

PrELIN: Provably Efficient Local-Information Networked Multi-Agent Reinforcement Learning

Ziyue Chu* Sharu Theresa Jose Leonardo Stella

School of Computer Science
University of Birmingham

CDC 2025 December 12

- ① Background & Motivation
- ② Problem Formulation & Algorithm
- ③ Theoretical Results
- ④ Results Evaluation & Proof
- ⑤ Experimental Validation
- ⑥ Conclusions & Future Work

Background: Learning in Multi-Agent Systems

Learning Paradigms:

- **Centralized learning¹:**
 - ✗ Exponential with agent #
 - ✗ Poor sample efficiency
- **Independent learning²:**
 - ✗ No inter-agent collaboration
 - ✗ Limited global information
 - ✗ Fails to reach global optimum

¹Lu, Chengxuan, et al. "Centralized RL for MA-cooperative environments." Evolutionary Intelligence (2024).

²Tan, Ming. "MARL: Independent vs. cooperative agents." ICML. 1993.

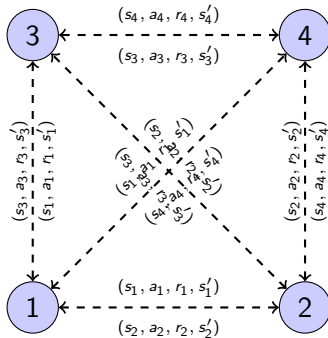
³Zhang, Kaiqing, et al. "Fully decentralized MARL with networked agents." ICML. PMLR, 2018.

Background: Learning in Multi-Agent Systems

Learning Paradigms:

- **Centralized learning¹:**
 - ✗ Exponential with agent #
 - ✗ Poor sample efficiency
- **Independent learning²:**
 - ✗ No inter-agent collaboration
 - ✗ Limited global information
 - ✗ Fails to reach global optimum
- **Networked decentralized learning³:**
 - ✓ **Balance** efficiency & collaboration
 - ✓ Scalable with communication

Communication Network



Agents exchange experience through local communication

¹Lu, Chengxuan, et al. "Centralized RL for MA-cooperative environments." Evolutionary Intelligence (2024).

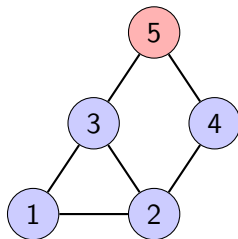
²Tan, Ming. "MARL: Independent vs. cooperative agents." ICML. 1993.

³Zhang, Kaiqing, et al. "Fully decentralized MARL with networked agents." ICML. PMLR, 2018.

Motivation: The Challenge of Networked MARL

Key Challenges:

- Q-learning with UCB is provably efficient only for single agent⁴
- Existing networked MARL⁵ requires **global network knowledge**
- Real-world constraints: dynamic, unknown topologies



Agents only know their neighbors
due to physical constraints

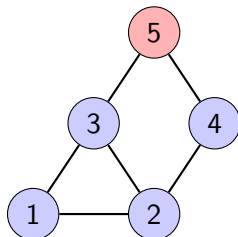
⁴ Jin, Chi, et al. "Is Q-learning provably efficient?." Advances in neural information processing systems 31 (2018).

⁵ Lidard, Justin, et al. "Provably efficient MARL with fully decentralized communication." 2022 American Control Conference (ACC).

Motivation: The Challenge of Networked MARL

Key Challenges:

- Q-learning with UCB is provably efficient only for single agent⁴
- Existing networked MARL⁵ requires **global network knowledge**
- Real-world constraints: dynamic, unknown topologies



Agents only know their neighbors
due to physical constraints

Research Question:

*How can we design a **provably efficient** **online** dec-MARL algorithm, given that agents **only** have access to **local** network information?*

⁴ Jin, Chi, et al. "Is Q-learning provably efficient?." Advances in neural information processing systems 31 (2018).

⁵ Lidard, Justin, et al. "Provably efficient MARL with fully decentralized communication." 2022 American Control Conference (ACC). 

Problem Formulation: Networked MA-MDP

Multi-Agent MDP: $\mathcal{M} = ([M], \mathcal{S}, \{A_i\}_{i=1}^M, P, \{r_h\}_{h=1}^H)$

Problem Formulation: Networked MA-MDP

Multi-Agent MDP: $\mathcal{M} = ([M], \mathcal{S}, \{A_i\}_{i=1}^M, P, \{r_h\}_{h=1}^H)$

Parallel MDP Decomposition:

- Strong homogeneous agents: $\mathcal{S}_i = \mathcal{S}_j$, $A_i = A_j$, $R_i = R_j$ for all $i, j \in [M]$

Problem Formulation: Networked MA-MDP

Multi-Agent MDP: $\mathcal{M} = ([M], \mathcal{S}, \{A_i\}_{i=1}^M, P, \{r_h\}_{h=1}^H)$

Parallel MDP Decomposition:

- Strong homogeneous agents: $\mathcal{S}_i = \mathcal{S}_j$, $A_i = A_j$, $R_i = R_j$ for all $i, j \in [M]$
- System transition factorizes: $P(s_h, a_h) = \prod_{i=1}^M P_h(s_{h,i}, a_{h,i})$
- Reduces exponential joint space to linear in M : $\pi^* = (\pi_1^*, \dots, \pi_M^*)$

Problem Formulation: Networked MA-MDP

Multi-Agent MDP: $\mathcal{M} = ([M], \mathcal{S}, \{A_i\}_{i=1}^M, P, \{r_h\}_{h=1}^H)$

Parallel MDP Decomposition:

- Strong homogeneous agents: $\mathcal{S}_i = \mathcal{S}_j$, $A_i = A_j$, $R_i = R_j$ for all $i, j \in [M]$
- System transition factorizes: $P(s_h, a_h) = \prod_{i=1}^M P_h(s_{h,i}, a_{h,i})$
- Reduces exponential joint space to linear in M : $\pi^* = (\pi_1^*, \dots, \pi_M^*)$
- Question: Can we go further?

Problem Formulation: Networked MA-MDP

Multi-Agent MDP: $\mathcal{M} = ([M], \mathcal{S}, \{A_i\}_{i=1}^M, P, \{r_h\}_{h=1}^H)$

Parallel MDP Decomposition:

- Strong homogeneous agents: $\mathcal{S}_i = \mathcal{S}_j$, $A_i = A_j$, $R_i = R_j$ for all $i, j \in [M]$
- System transition factorizes: $P(s_h, a_h) = \prod_{i=1}^M P_h(s_{h,i}, a_{h,i})$
- Reduces exponential joint space to linear in M : $\pi^* = (\pi_1^*, \dots, \pi_M^*)$
- Question: Can we go further?

Communication Network:

- Undirected graph: $\mathcal{G} = ([M], E)$ with Neighbor set: $\mathcal{N}(i)$
- **Key Assumption:** Agents only observe $\mathcal{N}(i)$, not global $\mathcal{G}$

Problem Formulation: Networked MA-MDP

Multi-Agent MDP: $\mathcal{M} = ([M], \mathcal{S}, \{A_i\}_{i=1}^M, P, \{r_h\}_{h=1}^H)$

Parallel MDP Decomposition:

- Strong homogeneous agents: $\mathcal{S}_i = \mathcal{S}_j$, $A_i = A_j$, $R_i = R_j$ for all $i, j \in [M]$
- System transition factorizes: $P(s_h, a_h) = \prod_{i=1}^M P_h(s_{h,i}, a_{h,i})$
- Reduces exponential joint space to linear in M : $\pi^* = (\pi_1^*, \dots, \pi_M^*)$
- Question: Can we go further?

Communication Network:

- Undirected graph: $\mathcal{G} = ([M], E)$ with Neighbor set: $\mathcal{N}(i)$
- **Key Assumption:** Agents only observe $\mathcal{N}(i)$, not global $\mathcal{G}$

Objective: Minimize system regret with learned policy $\pi^{k,m}$:

$$\rho = \sum_{m=1}^M \sum_{k=1}^K \left[V_{1,m}^*(s_{1,m}^k) - V_{1,m}^{\pi^{k,m}}(s_{1,m}^k) \right]$$

1. Online Value Iteration with Exploration:

$$Q_{h,m}^{k-1}(s, a) \leftarrow (1-\alpha_{t,m})Q_{h,m}^{k-1}(s, a) + \alpha_{t,m}[r + V_{h+1,m}^{k-1}(s') + \textcolor{red}{b}_{t,m}]$$

2. **Local** Information Exchange:

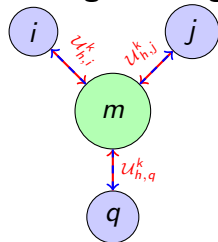
- Agent m broadcasts data to $\mathcal{N}(m)$
- Receives local data $\textcolor{red}{U}_{h,i}^k$ from neighbors
- Augmented dataset: $\mathcal{V} = \bigcup_{i \in \mathcal{N}(m)} \mathcal{U}_{h,i}^k$

3. UCB Exploration Bonus:

$$b_{t,m} = \frac{c}{\sqrt{\textcolor{red}{N}(m)}} \sqrt{\frac{H^3 \ell}{t}}$$

- Scales with local connectivity $\mathcal{N}(m)$
- Encourages exploration of uncertain states

Message Passing



Agent m exchanges
experience with neighbors

Main Results: Sample Complexity Bounds

Theorem 1: Sublinear System Regret Upper Bound

With probability at least $1 - p$, the system total regret is:

$$\rho = \mathcal{O} \left(\sqrt{f(M)H^4 SAT_\ell} \right).$$

Theorem 2: Sample Complexity with Polynomial Upper Bound

With UCB bonus $b_{t,m} = \frac{c}{\sqrt{\mathcal{N}(m)}} \sqrt{\frac{H^3 \ell}{t}}$, the episodic sample complexity is bounded by:

$$K(\epsilon, p) = \mathcal{O} \left(\frac{f(M)H^5 SA_\ell}{\epsilon^2 p^2} \right),$$

where $f(M) = \left(\sum_{m=1}^M \sqrt{1/\mathcal{N}(m)} \right)^2$ with $f(M) \in [M, M^2]$

Results Evaluation & Comparison

Algorithm	Regret Bound	Global Info?	Network Type
Jin et al. ⁴	$\tilde{O}(\sqrt{H^3 SAT})$	single-agent	
Lidard et al. ⁵	$\tilde{O}(\sqrt{MH^4 SAT})$	Yes	Fully-Connected
PrELIN (Ours)	$\tilde{O}(\sqrt{f(M)H^4 SAT})$	No	Any

Results Evaluation & Comparison

Algorithm	Regret Bound	Global Info?	Network Type
Jin et al. ⁴	$\tilde{O}(\sqrt{H^3 SAT})$	single-agent	
Lidard et al. ⁵	$\tilde{O}(\sqrt{MH^4 SAT})$	Yes	Fully-Connected
PrELIN (Ours)	$\tilde{O}(\sqrt{f(M)H^4 SAT})$	No	Any

Key Advantages of PrELIN:

- ✓ **Fully decentralized:** No global network knowledge required
- ✓ **Versatile & Practical:** Works with dynamic networks in real applications
- ✓ **Provably efficient:** Theoretical guarantees with sublinear regret

Network Topology Impact: $f(M)$ Analysis

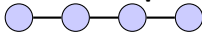
Network Type	$f(M)$	Sample Complexity
No connections (IDQ) ²	M^2	$\mathcal{O}(M^2 H^5 SA_\ell / \epsilon^2 p^2)$
Fully connected ⁵	M	$\mathcal{O}(M H^5 SA_\ell / \epsilon^2 p^2)$
Line graph	$M^2/2$	$\mathcal{O}(M^2 H^5 SA_\ell / \epsilon^2 p^2)$

No Connection



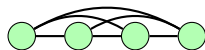
Worst case

Line Graph



No improvement

Fully Connected



Best case: $M \times$ speedup

Network Topology Impact: $f(M)$ Analysis

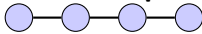
Network Type	$f(M)$	Sample Complexity
No connections (IDQ) ²	M^2	$\mathcal{O}(M^2 H^5 SA_\ell / \epsilon^2 p^2)$
Fully connected ⁵	M	$\mathcal{O}(M H^5 SA_\ell / \epsilon^2 p^2)$
Line graph	$M^2/2$	$\mathcal{O}(M^2 H^5 SA_\ell / \epsilon^2 p^2)$

No Connection



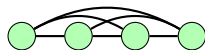
Worst case

Line Graph



No improvement

Fully Connected



Best case: $M \times$ speedup

Network Design Takeaway: Ensure $\sum_{m=1}^M \sqrt{1/\mathcal{N}(m)}$ grows sublinearly in M

Proof Sketch: From Regret to Sample Complexity

Step 1: Bound per-agent per-episode error recursively:

$$\delta_{h,m}^k \leq -\phi_{h+1,m}^k + \delta_{h+1,m}^k + \xi_{h+1,m}^k + \beta_{t,m} + \text{lower order terms}$$

Step 2: Key Lemma - UCB bonus and empirical transition error:

$$\sum_{i=1}^t \alpha_{t,m}^i \left[(\hat{P}^{k_i} - P_{h,m}) V_{h+1,m}^*(s, a) + b_{i,m} \right] \leq \mathcal{O} \left(\sqrt{\frac{H^3 \iota / t}{\mathcal{N}(m)}} \right) \triangleq \beta_{t,m}$$

Step 3: Aggregate over episodes and agents:

$$\sum_{m=1}^M \sum_{k=1}^K \delta_{1,m}^k \leq \sum_{m=1}^M \mathcal{O} \left(\sqrt{\frac{H^5 K \iota S A}{\mathcal{N}(m)}} \right) = \mathcal{O} \left(\sqrt{f(M) H^4 S A T \iota} \right)$$

Step 4: Apply Lemma 1 (Markov inequality) to convert regret $\rightarrow$ sample complexity

Experimental Setup: Cooperative Navigation

MPE Setup:

- 8 agents navigating to beacons
- 3 Different network topologies
- States: $(x, y) \in [0, 9]^2$ 2d-pos
- Action Set $\mathcal{A}_i = \{\uparrow, \downarrow, \leftarrow, \rightarrow, -\}$
- Step Reward: $r_{t,m}(s, a) = |s_{t+1,m} - x_{\text{tar}}| - |s_{t,m} - x_{\text{tar}}|$, $r \in [-1, 1]$

Training Performance:

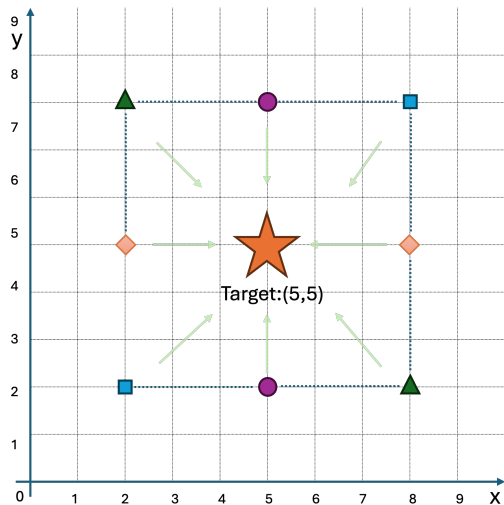
⇒ Optimal Episodic Rewards

$$V_{\max} = 3(\sqrt{2} + 1) * 4 \approx 28.97$$

Evaluations at Test Time:

⇒ Adversarially swap agents initial states;
Deploy the learned policy and Record.

Task Demo



Experimental Results: Cooperative Navigation

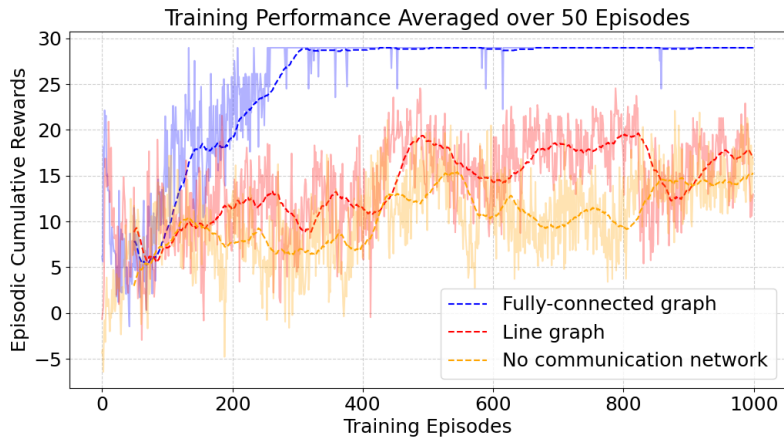


Table: Validation Results over 8 Episodes

Networks	Fully Connected	Line	Fully Disconnected
Mean $\pm$ std	28.97 ± 0.00	6.42 ± 2.26	6.11 ± 2.26

Summary

We developed **PrELIN**, a provably efficient networked MARL algorithm that:

- Requires only **local information** - no global network knowledge
- Achieves **sublinear regret**: $\tilde{O}(\sqrt{f(M)H^4SAT})$
- **Reduced sample complexity**: $\tilde{O}(f(M)H^5SA/\epsilon^2)$
- Adapts to **irregular and dynamic** network topologies

Summary

We developed **PrELIN**, a provably efficient networked MARL algorithm that:

- Requires only **local information** - no global network knowledge
- Achieves **sublinear regret**: $\tilde{O}(\sqrt{f(M)H^4SA})$
- **Reduced sample complexity**: $\tilde{O}(f(M)H^5SA/\epsilon^2)$
- Adapts to **irregular and dynamic** network topologies

Future Research:

- Time-varying & Complex network Analysis (e.g. RGG, Watts-Strogatz)
- Relaxing agents' identical learning goal assumption;
- Extension to continuous space, e.g. NN policy with TS exploration;

Thank you! Questions?

Presenter

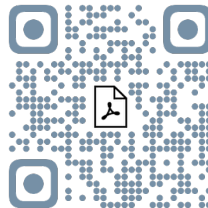
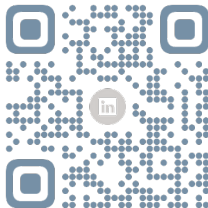
Ziyue Chu*

MARL Researcher from DLC Research Group of UoBirmingham

zero-cooper.github.io

Research Directions:

- Networked-MARL Thoery
- Learning Cooperative Games
- UAV Swarms Navigation



zxc332@student.bham.ac.uk



[Zero-Cooper_work](https://github.com/Zero-Cooper/Zero-Cooper_work)