Asymmetric Communication Policies in Multi-Agent Reinforcement Learning for Tethered Robots

Members:

Kelvin Mock, Kimia Ramezani, Chintan Shah

Carleton University

Agenda

- **Project Background**

1. Project Motivations
2. Project Setup

- **Algorithms**

1. Multi-Agent Proximal Policy Optimization (MAPPO)
2. Sparse Attention Mechanism
3. Graph Neural Networks (GNNs)

Agenda

- **Backup Plan**

1. Hierarchical Reinforcement Learning (HRL)
2. Self-Supervised Learning for Signal Interpretation
3. No-Communication Baseline as a Fall-Back

- **Performance Metrics**

Project Background

What and Why is this project?

Project Motivations

- Communication Challenges of MARL in real-world applications

Research	Real-World
Bidirectional Communication	One-way or Unreliable Transmissions

- Examples:

1. Warehouse & Supply Chain Robotics [6]
2. Military & Search-and-Rescue Operations
3. Autonomous Convoys [3]

- Asymmetric Communication Strategies

Project Setup

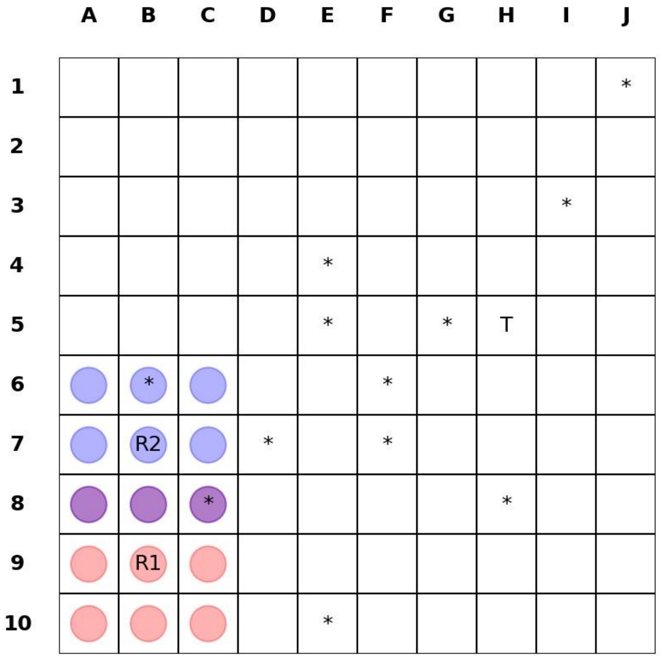

Figure 1: Grid-based structure of the environment.

Environment:

- 2D Grid-based
- * are obstacles
- T is the target for both bots to reach
- Randomize Positions of Objects
- Inform everyone others’ positions

Agents:

- R1: Leader – can speak not listen
- R2: Follower – can listen not speak

Action Space

- {↑, ↓, ⟵, ⟶, ↖, ↗, ↘, ↙, Stay}
- R1 is independent
- R2: signals from R1 + own perception
- Tether Constraint: Maintain a fixed distance possible.
- Simplicity: Initialize bots within the distance.

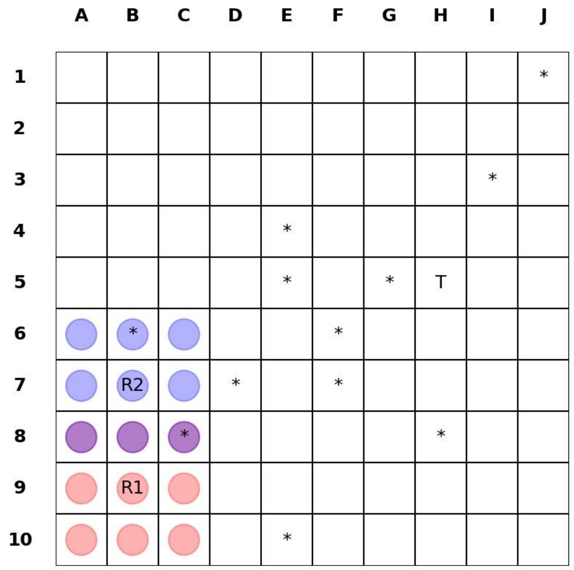

Figure 1: Grid-based structure of the environment.

Algorithms

How are we going to address the problem technically?

Carleton University

Related Sources

Multi-Agent Proximal Policy Optimization (MAPPO)

[5] Jungsoo Kim, Kyunghyun Cho, and David Sontag. “Communication-Efficient Multi-Agent Reinforcement Learning via Signaling”. In: Advances in Neural Information Processing Systems (NeurIPS 2021). 2021. https://proceedings.neurips.cc/paper/2021/hash/486c0401c56bf7ec2daa9eba58907da9-Abstract.html.

Sparse Attention Mechanism

[1] Abhijit Das, Sarthak Mittal, and Gaurav Sukhatme. “Tarmac: Targeted Multi-Agent Communication”. In: Proceedings of the 36th International Conference on Machine Learning (ICML 2019). 2019. https://proceedings.mlr.press/v97/das19a.html.

Related Sources

Graph Neural Networks (GNNs)

[9] Ryan Lowe, Jakub Sygnowski, Alexander I. Cowen-Rivers, Wendelin Böhmer, Jost Tobias Springenberg, Nicolas Heess, and Yuhuai Wu. “Multi-Agent Policy Optimization with Distributional Reinforcement Learning”. In: Advances in Neural Information Processing Systems (NeurIPS 2020). 2020.https://proceedings.neurips.cc/paper_files/paper/2020/hash/8b5c8441a8ff8e151b191c53c1842a38-Abstract.html.

Carleton University

Multi-Agent Proximal Policy Optimization (MAPPO) [5]

Purposes:

- Learn and Optimize cooperation.
- Decentralized + shared learning
- Partial Observability

Training Reward System:

- ✓ Successful Navigation
- ✓ Collision Avoidance
- ✓ Optimization: Minimized Steps
- ✓ Adhering to the Tether Constraint

Applying to Our Project:

- R1: to generate directional signals which maximized R2's efficiency
- R2: to learn when to trust R1 vs when to override

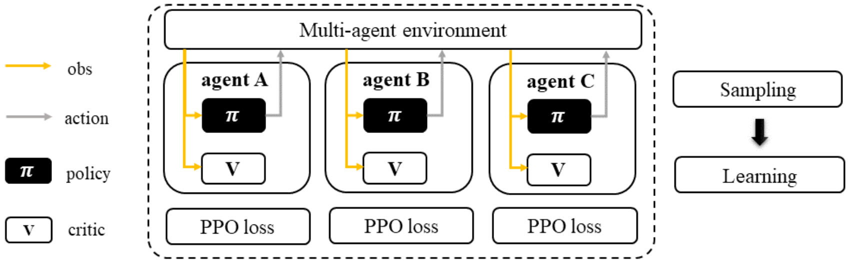

Figure 2. MAPPO Algorithm [7]

Carleton University

Sparse Attention Mechanism [1]

- For Efficient Signaling
- Challenges: Asymmetric Communication – ensuring one-way R1 → R2
- Solution: This helps R1 decide/regulate when to send important signals.

✓ R1 does not need to keep sending movement commands.

✓ R2 will not be overloaded → R2 can focus on critical instructions.

✓ Avoids Unnecessary (e.g., Redundant) Signals

Graph Neural Networks (GNNs) [9]

- Challenges: R2 might misinterpret signals from R1.
- R1’s signal will be **more complicated** than raw directional commands.
- Example:

Conveying “Move Right” with:

  - Obstacle Density Ahead?
  - How far is R2 away from an optimal trajectory?
  - Is the signal safe to follow? (Probably a Confidence Score)

✓ Signal is **encoded** from R1 using GNN with contextual information.

✓ R2 will learn to **decode** messages more intelligently.

✓ R2 will not blindly follow instructions.

Backup Plan

What else could be used if MAPPO fails/underperforms?

Related Sources

Hierarchical Reinforcement Learning (HRL)

[8] Ruyu Luo, Hui Tian,Wanli Ni, Julian Cheng, and Kwang-Cheng Chen. “Deep Reinforcement Learning Enables Joint Trajectory and Communication in Internet of Robotic Things”. In: IEEE Transactions on Wireless Communications 23.12 (Dec. 2024), pages 18154–18165. https://doi.org/10.1109/TWC.2024.3462450.

Carleton University

Hierarchical Reinforcement Learning (HRL)

- R1 as a **high-level planner** [8]: to provide route paths.
- R2 as a **low-level controller**: to make fine-grained movement decisions.
- Optimized performance where the environment is less complex.

Self-Supervised Learning

- For Signal Interpretation when R2 fails to interpret signals effectively
- Add a Prediction Model: with a **supervised loss function**.
- R2 can now learn to predict the **usefulness** of signals from R1
- Based on Past Experiences
- R2 can eventually adjust its **trust level** dynamically.

No Communication Baseline as a Fallback

- For the case when all RL-based solutions fail unpleasantly
- We compare the performance to a "**No Communication Baseline**".
- R2 navigates **independently**.

Alternative Purposes:

- To evaluate the significance of the leader-follower structure (beneficial?)
- To evaluate whether independent pathfinding is more optimal.

Performance Metrics

How do we make our model trustable?

Quality Assurance (QA) Evaluation

- **Completion Rate** – % of successful goal-reaching attempts
- **Navigation Efficiency** – Actual steps taken relative to the optimal path
- **Tether Constraint Violations** – exceeding max allowed distance
- **Collision Rate** – How often R2 collides with an obstacle
- Penalizing appropriately in our **reward function**
- Through **Comparisons**:

1. **No Communication Model** – one-way signals VS sole local sensing
2. **Fully Communicative Model** – whether bidirectional signal is better
3. **Asymmetric Model** (Ours) – evaluates learned policies against others

Thank You
Any Questions?

Carleton University

References

[1] Abhijit Das, Sarthak Mittal, and Gaurav Sukhatme. “Tarmac: Targeted Multi-Agent Communication”. In: Proceedings of the 36th International Conference on Machine Learning (ICML 2019). 2019. https://proceedings.mlr.press/v97/das19a.html.

[2] Chuangchuang Sun, Macheng Shen, and Jonathan P. How. “Scaling Up Multiagent Reinforcement Learning for Robotic Systems: Learn an Adaptive Sparse Communication Graph”. In: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Las Vegas, NV, USA, Oct. 2020, pages 11755–11762. https://doi.org/10.1109/IROS45743.2020.9341303.

[3] Federico Mason, Federico Chiariotti, Andrea Zanella, and Petar Popovski. “Multi-Agent Reinforcement Learning for Coordinating Communication and Control”. In: IEEE Transactions on Cognitive Communications and Networking 10.4 (Aug. 2024), pages 1566–1578. https://doi.org/10.1109/TCCN.2024.3384492.

Carleton University

References

[4] Gabriele Calzolari, Vidya Sumathy, Christoforos Kanellakis, and George Nikolakopoulos. “DMARL: A Dynamic Communication-Based Action Space Enhancement for Multi Agent Reinforcement Learning Exploration of Large Scale Unknown Environments”. In: 2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Abu Dhabi, UAE, Oct. 2024, pages 3470–3475. https://doi.org/10.1109/IROS58592.2024.10801319.

[5] Jungsoo Kim, Kyunghyun Cho, and David Sontag. “Communication-Efficient Multi-Agent Reinforcement Learning via Signaling”. In: Advances in Neural Information Processing Systems (NeurIPS 2021). 2021. https://proceedings.neurips.cc/paper/2021/hash/486c0401c56bf7ec2daa9eba58907da9-Abstract.html.

[6] Marc-André Blais and Moulay A. Akhloufi. “Reinforcement Learning for Swarm Robotics: An Overview of Applications, Algorithms, and Simulators”. In: Cognitive Robotics 3 (2023), pages 226–256. https://doi.org/10.1016/j.cogr.2023.07.004.

References

[7] MarLlib Documentation, "PPO Family," MarLlib: Multi-Agent Reinforcement Learning Library, 2025. [Online]. Available: https://marllib.readthedocs.io/en/latest/algorithm/ppo_family.html. [Accessed: 10-Feb-2025].

[8] Ruyu Luo, Hui Tian,Wanli Ni, Julian Cheng, and Kwang-Cheng Chen. “Deep Reinforcement Learning Enables Joint Trajectory and Communication in Internet of Robotic Things”. In: IEEE Transactions on Wireless Communications 23.12 (Dec. 2024), pages 18154–18165. https://doi.org/10.1109/TWC.2024.3462450.

[9] Ryan Lowe, Jakub Sygnowski, Alexander I. Cowen-Rivers, Wendelin Böhmer, Jost Tobias Springenberg, Nicolas Heess, and Yuhuai Wu. “Multi-Agent Policy Optimization with Distributional Reinforcement Learning”. In: Advances in Neural Information Processing Systems (NeurIPS 2020). 2020.https://proceedings.neurips.cc/paper_files/paper/2020/hash/8b5c8441a8ff8e151b191c53c1842a38-Abstract.html.

Carleton University

References

[10] Seongin Na, Hanlin Niu, Barry Lennox, and Farshad Arvin. "Bio-Inspired Collision Avoidance in Swarm Systems via Deep Reinforcement Learning". In: IEEE Transactions on Vehicular Technology 71.3 (Mar. 2022), pages 2511–2525. https://doi.org/10.1109/TVT.2022.3145346.

[11] Wei Qiu et al. "Off-Beat Multi-Agent Reinforcement Learning". In: Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023). IFAAMAS. London, United Kingdom, May 2023, pages 2424–2426.

Carleton University