Multi-Agent Reinforcement Learning: Overview

---

1. Multi-Agent Reinforcement Learning

Multi-Agent Reinforcement Learning involves multiple agents learning at the same time.

- More than one agent interacts with the environment
- Agents act in same time
- Actions influence other agents
- Agents may cooperate, compete, or both

Example:

Two robots must reach a shared goal at the same time.

Constraints:

- No communication
- Simultaneous actions
- Shared reward

---

2. Key Challenges in MARL

Learning in multi-agent systems introduces additional challenges.

- Each agent controls only its own actions
- The reward depends on joint behavior
- Coordination is required
- Timing is critical
- Non-stationarity environment

---

3. Non-Stationarity in Multi-Agent Learning

Non-stationarity is a fundamental challenge in MARL.

- In single-agent learning, the environment is stable
- In multi-agent learning, other agents are also learning
- Agent behavior changes over time
- The environment appears unstable from each agent's perspective

Consequences:

* Identical actions can lead to different outcomes
* Learning becomes unstable
* Convergence is more difficult

---

4. Independent Q-Learning

Independent Q-Learning is the simplest MARL approach.

- Each agent learns separately
- Each agent has its own Q-network
- Other agents are treated as part of the environment

Interpretation:

Each agent optimizes its own behavior without explicitly considering others.

Advantages:

* Simple implementation
* Useful as a baseline

Limitations:

* Strong non-stationarity
* Weak coordination
* Unstable learning

5. Parameter Sharing

Parameter Sharing extends Independent Q-Learning.

- All agents share the same Q-network
- Network parameters are identical
- Agents receive different observations

Advantages:

* Fewer parameters
* Faster training

Limitations:

* Agents cannot easily learn different roles
* Coordination remains limited
* Non-stationarity is not fully addressed

---

6. QMIX

QMIX is designed for cooperative multi-agent tasks.

- Each agent has its own Q-network
- A mixing network combines agent Q-values during training
- Global state information is used in the mixing process

Key principle:

Improving an individual agent's Q-value should not reduce the team's total value.

Interpretation:

Agents can act independently while still supporting team performance.

Benefits:

- Explicit coordination
- Improved stability
- Reduced impact of non-stationarity

---

7. MAPPO (Multi-Agent Proximal Policy Optimization)

MAPPO is a policy-based multi-agent method.

- Each agent learns a policy directly
- A centralized critic evaluates joint actions during training
- The critic is removed during execution

Interpretation:

Agents receive structured feedback about team behavior during training.

Benefits:

* High stability
* Strong coordination
* Effective in complex environments

MAPPO is widely considered a state-of-the-art MARL algorithm.

---

8. MADDPG + Communication (Multi-Agent Deep Deterministic Policy Gradient)

MADDPG is an off-policy actor–critic multi-agent algorithm.

- Each agent learns a deterministic policy for continuous actions
- A centralized critic observes joint states and actions during training
- Communication mechanisms allow agents to share learned messages

Interpretation:

Agents use centralized training and explicit communication to handle partial observability and coordinate their behaviors in continuous control tasks.

Benefits:

* Designed for continuous action spaces
* Can handle cooperative, competitive, or mixed settings
* Enables information sharing between agents

MADDPG + Communication is effective for complex continuous-control problems, but training becomes less stable as the number of agents increases.

9. Transformer-based MARL

Transformer-based MARL uses attention mechanisms to model interactions between agents.

- Policies or critics are built using transformer architectures
- Agents dynamically attend to relevant teammates
- Centralized training with decentralized execution

Interpretation:

Agents learn which teammates are most relevant for decision making at each time step.

Benefits:

* Scales well to many agents
* Captures complex coordination patterns
* Performs well in highly complex environments

Transformer-based MARL is a research-oriented approach commonly used for large-scale and highly dynamic multi-agent systems.

---

10. Key Insights

- Coordination and non-stationarity are fundamental challenges in MARL
- Independent learning approaches are simple to implement but often suffer from instability
- Centralized training helps solve non-stationarity and improve learning stability
- Decentralized execution is critical for scalability and real-world deployment

---

11.Concluding Summary

Multi-Agent Reinforcement Learning investigates how multiple agents can learn coordinated behaviors within shared environments. Modern approaches such as QMIX and MAPPO reduce the challenges of coordination and non-stationarity by using centralized training while preserving decentralized execution. This paradigm enables more stable training and improved performance in complex multi-agent systems.