20th IMEKO TC10 Conference
"Measurement for Diagnostics, Optimization and Control to Support Competitiveness and Innovation"
Lisbon, Portugal, September 28–29, 2026

Single-Agent Reinforcement Learning in Multi-Agent Production Environments: Effects of Action and Observation Overlap

Bence Pálvölgyi[1], Zsolt János Viharos[1,2]

[1]Research Laboratory on Engineering and Management Intelligence of the HUN-REN Institute for Computer Science and Control (SZTAKI), Center of Excellence of the Hungarian Academy of Sciences (MTA),
Budapest, 1111, Hungary,
palvolgyi.bence@sztaki.hu, viharos.zsolt@sztaki.hu
[2]Faculty of Economics and Business of the John von Neumann University,
Kecskemét, 6000, Hungary

Abstract **– Reinforcement Learning (RL) has demonstrated strong performance in production and manufacturing systems, where decision processes can be modeled as single-agent environments with well-defined state and action spaces. However, more complex real-world production processes inherently involve multiple interacting entities, making Multi-Agent Reinforcement Learning (MARL) the natural tool of optimization. Even if similar to RL, MARL introduces new challenges, including non-stationarity, scalability issues, and increased training complexity due to agent interactions.**

This paper investigates an alternative approach that employs a single-agent RL framework within a multi-agent environment. Instead of learning multiple coordinated policies, a single agent is trained to control multiple entities, with shared information. The study explores varying degrees of action and observation overlap between the controlled entities, analyzing how shared and independent components influence learning efficiency and policy quality. By comparing these configurations on the same test environment, the work provides insights into whether single-agent abstractions can effectively approximate multi-agent behavior. The results highlight the trade-offs between model simplicity and representational complexity, raising avareness for applying RL in complex production systems without the full overhead of MARL.

Keywords – IMEKO, TC10, Diagnostics, Optimisation, Control, Multi-Agent Reinforcement Learning, Industry, Innovation and Infrastructure.*

I. INTRODUCTION

Reinforcement Learning (RL), typically formulated as a Markov Decision Process, has been widely used for simulating and optimizing production and manufacturing systems, where sequential decision-making can be learned directly from interaction with a modeled environment. In such settings, single-agent RL provides a relatively straightforward framework, assuming a centralized controller operating over a unified state and action space.

However, more complex real-world production environments inherently consist of multiple interacting entities, such as machines, transport units, and processing stations, whose behaviors are interdependent. These systems are more naturally described within the framework of Multi-Agent Reinforcement Learning (MARL). Unlike single-agent RL, MARL problems, among others, require careful categorization along several dimensions, as stated in the paper written by Wong A. et al. [1]. Environments can be cooperative, collaborative, competitive, or mixed depending on the reward structure, while interaction patterns may follow agent-environment cycles (turn-based execution) or fully parallel action schemes, shown in Fig. 1. Additionally, agents themselves may be homogeneous, as sharing identical capabilities and policies, or heterogeneous, with distinct roles and action spaces.

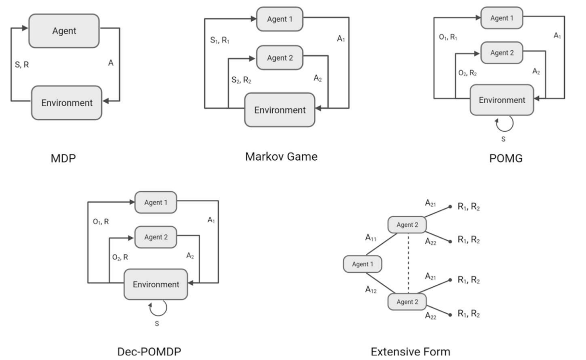

Fig. 1. Multi-Agent Environment types (Wong et al. [1])

From a learning perspective, MARL introduces further design choices regarding control and training structure. Agents may operate in fully decentralized settings, under centralized control, or within a hybrid paradigm such as Centralized Training with Decentralized Execution (CTDE). Techniques such as parameter sharing are often employed in homogeneous settings to improve scalability and sample efficiency, while heterogeneous systems typically require more specialized representations.

The use case investigated in this paper corresponds to a cooperative MARL environment with agent-environment cycle dynamics, designed to capture key properties of an industrial production system. Within this framework, both homogeneous and heterogeneous agent configurations are considered, forming the primary axis of comparison to the single-agent solution. By analyzing these settings, the study aims to better understand how structural differences in agent design influence learning behavior and the effectiveness of single-agent abstractions in approximating multi-agent coordination.

II. RELATED RESULTS IN THE LITERATURE

The increasing complexity of Multi-Agent Reinforcement Learning (MARL) has led to the development of structured frameworks aimed at better understanding and evaluating algorithm performance in dynamic environments. The work done by Hu, S. et al. [2] introduces adaptability as a central concept for assessing MARL systems under changing conditions. It highlights that real-world multi-agent systems are subject to variability in agent populations, task objectives, and execution environments, which limits the direct applicability of MARL methods developed for static benchmarks. The proposed framework distinguishes between learning adaptability, policy adaptability, and scenario-driven adaptability, emphasizing the importance of robustness beyond controlled experimental setups. This perspective is particularly relevant for production-like environments, where system conditions evolve continuously and require stable coordination strategies.

From an algorithmic standpoint, a survey paper by Liang J. et al. [3] provides a comprehensive categorization of MARL methods based on both learning paradigms and reward structures. Building on the transition from single-agent Markov Decision Process to multi-agent Markov games, the survey groups algorithms into value-based, policy-based, and actor-critic approaches, while further distinguishing between cooperative, competitive, and mixed settings. It also identifies core challenges inherent to MARL, including high dimensionality, non-stationarity, partial observability, and scalability. Representative methods such as Proximal Policy Optimization (PPO) and Multi-Agent Deep Deterministic Policy Gradient (MADDPG) are discussed in terms of their contributions and limitations, providing a structured overview of the design space relevant to multi-agent learning.

Complementing algorithmic analyses, Zepeng et al. [4] highlight the breadth of MARL applications across domains such as autonomous vehicles, energy systems, scheduling, and resource management. The survey emphasizes that, despite strong performance in simulated benchmarks, practical deployment remains constrained by challenges such as hyperparameter sensitivity and computational complexity. It also reviews benchmark environments and task formulations, demonstrating the adaptability of MARL to diverse real-world problems, while underlining the need for scalable and robust training methodologies.

Specific architectural approaches further refine the design of MARL systems. The concept of centralized training with decentralized execution (CTDE), as discussed by Amato et al. [5], allows agents to leverage global information during training while maintaining decentralized policies at execution time. This paradigm addresses coordination challenges without violating practical constraints of distributed systems. In homogeneous agent settings, parameter sharing has been shown to significantly improve scalability and sample efficiency. The work "Parameter Sharing Deep Deterministic Policy Gradient for Cooperative Multiagent Reinforcement Learning" demonstrates that sharing actor and/or critic networks enables faster learning and reduced memory requirements, particularly when agents are exchangeable and operate under shared reward structures.

The distinction between homogeneous and heterogeneous agents introduces further complexity. In homogeneous settings, studies such as seen in the paper done by Chu, X. et al. [6] show that cooperative behavior can emerge from shared policies with minimal communication, and that local perception and limited interaction can sometimes outperform fully informed strategies. In contrast, heterogeneous agent systems require more expressive models to capture differing roles and capabilities.

Kent et al. [7] address this by proposing algorithms with theoretical guarantees, such as HATRPO and HAPPO, which ensure stable learning and convergence in heterogeneous environments. These approaches demonstrate improved coordination and performance compared to standard MARL baselines, highlighting the importance of explicitly modeling agent diversity.

Overall, the literature indicates that while MARL provides a powerful framework for modeling complex multi-agent systems, its practical application is constrained by scalability, stability, and adaptability challenges. These limitations motivate alternative formulations, such as single-agent control of multi-entity systems with structured comparisons between homogeneous and heterogeneous agent configurations, as investigated in this work.

III. DESCRIPTION OF THE METHOD

The study investigates the effect of observation-space decomposition in a turn-based multi-agent reinforcement learning environment. A custom gridworld environment was implemented using the AEC (Agent Environment Cycle) model and evaluated with PPO agents. The environment consists of two agents acting sequentially in a shared discrete state space. Each agent controls independent movement actions, while the environment state contains the positions and task-related variables of both agents.

Two experimental formulations were investigated. In the first formulation, a decentralized multi-agent setup was used, where each agent received an individual observation vector. Different observation overlap configurations were tested, including fully disjoint and fully overlapping observation spaces. In the fully disjoint setting, each agent observed only its own subset of the global state. In the fully overlapping setting, both agents received the complete environment state.

In each formulation, the same environment dynamics were reformulated as a single-agent problem. Instead of multiple agents, one centralized agent controlled the actions of the currently active entity. The observation space consisted of the union of all agent observations, extended with an additional agent-selection parameter indicating which logical agent was currently acting. This formulation preserves the sequential decision process while removing decentralized policy execution.

The objective of the experiments was to compare the learning behavior and task performance of decentralized and centralized formulations under different information-sharing conditions. Training performance was evaluated using cumulative episode reward and task completion statistics over multiple training iterations.

Finally, the two methods were tested in a simplified simulated version of a real world production process, where two agents have partial overlap.

IV. RESULTS AND DISCUSSIONS

Preliminary results indicate that the relationship between observation overlap and policy structure significantly affects learning performance. In the fully overlapping observation setting, the centralized single-agent formulation achieved faster convergence and higher cumulative rewards compared to the decentralized multi-agent configuration. This behavior suggests that when all relevant information is globally accessible, centralized policy optimization can exploit the complete state representation more efficiently than independently acting agents.

In contrast, experiments using fully disjoint observation spaces produced different behavior. Under restricted observability, the decentralized multi-agent formulation outperformed the centralized single-agent representation in both training stability and final task performance. One possible explanation is that decentralized agents specialize more effectively within their limited observation domains, while the centralized formulation must simultaneously learn policy switching behavior and implicit agent-role separation using the shared policy representation.

The early results suggest that the degree of observation overlap may strongly influence whether centralized or decentralized reinforcement learning formulations are more suitable for a given task. While fully overlapping observations appear to favor centralized optimization, disjoint observation spaces may benefit from decentralized policy structures that preserve independent agent representations.

The presented results are preliminary, and further experiments are required to evaluate statistical significance, scalability to higher-dimensional environments, and the effect of partial observation overlap configurations. Future work will also investigate asymmetric action spaces, shared-resource constraints, and interaction-driven coupling mechanisms between agents.

V. CONCLUSIONS AND OUTLOOK

In this section the results of the paper are to be summarized and outlook for further research can be appointed shortly here.

VI. ACKNOWLEDGMENTS

The research was supported by the European Commission (EC) through the DiGreeS project under grant No. 101178079 and by AI EDIH Hungary project under EC grant No. 101120929.

Special thank has to be expressed for Jenő Csanaki, Krisztián Meskó, Zsolt Nagy at the Opel Szentgotthárd Kft. for their continuous collaboration.

On behalf of Project "Comprehensive testing of machine-learning algorithms" we thank for the usage of HUN-REN Cloud (https://science-cloud.hu/) that significantly helped us achieving the results published in

this paper.

REFERENCES

[1] **Wong, A., Bäck, T., Kononova, A.V. et al.**: Deep multiagent reinforcement learning: challenges and directions. Artif Intell Rev 56, 5023–5056, 2023. https://doi.org/10.1007/s10462-022-10299-x

[2] **Hu, S., Hady, M. A., Qiao, J., Cao, J., Pratama, M., & Kowalczyk, R.** Adaptability in Multi-Agent Reinforcement Learning: A Framework and Unified Review. (2025). arXiv preprint arXiv:2507.10142.

[3] **Liang J, Miao H, Li K, Tan J, Wang X, Luo R, Jiang Y.** A Review of Multi-Agent Reinforcement Learning Algorithms. *Electronics*. 2025; 14(4):820. https://doi.org/10.3390/electronics14040820

[4] **Zepeng, N., Lihua, X.,**: A survey on multi-agent reinforcement learning and its application, 2024, *Journal of Automation and Intelligence*, Volume 3, Issue 2.

[5] **Amato, C.:** An introduction to centralized training for decentralized execution in cooperative multi-agent reinforcement learning. 2024. arXiv preprint arXiv:2409.03052.

[6] **Chu, X., & Ye, H.**: Parameter sharing deep deterministic policy gradient for cooperative multi-agent reinforcement learning. 2017. arXiv preprint arXiv:1710.00336.

[7] **Kent T, Richards A, Johnson A.**: Homogeneous Agent Behaviours for the Multi-Agent Simultaneous Searching and Routing Problem. Drones. 2022; 6(2):51. https://doi.org/10.3390/drones6020051.