $\beta_{test}$ is increased, and improves utility as it is decreased. This is a major strength of SO and FO.




**(a) System Utility** **(b) Variance**

**Figure 2: Evaluation of SO models trained on $\beta_{train}$ and evaluated on $\beta_{test}$ for the Matthew environment. Brighter colors indicate better outcomes.**

JO is useful when a single model is needed given a desired $\beta$. For flexibility, SO and FO are preferred, with FO being the better choice if a utility model is already known or if the utility function is a black box. SO is the best approach for learning both fairness and utility together, having the strengths of both JO and FO.

## 4 CONCLUSION

We proposed DECAF, a framework for learning long-term utility and fairness estimates in multi-agent resource allocation. DECAF is among the first approaches to optimize fair resource allocation under resource constraints, supporting diverse problem settings by decoupling fairness and utility metrics. Split and Fair-Only optimization enable online trade-offs between utility and fairness without retraining, enhancing interpretability. Our results demonstrate the flexibility and effectiveness of our approaches across various scenarios. DECAF is the first general approach for integrating fairness into constrained multi-agent resource allocation using Q-learning, paving the way for future advancements in fair AI decision-making.

## REFERENCES


[1] Hado van Hasselt, Arthur Guez, and David Silver. 2016. Deep reinforcement learning with double Q-Learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*. 2094–2100.

[2] Jiechuan Jiang and Zongqing Lu. 2019. Learning fairness in multi-agent systems. In *Proceedings of the Conference on Neural Information Processing Systems*. 13854–13865.

[3] Amanda R. Kube, Sanmay Das, and Patrick J. Fowler. 2019. Allocating interventions based on predicted outcomes: A case study on homelessness services. In *Proceedings of the AAAI Conference on Artificial Intelligence*. 622–629.

[4] Amanda R. Kube, Sanmay Das, and Patrick J. Fowler. 2023. Community-and data-driven homelessness prevention and service delivery: optimizing for equity. *Journal of the American Medical Informatics Association* 30, 6 (2023), 1032–1041.

[5] Ashwin Kumar, Yevgeniy Vorobeychik, and William Yeoh. 2023. Using Simple Incentives to Improve Two-Sided Fairness in Ridesharing Systems. In *Proceedings of the International Conference on Automated Planning and Scheduling*. 227–235.

[6] Ashwin Kumar and William Yeoh. 2025. DECAF: Learning to be Fair in Multi-agent Resource Allocation. arXiv:2502.04281 [cs.LG] https://arxiv.org/abs/2502.04281

[7] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. 2013. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602* (2013).

[8] Sanket Shah, Meghna Lowalekar, and Pradeep Varakantham. 2020. Neural approximate dynamic programming for on-demand ride-pooling. In *Proceedings of the AAAI Conference on Artificial Intelligence*. 507–515.

[9] Umer Siddique, Paul Weng, and Matthieu Zimmer. 2020. Learning fair policies in multi-objective (deep) reinforcement learning with average and discounted rewards. In *Proceedings of the International Conference on Machine Learning*. 8905–8915.

[10] Richard S. Sutton and Andrew G. Barto. 1998. *Reinforcement Learning: An Introduction*. The MIT Press, Cambridge, MA.

[11] Matthieu Zimmer, Claire Glanois, Umer Siddique, and Paul Weng. 2021. Learning fair policies in decentralized cooperative multi-agent reinforcement learning. In *Proceedings of the International Conference on Machine Learning*. 12967–12978.