# MAX-POLICY SHARING FOR MULTI-AGENT REINFORCEMENT LEARNING IN AUTONOMOUS MOBILITY ON DEMAND

Ebtehal T. Alotaibi[1,2] and Michael Herrmann[1]

[1] Institute of Perception, Action and Behaviour, University of Edinburgh, Edinburgh, UK
[2] Computer Science Department, Imam Mohammad Ibn Saud Islamic University, SA

***ABSTRACT***

*Autonomous-Mobility-on-Demand (AMoD) systems can revolutionize urban transportation by providing mobility as a service without car ownership. However, optimizing the performance of AMoD systems presents a challenge due to competing objectives of reducing customer wait times and increasing system utilization while minimizing empty miles. To address this challenge, this study compares the performance of max-policy sharing agents and independent learners in an AMoD system using reinforcement learning. The results demonstrate the advantages of the max-policy sharing approach in improving Quality of Service (QoS) indicators such as completed orders, empty miles, lost customers due to competition, and out-of-charge events. The study identifies the importance of striking a balance between competition and cooperation among individual autonomous vehicles and tuning the frequency of policy sharing to avoid suboptimal policies. The findings suggest that the max-policy sharing approach has the potential to accelerate learning in multi-agent reinforcement learning systems, particularly under conditions of low exploration.*

***KEYWORDS***

*Mulit-Agents, Reinforcement Learning, Consensus Learner, Max-Policy Sharing, Autonomous Mobility on Demand.*

## 1. INTRODUCTION

Autonomous Mobility on Demand (AMoD) systems have garnered significant attention in recent years, as they present a promising solution for improving urban transportation efficiency and reducing environmental impact. The optimal fleet size for an Autonomous Mobility on Demand (AMoD) system is a crucial factor in its overall efficiency and profitability. The fleet size needs to be big enough to meet the expected demand for the service, but not so large that it results in idle vehicles and unnecessary costs. In AMoD system, travelers can be picked up at any time and location by AVs and transported to where they need to go without owning a car, they would purchase mobility as a service. The performance of Autonomous Mobility on Demand (AMoD) systems is significantly influenced by the contradictory objectives of reducing customer wait times and increasing system utilization while minimizing empty miles. These conflicting goals can lead to competing and cooperating behavior among individual autonomous vehicles (AVs). In an effort to reduce wait times, AVs may adopt a competitive approach, aiming to be closer to areas with high expected demand, and thus positioning themselves to quickly respond to incoming requests. This competition, however, can result in an oversupply of vehicles, leading to

David C. Wyld et al. (Eds): ICDIPV, CBIoT, ICAIT, WIMO, NC, CRYPIS, ITCSE, NLCA, CAIML -2023
pp. 423-434, 2023. CS & IT - CSCP 2023 DOI: 10.5121/csit.2023.131333

higher capital expenditures, lower utilization rates, and increased empty miles as AVs roam in search of passengers. Conversely, promoting system utilization and reducing empty miles requires a more cooperative behavior among individual AVs, with an emphasis on optimizing fleet distribution and rebalancing vehicles based on demand patterns. While this cooperation can lead to better overall system efficiency, it might also result in longer wait times for customers as fewer vehicles are immediately available in high-demand areas. Boosting the right balance between these competing priorities is critical for optimizing AMoD performance, ensuring a high level of service for users and efficient use of resources.

Multi-agent reinforcement learning (MARL) offers a framework for addressing this complex problem by enabling AVs to learn to balance between competition and cooperation and adapt their behavior in a dynamic environment. The problem is formulated as a fleet of autonomous vehicles serves passenger demands by optimizing customer wait times, system utilization, and minimizing empty miles. The key challenge is to strike the right balance between competition and cooperation among individual AVs to achieve optimal AMoD performance. The agents a state space defined by location, time, battery level, and bidding value. Agents can perform four actions: take order, recharge, bid, and no operation. They must decide when to negotiate with nearby agents for orders, with the lowest bidder (shortest distance) winning the order.

## 2. BACKGROUND

As our research contributes to both the Autonomous Mobility on Demand (AMoD) literature and the Multi-Agent Reinforcement Learning (MARL) community, this section is divided into two parts to highlight the unique contributions to each area of study.

### 2.1. Multi-Agent Reinforcement Learning (MARL)

According to the learning architecture [1], MARL can be categorized into the following subtypes.

#### 2.1.1. Centralized training centralized execution (CTCE)

It models a joint policy that maps observations to individual actions under a set of distributions. With the CTCE paradigm, multi-agent problems can be directly addressed with single-agent methods such as actor-critics [2] or policy gradient algorithms [3]. However, by increasing the number of agents, state-action spaces grow exponentially. Individual policies for each agent can be formulated to overcome the so-called curse of dimensionality.

#### 2.1.2. Distributed Training Decentralized Execution (DTDE)

Where each agent has an assigned policy which maps local observations to individual actions. There is no sharing of information among agents, so each agent learns independently. Since agents lack access to the knowledge of others and do not perceive joint actions, the DTDE paradigm has the fundamental flaw that the environment appears non-stationary from their perspective.

Independent Learners are agents that learn their own policies separately without explicitly considering the presence of other agents. Each agent is trained using its own local observations and rewards, and they don't share information with each other during the learning process.

#### 2.1.3. Centralized Training Decentralized Execution (CTDE)

Where each agent holds an individual policy that maps local observations to a distribution of individual actions. Agents are provided with additional information during training, which is discarded during testing. The CTDE paradigm presents the state-of-the-art practice for learning

with multiple agents [4-5]. Sharing mutual information can ease the training process and increase learning speed when compared against independently trained agents. When all agents have access to additional information during training, they can avoid non-stationarity.

According to [6], the authors have classified the CTDE further into a joint-action learners (Fully Observable Critic, Value Function Factorization), and individual-action learners (Consensus algorithms and Learn to Communicate). While the Independent Learners in DTDE involve distributed training and decentralized execution, with each agent learning separately. the Fully Observable Critic and Value Function Factorization techniques involve centralized training with a shared critic, but they differ in how they represent the joint value function. Fully Observable Critic directly learns a joint action-value function, while Value Function Factorization decomposes it into simpler components. Both approaches use decentralized execution, where agents act independently based on their learned policies. Consensus algorithms focus on reaching agreements among agents, while Learn to Communicate approaches emphasize learning communication strategies for better coordination. Both can involve centralized or distributed training, depending on the specific implementation, and both use decentralized execution.

In cooperative multi-agent deep reinforcement learning, multiple agents work together to solve a common problem or achieve a shared goal. A consensus approach in this context refers to methods and algorithms used to achieve agreement among agents on certain aspects of their learning or decision-making process. This agreement helps the agents to learn more effectively and efficiently, as they can share information, coordinate actions, and make better decisions based on the collective knowledge and experience of the group.

Differentiable Inter-Agent Learning (DIAL) is an end-to-end learning approach proposed in [7], which means that the agents' actions and communication strategies are learned simultaneously. The authors show that DIAL can effectively train agents to cooperate and communicate in complex tasks, outperforming traditional reinforcement learning methods and other multi-agent learning techniques. However, it is not clear how the DIAL would adapt to dynamic environments where agents need to update their strategies and communication as the environment changes.

In their study, Arshavskaya et al. [8] examine a scenario in which each agent gathers local observations, applies its individual policy, and receives a unique reward. They introduce a tabular policy optimization consensus algorithm that utilizes Boltzmann's law (resembling the soft-max function) to address this challenge. The agreement algorithm assumes that an agent can share its local reward, a count of observations, and the action taken for each observation with nearby agents. The algorithm aims to maximize the weighted average of local rewards. In doing so, the researchers guarantee that each agent's learning is on par with what a centralized learner could achieve, ultimately reaching a local optimum.

## 2.2. MARL for Autonomous Mobility on Demand

Enders et al. [9] presented a technique to tackle the autonomous mobility on demand (AMoD) challenge by integrating a multi-agent deep reinforcement learning (DRL) algorithm for making proactive request assignment and rejection decisions. The main goal is to enhance the operating profit of an AMoD operator. The key difference between our approach and this study is their application of CTDE with value decomposition, where agents gradually learn advantageous joint actions over time. On the other hand, our strategy recommends independent learners possessing a communication channel to benefit from the contributions of other agents. This is expected to result in an increased learning rate and a broader exploration of experience areas.

In [10] study, the authors developed a real-time control policy based on deep reinforcement learning to operate an AMoD fleet of vehicles and determine ride pricing. The real-time control

policy simultaneously makes decisions for: 1) vehicle routing to serve passenger demand and rebalance empty vehicles, and 2) ride pricing to adjust potential demand, stabilizing the network and maximizing profits. The problem is defined as a multi-agent setup, where each part of the transportation network node represents a single agent managing its vehicles and orders, aiming to optimize the joint action in a CTDE structure. The authors compared their results with a single-agent DTDE approach and observed significant improvement in the multi-agent model.

Additionally, certain studies concentrate on addressing specific issues within the AMoD domain. Fluri et al. [11] discusess the rebalancing problem of AMoD fleet. They derive a cascade-based reinforcement learning model that captures the crucial spatio-temporal features of the rebalancing problem and defines a state-action space, ensuring relatively fast and stable convergence.

In [12] the paper explores a strategic charging pricing scheme for charging station operators (CSOs) using a non-cooperative Stackelberg game framework. The proposed framework formulates the AMoD operator's responsive actions (order-serving, repositioning, and charging) as a multi-commodity network flow model to tackle an energy-aware traffic flow problem. Concurrently, a soft actor-critic-based multi-agent deep reinforcement learning algorithm is developed to address the proposed equilibrium framework, considering privacy-preservation constraints among CSOs.

Reference [13] considers the problem of uncertainty in the EV rebalancing and charging.It develops a constrained multi-agent reinforcement learning (MARL) framework. Then introduces a robust and constrained MARL algorithm (ROCOMA) that trains an EV rebalancing policy to balance the supply-demand ratio and charging utilization rate across the entire city under state transition uncertainty.

## 3. PROBLEM FORMULATION

We model the environment as a Markov Decision Process (MDP) for each agent. The state space is defined by a tuple (*l: location, t:time, e:battery level, b:bidding value)*, where *location* represents the agent's current position, *time* is the current time step, *battery level* indicates the remaining energy of the agent, and *bidding value* is the agent's current bid for a task. The action space consists of four actions: *take order*, *recharge*, *bid*, and *no operation.*

Agents need to decide when to negotiate with other agents in the vicinity to complete announced orders. The negotiation process involves each agent bidding based on the distance from their current location to the order's location. The agent with the lowest bid (i.e., the shortest distance) is awarded the order. The reward associated with each action is contingent on the agent's bidding and positional status. If the agent has placed a bid and the order remains unclaimed, or if the agent's distance from the order location is less than the announced bid, the agent receives a reward of +1 for taking the order and -1 for taking the order in other cases. The bid value is updated to 1 exclusively when the agent places a bid for a particular order. To expedite the negotiation process, the agent is not allowed to re-bid for the same order. The agent experiences a penalty of -1 for each time step during its operation. Additionally, the total reward is adjusted based on the agent's current battery level (e), a process referred to as depreciation. This depreciation factor reflects how the reward value decreases in relation to the current battery level.

$$R(s, a = take\text{-}order) = \begin{cases} +1, if\ s[b] = 1 \\ -1\ if\ s[b] = 0 \end{cases}$$

$$R(s, a = recharge) = R_{no\text{-}op} = -1$$

$$R(s, a = bid) = \begin{cases} +1, if\ s[b] = 0 \\ -1\ if\ s[b] = 1 \end{cases}$$

$$R(s,a) = R(s,a) - \left\| |R(s,a)| - \left| R(s,a) \times \frac{e}{100} \right| \right|$$

### 3.1. Multi-Agent Formulation

For N = {1, … , N}, denotes the set of interacting agents, the Markov decision process is formalized by the tuple for each $i \in N$: $<\mathcal{S}^i, \mathcal{A}^i, \mathcal{P}^i, \mathcal{R}^i, \gamma>$, where $\mathcal{S}^i, \mathcal{A}^i$ are the state and action space, respectively, $\mathcal{P}^i : \mathcal{S}^i \times \mathcal{A}^i \rightarrow \mathcal{P}^i(\mathcal{S}^i)$ is the transition function describing the probability of a state transition, $\mathcal{R}^i : \mathcal{S}^i \times \mathcal{A}^i \times \mathcal{S}^i \rightarrow \mathbb{R}$ is the reward function providing an immediate feedback to the agent, and $\gamma \in [0, 1)$ describes the discount factor. The agent's goal is to act in such a way as to maximize the expected performance on a long-term perspective with regard to an unknown transition function $\mathcal{P}^i$. A policy $\pi$ is a distribution over actions given states,

$$\pi^i(a,s) = \mathcal{P}^i[\mathcal{A}^i_t = a, \mathcal{S}^i_t = s]$$

In a similar manner, the action-value function $Q_\pi^i : \mathcal{S}^i \times \mathcal{A}^i \rightarrow \mathbb{R}$ describes the utility of being in state $s$, performing action $a$, and following the policy $\pi^i$ thereafter, that is;

$$Q_\pi^i(s, a) = \mathbb{E}_{s_{t+1} \sim P^i, a_t \sim \pi^i} \left[ \sum_{t=0}^{\infty} \gamma^t \; R(s,a)^i \right]$$

Epsilon-greedy (ϵ-greedy) is a widely-used Q-learning approach to balance exploration and exploitation. It selects the optimal action with a probability of (1-epsilon) and a random action with probability epsilon. A higher epsilon promotes exploration, while a lower epsilon favors exploitation.

We have used epsilon-greedy exploration, i.e. an agent selects the action with the highest estimated Q-value with a probability of (1-ϵ), while performing an exploratory move probability of ϵ.

$$\pi^{i*}(s,a) = \begin{cases} 1, if\ a = argmax_{a \in A^i} Q^i_{\pi^*}(s, a) \\ 0, \qquad\qquad otherwise \end{cases} \quad (1)$$

During the initial phase of learning, it is often useful to set a higher epsilon value to encourage exploration and learn about the environment. As the agent gains more knowledge, the epsilon value can be gradually decreased to favor exploitation and improve performance.

From the theorem, we can find an optimal policy immediately by maximizing $Q^i_{\pi^*}(s, a)$ over all actions.

The consensus approaches in multi-agent reinforcement learning can vary widely depending on the specific algorithm, communication method, and problem domain. Agents can exchange information about their observations, actions, and rewards to help each other learn more quickly and accurately. This can involve sharing entire Q-value tables, gradients, or other learning-related data. The epsilon definition used in epsilon-greedy as an exploration-exploitation indicator can also be repurposed as a sharing rate in the max-policy sharing approach of multi-agent reinforcement learning. A higher epsilon value corresponds to a higher willingness to explore new actions and thus, a lower willingness to share Q-value tables with other agents. Conversely, a lower epsilon value corresponds to a greater emphasis on exploiting the current knowledge and thus, a higher willingness to share Q-value tables with other agents.

In the max-policy sharing approach, after enough exploration, the agents share their Q-value tables with each other. The sharing mechanism can be implemented in different ways, such as averaging, max-pooling, or a custom function that combines the information from the Q-value tables. By combining the exploration-exploitation indicator in epsilon-greedy with the sharing

rate in max-policy sharing, agents can achieve a balance between cooperation and competition among agents, leading to improved learning outcomes in multi-agent reinforcement learning. The sharing rate can be adjusted by varying the epsilon value to control the degree of exploration and exploitation in the learning process.

**Definition:** In the Distributed training decentralized execution (DTDE), The Consensus Learner is voting to the best policy as;

$$\overline{\mathcal{M}}\ (\mathrm{s}, \mathrm{a}) = \overline{\sum_{i=0}^{N} \pi^{i*}(s,a)}$$

Where $\mathcal{M}$ calculate the frequency of selecting action a for a state s by all optimal policies. Therefore, the optimal policy in (1) will be calculated as;

$$\overline{\pi^{i*}(s) = argmax_{a \in A^i}\ \mathcal{M}\ (s,a)}$$

By using max-pooling, the agents can benefit from the best Q-value estimates learned by any agent in the group, potentially accelerating their learning process. However, it is essential to balance the frequency of sharing and exploration to avoid premature convergence to suboptimal policies.

## 4. RESULTS AND DISCUSSION

The main objectives of our experiments are to verify the following hypotheses: 1) The recommended max-policy sharing strategy for consensus learners can speed up policy learning in comparison to independent techniques. 2) Homogeneous teams benefit from exchanging learned policies rather than augmenting their numbers throughout the training process. Furthermore, we aim to explore the effects of max-policy sharing MARL on AMoD indicators. As rewards increase, we expect a rise in the positive aspects of the reward function, such as completed orders and rebalancing, while observing a decline in negative aspects like empty miles and recharging.

A range of experiments were carried out to compare max-policy consensus learners with independent learners. These experiments were split into two groups: the first group focused on analyzing the performance of MARL agents, while the second group discussed the results from an AMoD perspective.

### 4.1. Mutl-Agent Reinforcement Learning Results

In this series of experiments, the average reward value per episode for agents is presented during the training process. Consensus learners undergo training with 50 orders, each lasting for 4 timesteps, while independent learners are trained with 100 orders, also lasting for 4 timesteps.

The experimental results in Figure 1 demonstrate the advantages of employing the max-policy sharing approach in multi-agent reinforcement learning systems under conditions of low exploration. When comparing the performance of independent learners with max-policy sharing learners, we observed that the latter achieved superior rewards in fewer episodes, indicating a more efficient learning process. The improved performance of max-policy sharing learners can be attributed to the strategic timing of initiating the policy sharing mechanism. By starting policy sharing only when the exploration rate (epsilon) is low, we ensure that agents have already undergone substantial learning and refined their policies to a point where they are more likely to produce valuable information for their peers. Consequently, the shared policies reflect higher-quality action choices, which, in turn, enable the receiving agents to improve their performance more effectively.

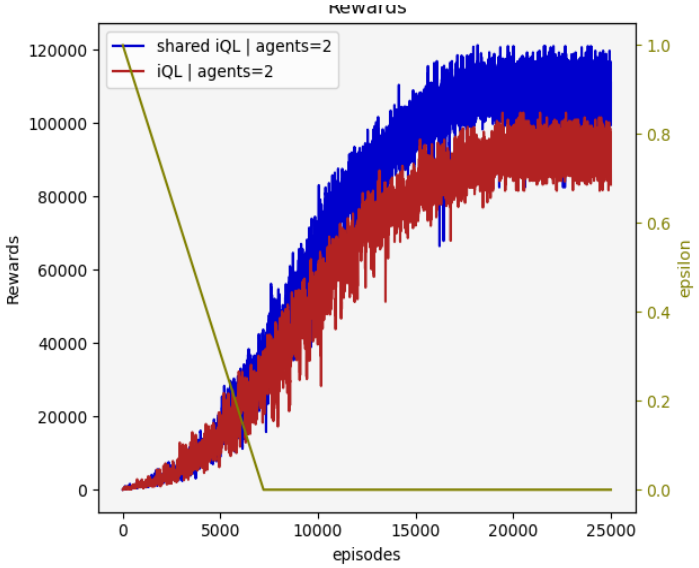


Figure 1 Comparison of average rewards between Max-policy sharing learners and independent learners, illustrating the impact of epsilon (ϵ) value on sharing rate and exploration-exploitation balance. A higher ϵ value leads to lower sharing rate and higher exploration, while a lower epsilon value results in higher sharing rate and higher exploitation.

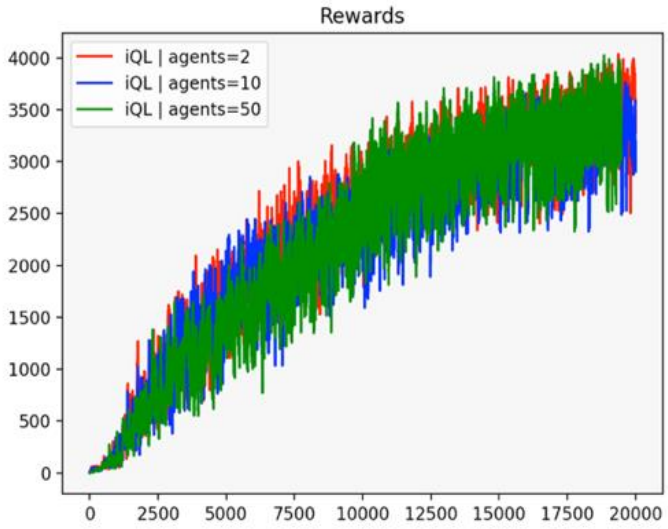


Figure 2 Comparison of the learning stage between independent learners on the same task, as the number of learners is increased. The learning stage does not further improve as each independent learner learns separately and increasing their number does not result in any collective learning.

Moreover, our empirical results in Figure 2 show that independent learners do not benefit from an increased number of agents, as the average rewards remain similar for groups of 2, 10, and 50 agents. This observation suggests that simply adding more agents to the learning process does not lead to better performance. In contrast, the max-policy sharing approach allows agents to leverage the collective knowledge of their peers, leading to more efficient learning and improved overall performance. The reduced exploration rate also plays a crucial role in the success of the

max-policy sharing approach. Lower epsilon values indicate that agents have already conducted extensive exploration of the environment and are thus more focused on exploiting their acquired knowledge. In this context, sharing policies can lead to more informed action selection, as the agents are primarily voting on the most promising actions learned by their peers.

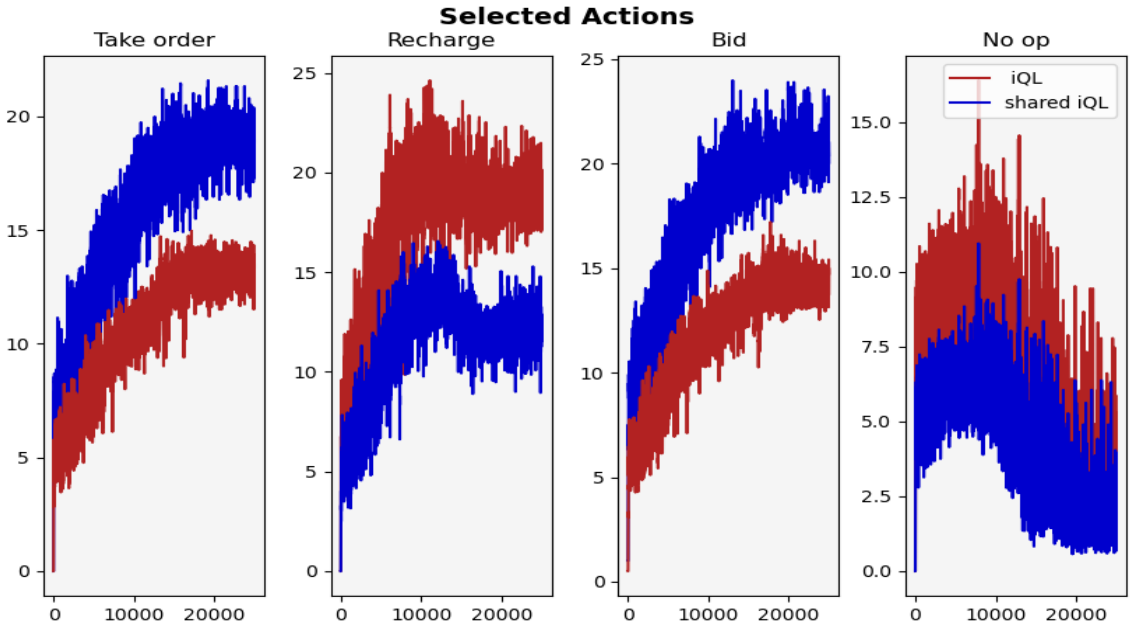


Figure 3 Action frequency distribution of Max-policy sharing learners during learning

Figure 3 illustrates the frequency of action selection throughout the learning process. As anticipated, max-policy sharing agents excel over independent learners in terms of the frequency of take order and bid actions, mainly due to the accelerated learning achieved through the sharing of policies. Furthermore, an interesting trend is observed in the recharge action frequency for both types of agents. At the beginning of the learning process, the frequency of the recharge action increases, indicating that the agents are moving and consuming battery while exploring the environment and learning the optimal policy. As learning progresses, the frequency of the recharge action starts to decrease slowly, which shows that the agents are adapting their strategies to balance between completing orders while maintaining battery levels. The final figure presents the decreasing frequency of the no operation action, which suggests a more productive and efficient team of agents. Thus, higher utilization. As agents learn to make more informed decisions and take appropriate actions, resulting in a more active and coordinated team.

## 4.2. MARL for AMoD Results

In this section, we have emphasized the influence of trained agents on the indicators of AMoD, specifically the mean number of orders completed per episode, the total number of miles traveled without passengers, and the utilization rate measured by the frequency of out-of-charged events.

Figure 4 illustrates that, as a result of the increased reward value primarily driven by the take order action, max-policy sharing learners outperform independent learners in terms of the number of completed orders during the learning process. This outcome can be attributed to sharing learners exploring a larger area of the search space and exchanging their findings, which, in turn, enhances the overall service quality.

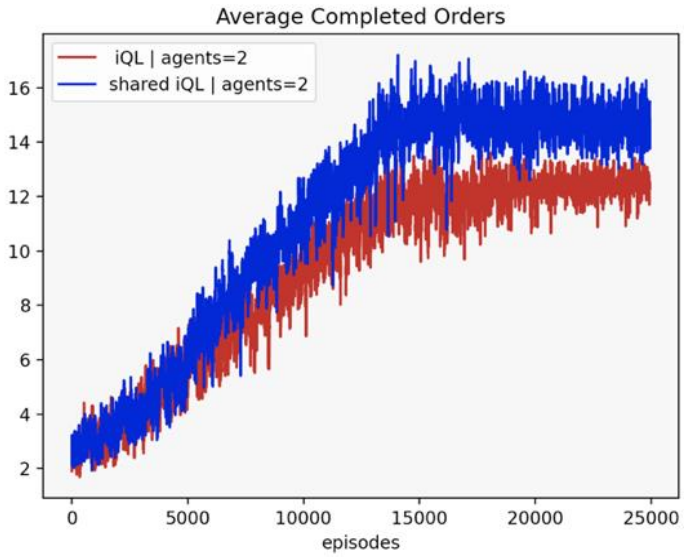


Figure 4 The average productivity, measured by the number of completed orders per agent, increases with no instances of ignored orders, for both max-policy sharing and independent learners.

Furthermore, in terms of empty miles, both types of learners demonstrate an improvement in reducing these miles throughout the learning process (Figure 5). This reduction is a direct consequence of decreasing the frequency of recharge and no operation actions, as agents learn to optimize their decisions and focus more on completing orders.

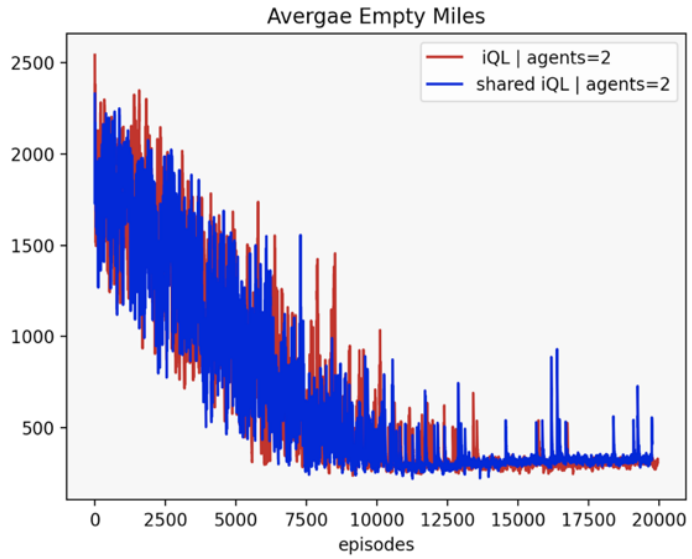


Figure 5 The average number of empty miles generated per agent decreases over the time, demonstrating improved agent balance and productivity, as well as zero instances of empty mile generation for both max-policy sharing and independent learners.

In Figure 6, a substantial decrease in the number of lost customers due to competition is observed for both categories of learners, indicating a noteworthy shift towards cooperative behavior among the agents. Moreover, the results shown in Figure 7 demonstrate a significant decline in the

occurrence of out-of-charge events, thereby underscoring a more harmonious team dynamic. This outcome serves to emphasize that the agents are more effectively distributed in terms of workload, with an equitable balance being achieved across the team, such that no individual agent is overburdened or underutilized. Such findings hold profound implications for the design and implementation of efficient and reliable autonomous mobility on demand (AMoD) systems, as they suggest that maximizing cooperation among agents and ensuring equitable workload distribution can lead to a considerable reduction in lost customers and out-of-charge events, resulting in improved Quality of Service (QoS) indicators.

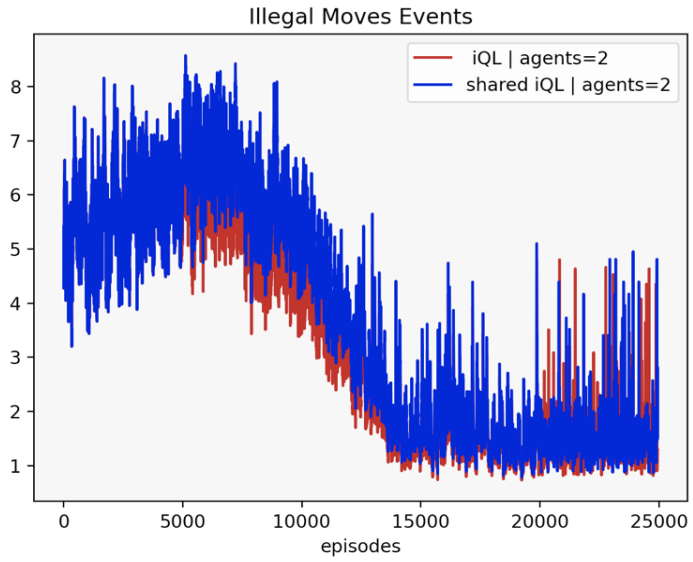


Figure 6 The average number of lost-order events per agent decreases over time, resulting in higher productivity and ignored lost customers.

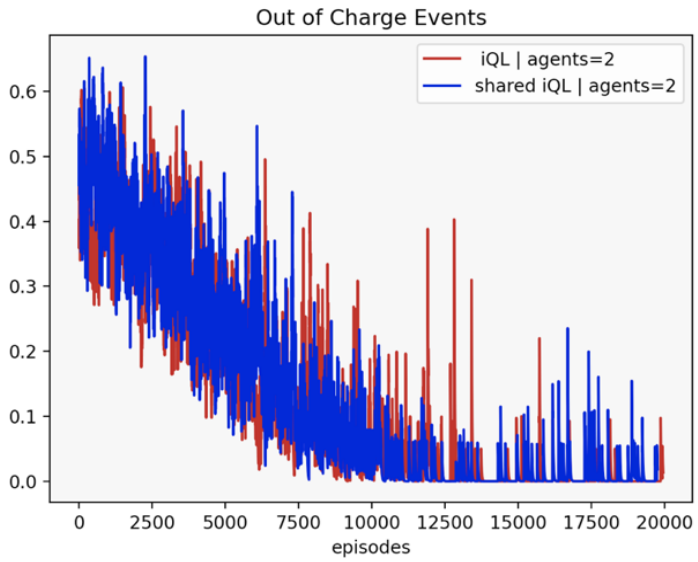


Figure 7 The average number of out-of-charge events per agent decreases over time, indicating improved agent balance and zero instances of agents running out of charge.

It is vital to acknowledge that the max-policy sharing approach's effectiveness relies on several factors such as the problem domain, environment, and sharing rate. Striking a balance between exploration and sharing during the learning process, as well as fine-tuning the frequency of policy sharing, is crucial to prevent convergence to suboptimal policies.

The scalability of the max-policy sharing approach also warrants discussion. Our study demonstrated its superiority over independent learning with a small number of agents; however, the performance of this approach with a larger number of agents and increased complexity remains uncertain. We utilized a simple communication and voting mechanism, where agents share their policies and choose the most popular action for each state. Nevertheless, more sophisticated communication and voting mechanisms might better harness the collective intelligence of the agents, resulting in improved performance.

## 5. CONCLUSION

In this study, we compared the effectiveness of max-policy sharing agents and independent learners in an Autonomous Mobility on Demand (AMoD) system using reinforcement learning. Our results showed that the max-policy sharing approach outperformed independent learning in several Quality of Service (QoS) indicators, including completed orders, empty miles, lost customers due to competition, and out-of-charge events.

However, this study raises a broader question about the benefits of Multi-Agent Systems (MAS) versus single agents, which requires further investigation. Specifically, we observed that independent learners do not benefit from an increased number of agents empirically, indicating that adding more agents to the learning process does not necessarily improve performance. In contrast, the max-policy sharing approach allows agents to learn faster by sharing policies and leveraging the most common action choices among their peers.

Potential future research directions include exploring weighted voting mechanisms that prioritize policies of higher performing agents or mechanisms that dynamically adjust the sharing rate based on agent performance. Moreover, examining the impact of the size and structure of the communication network on the max-policy sharing approach's performance could provide valuable insights.

## REFERENCES

[1] S. Gronauer and K. Diepold, “Multi-agent deep reinforcement learning: a survey,” Artificial Intelligence Review, Apr. 2021, doi: https://doi.org/10.1007/s10462-021-09996-w.

[2] V. Mnih et al., “Asynchronous Methods for Deep Reinforcement Learning,” arXiv.org, 2016. https://arxiv.org/abs/1602.01783.

[3] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, “Proximal Policy Optimization Algorithms,” arXiv.org, 2017. https://arxiv.org/abs/1707.06347

[4] L. Kraemer and B. Banerjee, “Multi-agent reinforcement learning as a rehearsal for decentralized planning,” Neurocomputing, vol. 190, pp. 82–94, May 2016, doi: https://doi.org/10.1016/j.neucom.2016.01.031.

[5] F. A. Oliehoek, M. T. J. Spaan, and N. Vlassis, “Optimal and Approximate Q-value Functions for Decentralized POMDPs,” Journal of Artificial Intelligence Research, vol. 32, pp. 289–353, May 2008, doi: https://doi.org/10.1613/jair.2447

[6] A. OroojlooyJadid and D. Hajinezhad, “A Review of Cooperative Multi-Agent Deep Reinforcement Learning,” arXiv:1908.03963 [cs, math, stat], Apr. 2021, Available: https://arxiv.org/abs/1908.03963

[7] S. Sukhbaatar, A. Szlam, and R. Fergus, “Learning Multiagent Communication with Backpropagation,” arXiv:1605.07736 [cs], Oct. 2016, Available: https://arxiv.org/abs/1605.07736

[8] P. Varshavskaya, L. Pack, and D. Rus, “Efficient Distributed Reinforcement Learning Through Agreement.” Accessed: May 07, 2023. [Online]. Available: https://people.csail.mit.edu/lpk/papers/dars08.pdf

[9] T. Enders, J. Harrison, M. Pavone, and M. Schiffer, “Hybrid Multi-agent Deep Reinforcement Learning for Autonomous Mobility on Demand Systems,” arXiv:2212.07313 [cs, eess], Dec. 2022, Accessed: May 07, 2023. [Online]. Available: https://arxiv.org/abs/2212.07313

[10] C. Wang and A. Turan, “Multi-Agent Renforcement Learning for Dynamic Pricing and Fleet Management in Autonomous Mobility-On-Demand Systems,” 2022. Accessed: May 07, 2023. [Online]. Available: https://repository.arizona.edu/bitstream/handle/10150/666922/ITC_2022_22-02-04.pdf?sequence=1

[11] C. Fluri, C. Ruch, J. Zilly, J. Hakenberg, and E. Frazzoli, “Learning to Operate a Fleet of Cars,” IEEE Xplore, Oct. 01, 2019. https://ieeexplore.ieee.org/abstract/document/8917533 (accessed May 07, 2023).

[12] Y. Lu, Y. Liang, Z. Ding, Q. Wu, T. Ding, and W.-J. Lee, “Deep Reinforcement Learning based Charging Pricing for Autonomous Mobility-on-Demand System,” IEEE Transactions on Smart Grid, pp. 1–1, 2021, doi: https://doi.org/10.1109/tsg.2021.3131804.

[13] S. He, Y. Wang, S. Han, S. Zou, and F. Miao, “A Robust and Constrained Multi-Agent Reinforcement Learning Framework for Electric Vehicle AMoD Systems,” 2022. Accessed: May 07, 2023. [Online]. Available: https://arxiv.org/pdf/2209.08230.pdf

## AUTHORS

**Ebtehal Alotaibi** is currently pursuing a PhD in the Institute of Perception, Action and Behavior at the University of Edinburgh, UK, where she is conducting research on developing innovative multi-robot reinforcement learning techniques for autonomous mobility on demand (AMoD). She holds first-class honors degrees in Computer Science at both the MSc and BSc levels from Imam Mohammad Ibn Saud Islamic University, SA. With research interests spanning autonomous vehicles, heuristics, optimization, and robotics, Ebtehal has published multiple research articles in her field. Moreover, she has received a US patent for a multi-UAV collaboration system designed for search and rescue missions.

**J. MICHAEL HERRMANN** received the Ph.D. degree from the University of Leipzig, in 1993. His Ph.D. focused on the mathematical aspects of artificial neural networks. He has been a Research Assistant, since 1992, and has held a postdoctoral position at Denmark and Japan. In 2008, after a temporary position as an Assistant Professor with the University of Goöttingen, he was appointed as a Lecturer of robotics with the School of Informatics, The University of Edinburgh. He is currently a Lecturer with the Institute of Perception, Action and Behaviour, The University of Edinburgh. His research interests are in self-organization, robot learning, neural avalanches, auditory systems, bio-medical data analysis, metaheuristic optimization, AMoD, and information theory.