REPORT

Name: Gavit Deepesh Ravikant 20114

Teammates: Shirish Jha (2410705), Mohit Wadhwa (22205)

Capture The Flag Using Multi-Agent Reinforcement Learning Algorithms

Abstract

This report explores the application of Multi-Agent Reinforcement Learning (MARL) algorithms in a simulated Capture The Flag (CTF) environment. Specifically, it compares the performance of Independent Q-Learning (IQL) and Multi-Agent Proximal Policy Optimization (MAPPO) in handling coordination, strategy formation, and adaptability. The study reveals that MAPPO outperforms IQL in dynamic and cooperative settings due to its ability to foster inter-agent coordination, while IQL performs adequately in simpler, less dynamic environments.

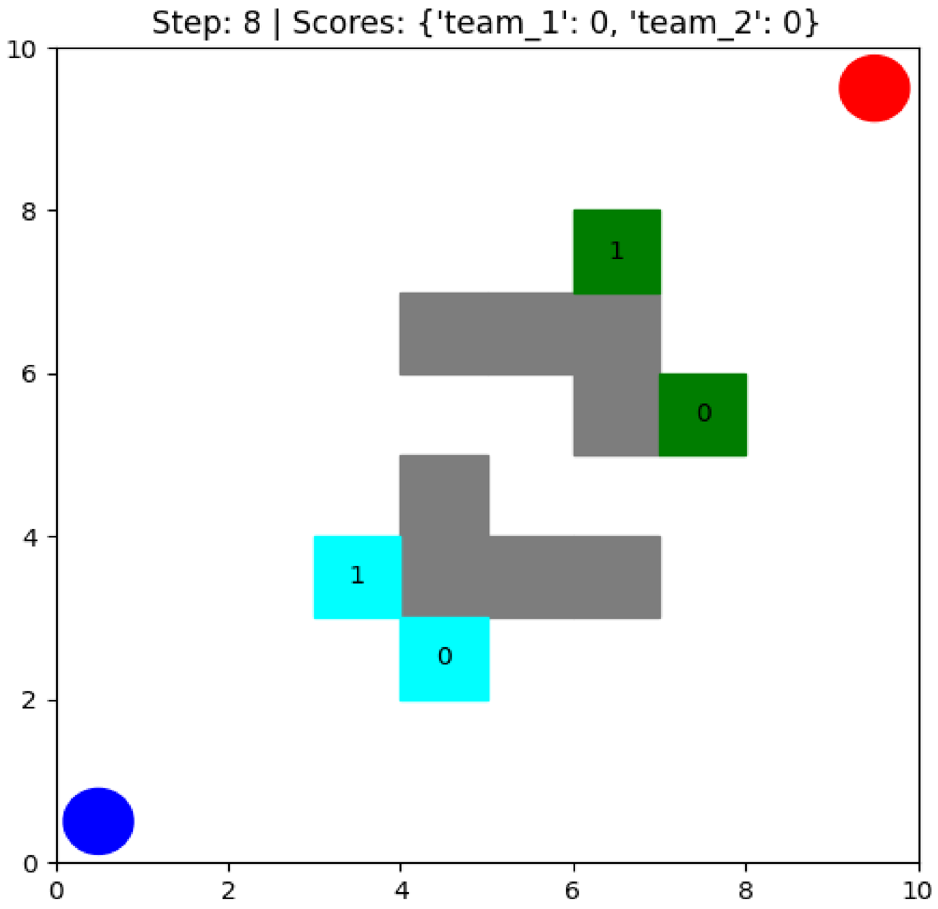

Figure 1.1: Environment for Capturing The Flag for Multiagents

1)Introduction

Background

Multi-Agent Reinforcement Learning (MARL) has emerged as a pivotal field in artificial intelligence, focusing on scenarios where multiple agents learn and interact within a shared environment. The complexity of MARL arises from the need for agents to not only learn optimal policies but also to adapt to the presence and actions of other agents. Capture The Flag (CTF) serves as an excellent testbed for MARL algorithms due to its inherent requirements for teamwork, strategy, and adaptability.

Motivation

The primary motivation of this study is to investigate how different MARL algorithms handle the challenges posed by a competitive and dynamic environment like CTF. By comparing Independent Q-Learning (IQL) and Multi-Agent Proximal Policy Optimization (MAPPO), we aim to understand the strengths and limitations of each algorithm in terms of coordination, strategy formation, and adaptability.

Problem Statement

The project focuses on simulating a CTF game to evaluate the capabilities of MARL algorithms. The objective is to assess how IQL and MAPPO perform in terms of learning effective strategies, coordinating among agents, and adapting to environmental changes in a competitive multi-agent setup.

Environment Description

Teams and Objectives

- **Teams**: The game consists of two opposing teams, each comprising two agents.
- **Objectives**:
    - **Offense**: Capture the opponent's flag and return it to the team's base.
    - **Defense**: Protect the team's own flag from being captured by opponents.

Defense Mechanism

A defense mechanism is activated when an opposing agent enters a visual range of 3 units around a team's flag. This mechanism simulates the detection and interception of intruders, adding a strategic layer to the game.

Static Elements

- **Obstacles**: Pre-placed within the environment and remain unchanged during simulations. They add complexity by restricting movement paths.
- **Flag Positions**: Fixed locations for each team's flag, serving as critical points for offense and defense.

Agent Movement

Agents can move freely within the environment and are allowed to occupy the same grid cell simultaneously. This design choice simplifies collision handling and focuses the study on strategic decision-making rather than movement constraints.

Core Challenges

- **Reward Structure Design**: Crafting a reward system that balances immediate exploration with strategic actions like offense and defense. Ensuring that delayed rewards (e.g., successful defense) do not discourage agents from exploring the environment.
- **Coordination Among Agents**: Enabling agents to develop strategies that require cooperation, such as coordinated attacks or defensive formations.

- **Adaptability to Environmental Changes**: Evaluating how algorithms adjust to modifications in the environment, such as altered obstacle placements.

---

2)Algorithms Used

Independent Q-Learning (IQL)

Description

IQL treats each agent as an independent learner. Each agent individually estimates Q-values and updates its policy based on personal experiences without considering the actions or policies of other agents.

Advantages

- **Simplicity**: Straightforward to implement and understand.
- **Scalability**: Works well in environments where agents operate independently.

Limitations

- **Non-Stationarity**: The environment appears non-stationary to each agent due to the actions of others, leading to convergence issues.
- **Lack of Coordination**: Agents cannot develop cooperative strategies, limiting effectiveness in tasks requiring teamwork.

Multi-Agent Proximal Policy Optimization (MAPPO)

Description

MAPPO extends Proximal Policy Optimization (PPO) to multi-agent settings using the Centralized Training with Decentralized Execution

(CTDE) paradigm. During training, agents share information to develop better strategies but execute actions independently during gameplay.

Advantages

- **Inter-Agent Coordination**: Facilitates the development of cooperative or adversarial strategies among agents.
- **Stable Training**: PPO's inherent stability benefits are extended to multi-agent scenarios.

Limitations

- **Computational Cost**: Centralized training requires more computational resources.
- **Dependency on Shared Information**: Effectiveness relies on the quality and relevance of the shared information during training.

Model Parameters

Policy Network:

Input Layer: 100 neurons (corresponding to the flattened observation space).

Hidden Layers: Two fully connected layers with 128 neurons each, activated by ReLU functions.

Output Layer: 5 neurons representing the action logits.

Optimizer: Adam optimizer with a learning rate of 0.0003.

Policy Loss: Clipped surrogate objective to ensure stability during training

$$L_{\text{policy}} = -E\,[\min\,(r_t\,(\theta)A,\ \text{clip}\,(r_t\,(\theta),\ 1-\epsilon,\ 1+\epsilon)\ A)]$$

where $r_t\,(\theta) = \frac{\pi_\theta\,(a|s)}{\pi_{\theta\text{old}}\,(a|s)}$ is the probability ratio.

3)Core Environment Components

Markov Decision Process (MDP)

The CTF environment is modeled as an MDP defined by a tuple (S,A,R,P,γ)(S, A, R, P, \gamma)(S,A,R,P,γ), where:

State Space (SSS)

- Represented as S⊂R10×10∖S′S \subset \mathbb{R}^{10 \times 10} \setminus S'S⊂R10×10∖S′, where S′S'S′ refers to cells occupied by obstacles or flags.
- **Agent Perception**:
  - **Position**: Agents are aware of their current location.
  - **Nearby Obstacles**: Information about obstacles in adjacent cells.
  - **Flags**: Relative positions of both the team's own flag and the opponent's flag.

Action Space (AAA)

- **Discrete Actions**:
  - **Up**
  - **Down**
  - **Left**
  - **Right**
  - **Stay**: Remain in the current position.

Reward Distribution (RRR)

Rewards are designed to incentivize strategic behaviors:

- **+100**: Capturing the opponent's flag and returning it to the base.
- **+25**: Successfully defending the team's flag by intercepting an opponent.

- **-25**: Being caught while intruding in the opponent's defensive area.
- **-2**: Remaining idle in the same position to discourage inactivity.

4)Evaluation Metrics

Performance Metrics

- **Win Rate**: Percentage of games where a team successfully captures the opponent's flag.
- **Average Scores**: Mean score achieved by each team, reflecting overall performance.
- **Draw Rate**: Frequency of games ending without a decisive winner.
- **Performance Stability**: Variance in scores across games, indicating reliability.

Training Metrics

- **Reward Trends**: Monitoring the average reward per episode to assess learning progress.
- **Loss Progression**: Tracking the convergence behavior of the learning algorithm during training.

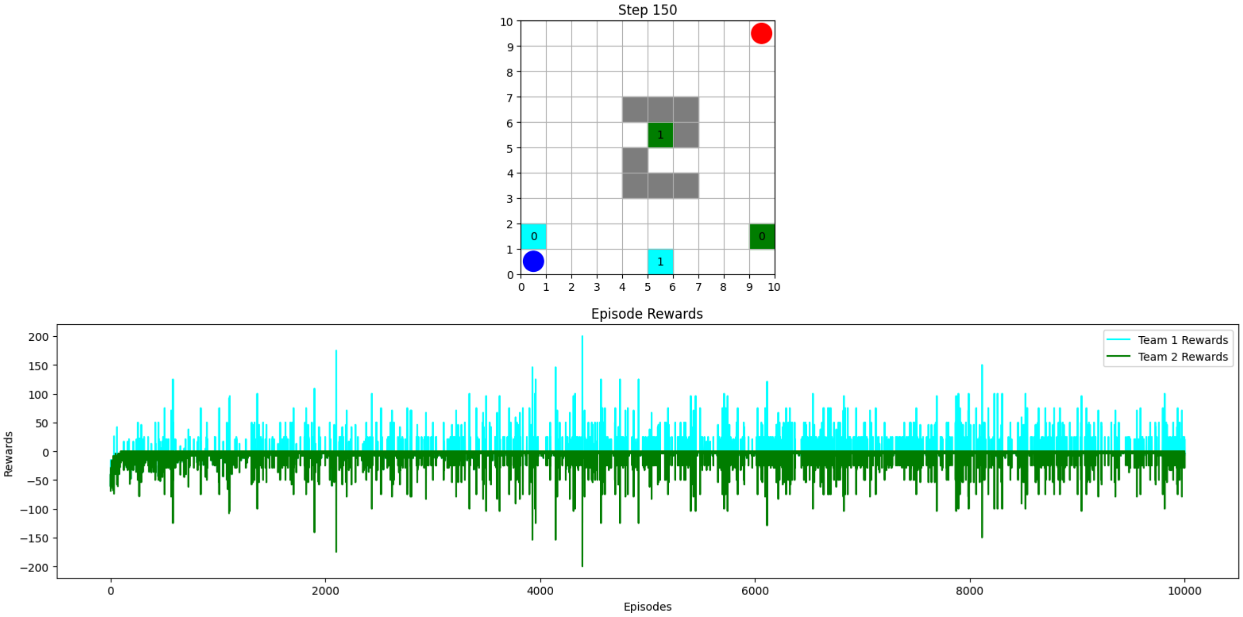

Figure 2: Loss progression during training (IQL)

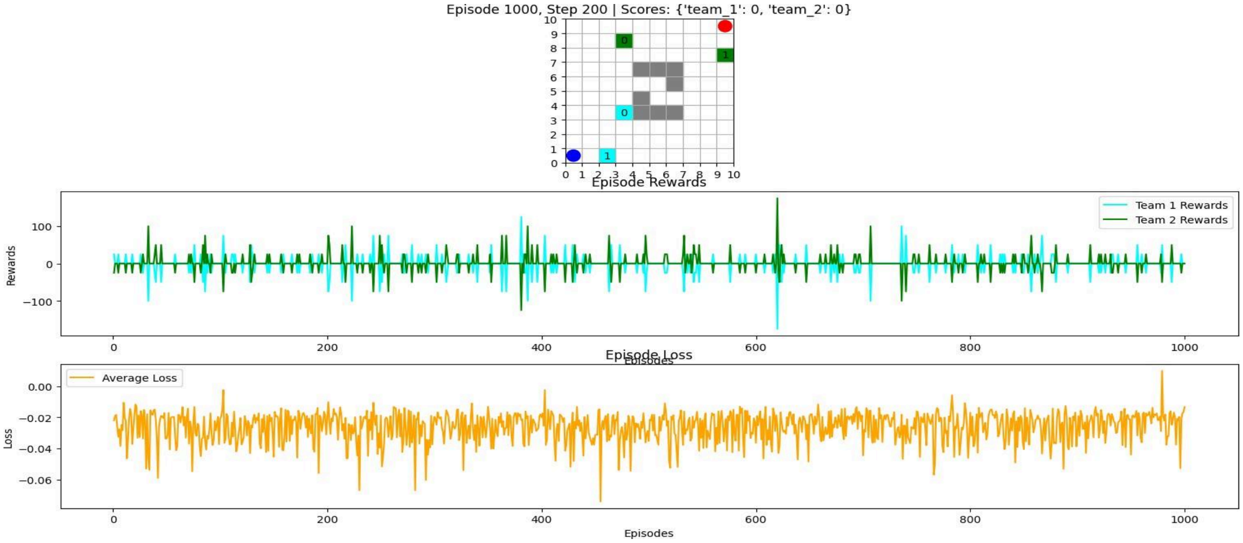

Figure 3: Loss & Reward progression during training (MAPPO)

5)Results and Observations

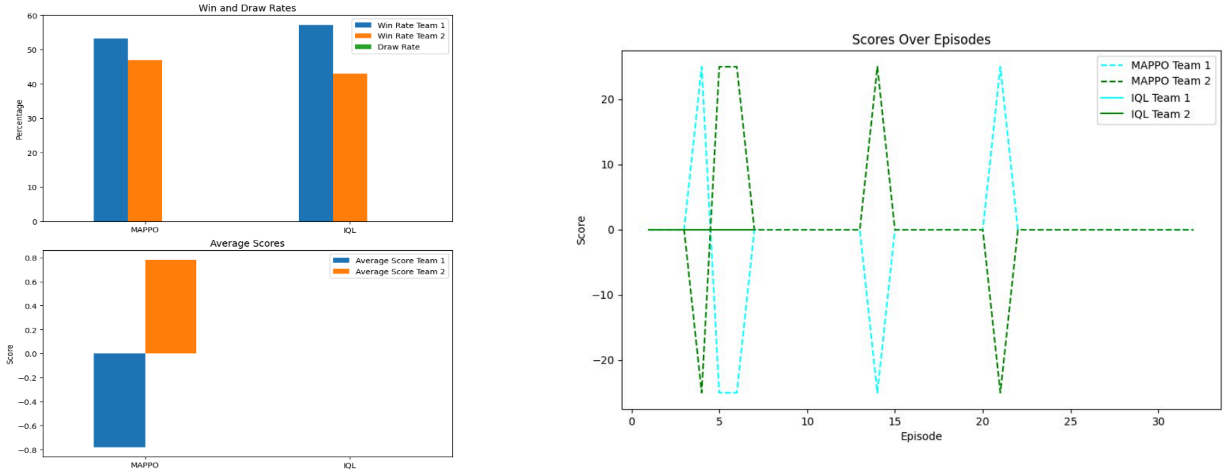

Figure 4: (Left) Win and Draw Rates; (Right) Scores Over Episodes.

Win and Draw Rates

MAPPO:

- **Team 1 Win Rate**: 53.13%
- **Team 2 Win Rate**: 46.88%
- **Draw Rate**: 0% (All games ended decisively)

IQL:

- **Team 1 Win Rate**: 57.14%
- **Team 2 Win Rate**: 42.86%
- **Draw Rate**: 0% (All games ended decisively)

Average Scores

MAPPO:

- **Team 1 Average Score**: -0.78125
- **Team 2 Average Score**: 0.78125

IQL:

- **Both Teams Average Score**: 0.0

Average Score Difference

MAPPO:

- Demonstrated dynamic interactions with an average score difference of -1.5625 between the teams.

IQL:

- Showed balanced gameplay, reflected in a score difference of 0.0.
-

Environment Comparisons

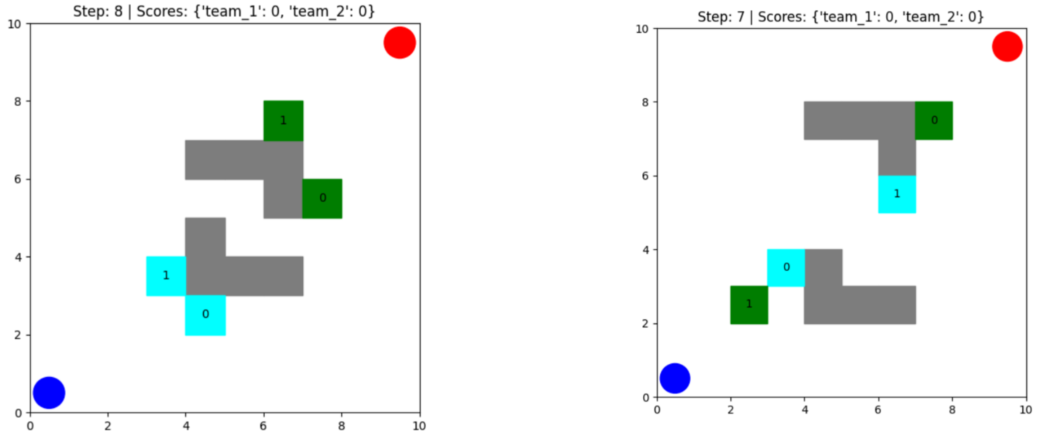
Step: 8 | Scores: {'team_1': 0, 'team_2': 0}
0
2
4
6
8
10
0
2
4
6
8
10
1
0
1
0
Step: 7 | Scores: {'team_1': 0, 'team_2': 0}
0
2
4
6
8
10
0
2
4
6
8
10
0
1
0
1

Figure 5: a) (Original Environment) b) (Changed Environment)

Original Environment

MAPPO:

- Agents displayed coordinated strategies in both offense and defense.
- Teams adapted dynamically to gameplay, resulting in varied and unpredictable outcomes.

IQL:

- Agents operated independently, leading to balanced but less dynamic gameplay.
- The lack of coordination resulted in more predictable strategies and outcomes.

Changed Environment (Altered Obstacles)

MAPPO:

- Maintained adaptability and balance despite environmental changes.
- **Win Rate**:
    - **Team 1**: 48.65%
    - **Team 2**: 51.35%
- **Average Scores**:
    - **Team 1**: +2.03
    - **Team 2**: -2.03

IQL:

- Independent strategies struggled with the altered environment.
- **Win Rate**:
    - **Team 1**: 53.33%
    - **Team 2**: 46.67%
- **Average Scores**:
    - **Team 1**: -5.00
    - **Team 2**: +5.00
- Significant disparity in scores indicated difficulty in adapting to changes.

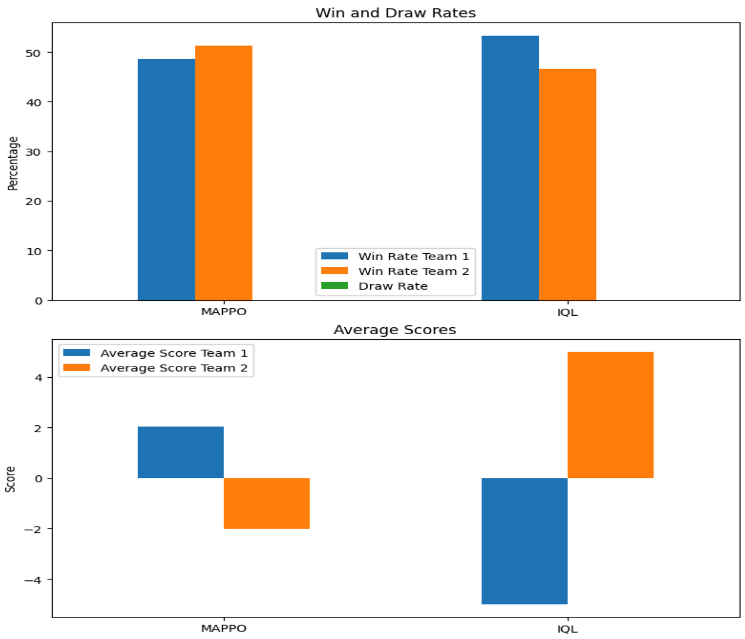

Figure 6: Win and Draw Rates

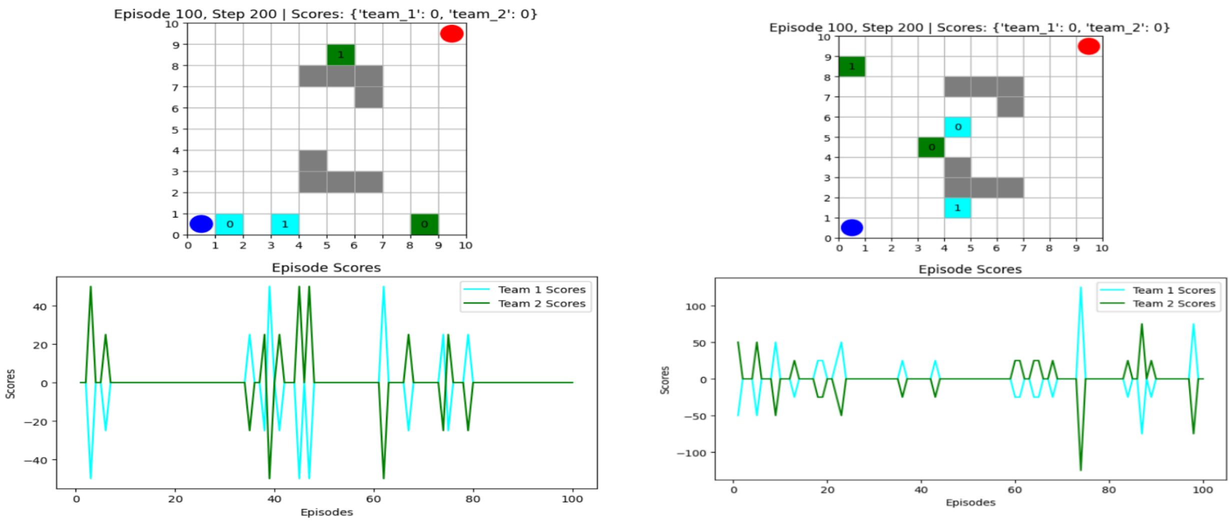

Figure 7: Testing Metrics i.e loss of IQL *(left)* & MAPPO *(right)*

6)Conclusions

Key Findings

- **MAPPO**:

    - Excels in environments requiring coordination and adaptability.
    - Agents developed cooperative strategies, enhancing team performance.
    - Performance remained stable across different environmental settings.
- **IQL**:
    - Effective in simpler environments with low coordination demands.
    - Struggled with adaptability when the environment changed.
    - Independent learning led to limitations in strategy development.

Recommendations

- **Use MAPPO**:
    - For tasks requiring robust teamwork and strategic coordination.
    - In dynamic environments where adaptability is crucial.
    - Despite higher computational costs, the benefits outweigh the expenses in complex tasks.
- **Use IQL**:
    - In simpler, static environments where agent independence is sufficient.
    - When computational resources are limited.
    - For preliminary studies before scaling up to more complex algorithms.

Final Thoughts

This study underscores the importance of selecting appropriate MARL algorithms based on the specific requirements of the environment and tasks at hand. MAPPO's ability to foster coordination and adaptability makes it suitable for complex, dynamic scenarios like CTF. In contrast, IQL's simplicity makes it suitable for less demanding environments. Future work could explore hybrid approaches or alternative algorithms to further enhance multi-agent coordination and performance.

References

1. **Sutton, R. S., & Barto, A. G.** (2018). *Reinforcement Learning: An Introduction*. MIT Press.
2. **Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O.** (2017). *Proximal Policy Optimization Algorithms*. arXiv preprint arXiv:1707.06347.
3. **Lowe, R., Wu, Y., Tamar, A., Harb, J., Abbeel, P., & Mordatch, I.** (2017). *Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments*. Advances in Neural Information Processing Systems.
4. **Busoniu, L., Babuska, R., & De Schutter, B.** (2008). *A Comprehensive Survey of Multi-Agent Reinforcement Learning*. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews).