Graph Neural Networks for Multi-Agent Cooperation

Michael Elrod
Clemson University School of Computing
Co-authors: Niloufar Mehrabi, Rahul Amin, Manveen Kaur, Long Cheng, Jim Martin, Abolfazl Razi
3/3/2026

Problem Statement

Drone fleet applications are growing:
- Disaster Response
- Environmental Monitoring
- Surveillance

Mission planning challenges for cooperative drone fleets:
- Partial observability
- Energy Constraints (Communication)
- Uncertain environments

Solution
- Graph Neural Network (GNN)
- Attention Transformer
- Deep Reinforcement Learning

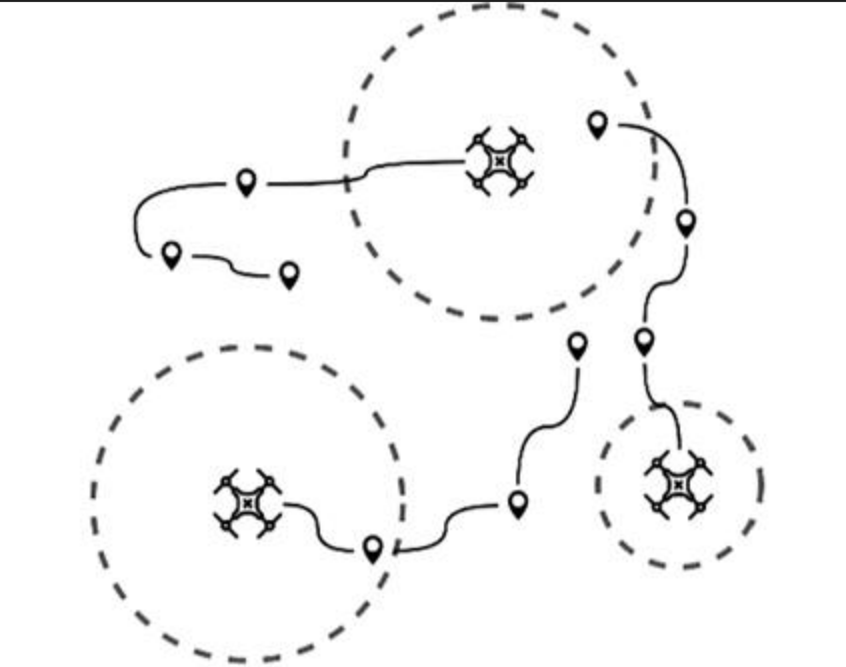
Limitations of Existing Approaches

Traditional methods:

- TSP solvers → struggle with dynamic environments
- Numerical optimization → requires prior information
- Greedy algorithms → suboptimal coordination
- DRL alone → poor multi-agent scaling and coordination

Proposed Method:

- Handle partial observability
- Enable information sharing
- Scalable



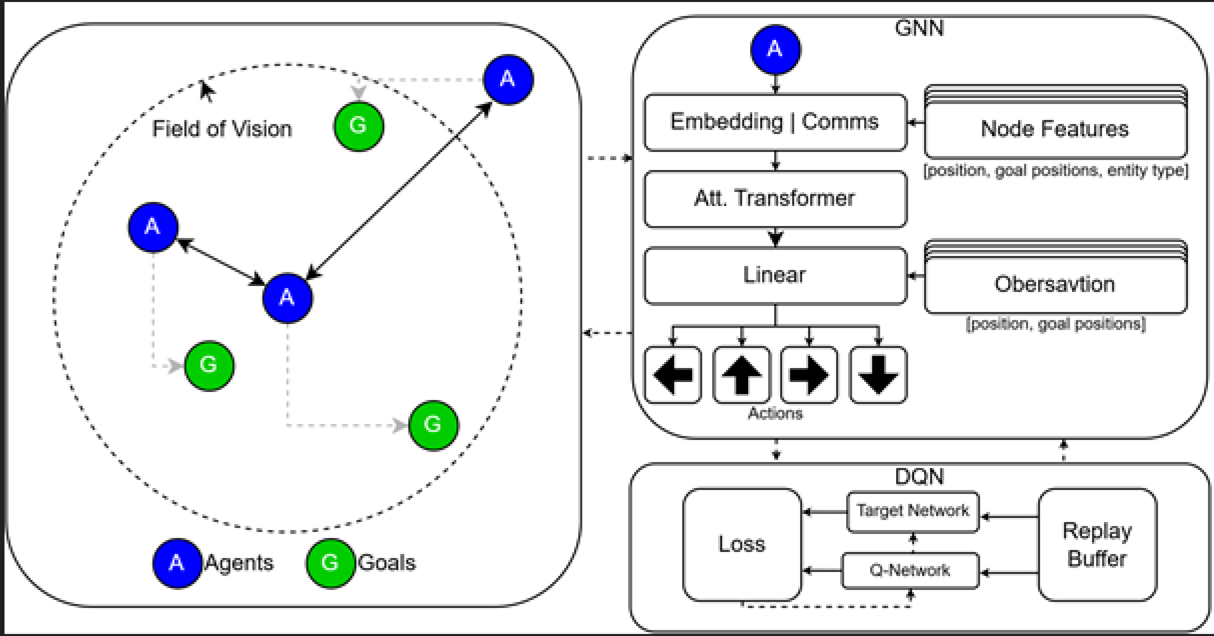
My Approach

Framework integrating:

- Graph Neural Network (GNN)
- Transformer-based message processing
- Deep Q-Learning

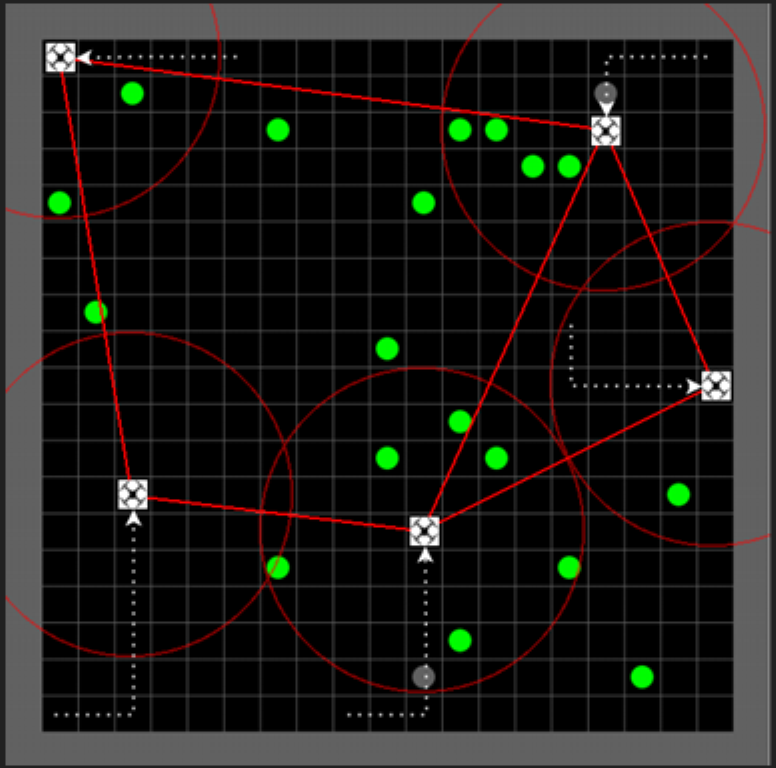
Key advantages:

- Efficient information exchange
- Enhanced coordination
- Adaptive decision-making



System Model

- 2D grid environment
- Agents represented by drone images
- Randomly distributed objectives shown as green/gray dots
- Constraints:
    - Vision radius (red circles)
    - Communication links (red lines)

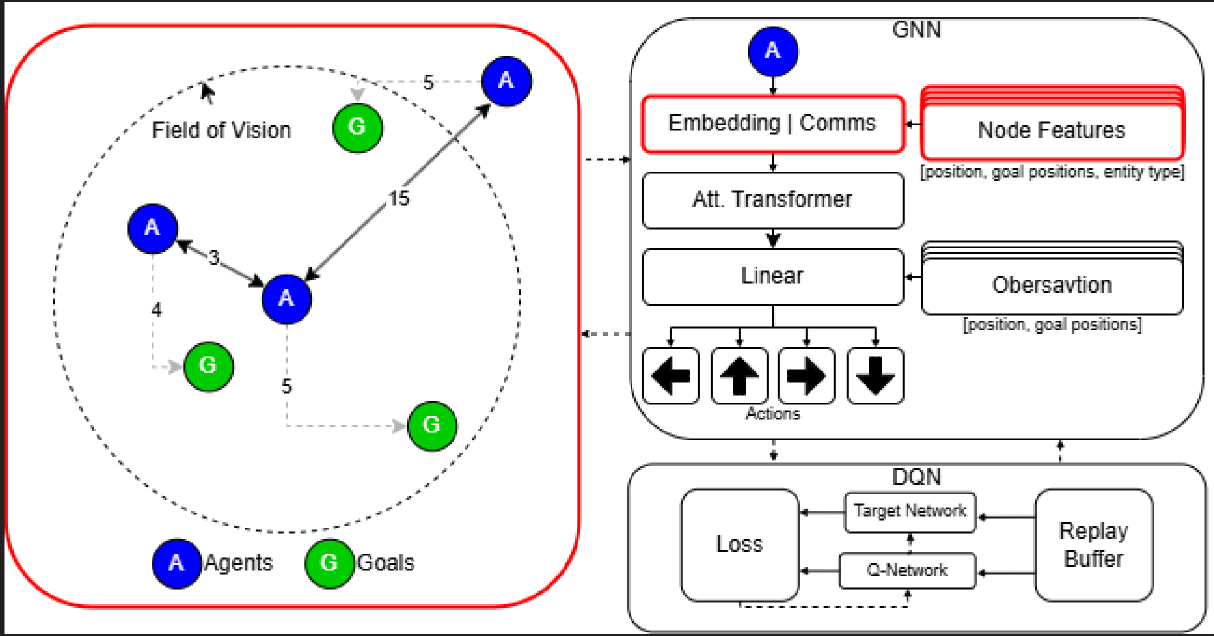


Network Architecture

Graph Neural Network:
- Entity embedding layer
- Node feature processing

Transformer:
- Multi-head attention
- Computes edge weights
- Feature prioritization

Deep Q-Learning:
- Off policy algorithm
- Prioritized experience replay buffer

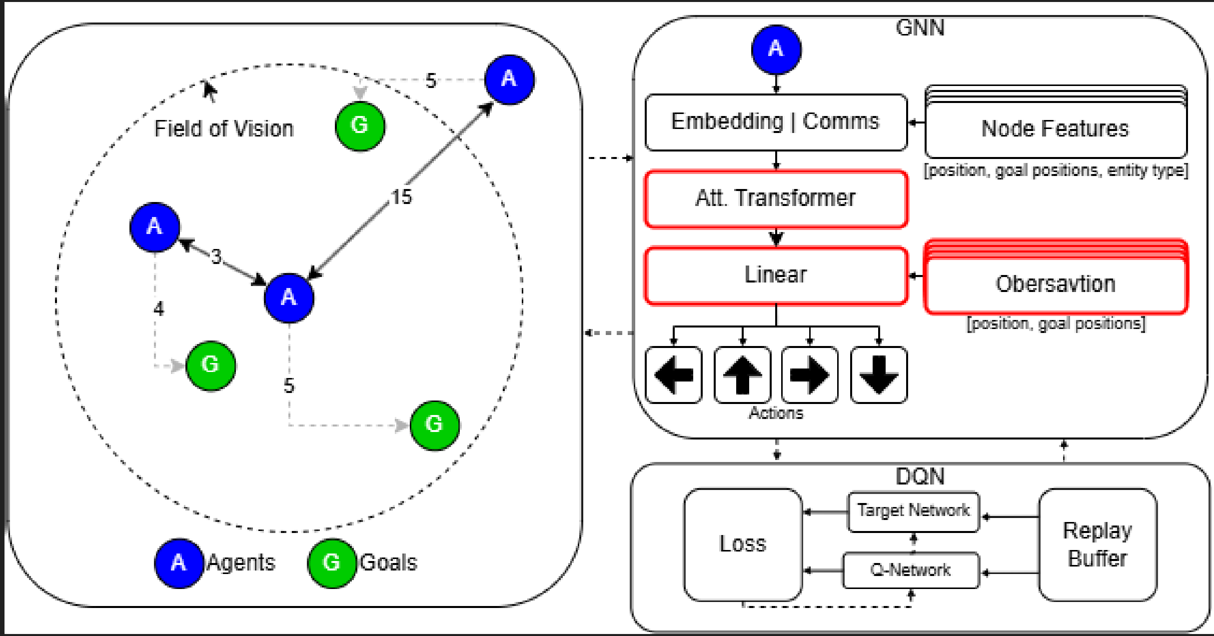


Network Architecture

Graph Neural Network:
- Entity embedding layer
- Node feature processing

Transformer:
- Multi-head attention
- Computes edge weights
- Feature prioritization

Deep Q-Learning:
- Off policy algorithm
- Prioritized experience replay buffer

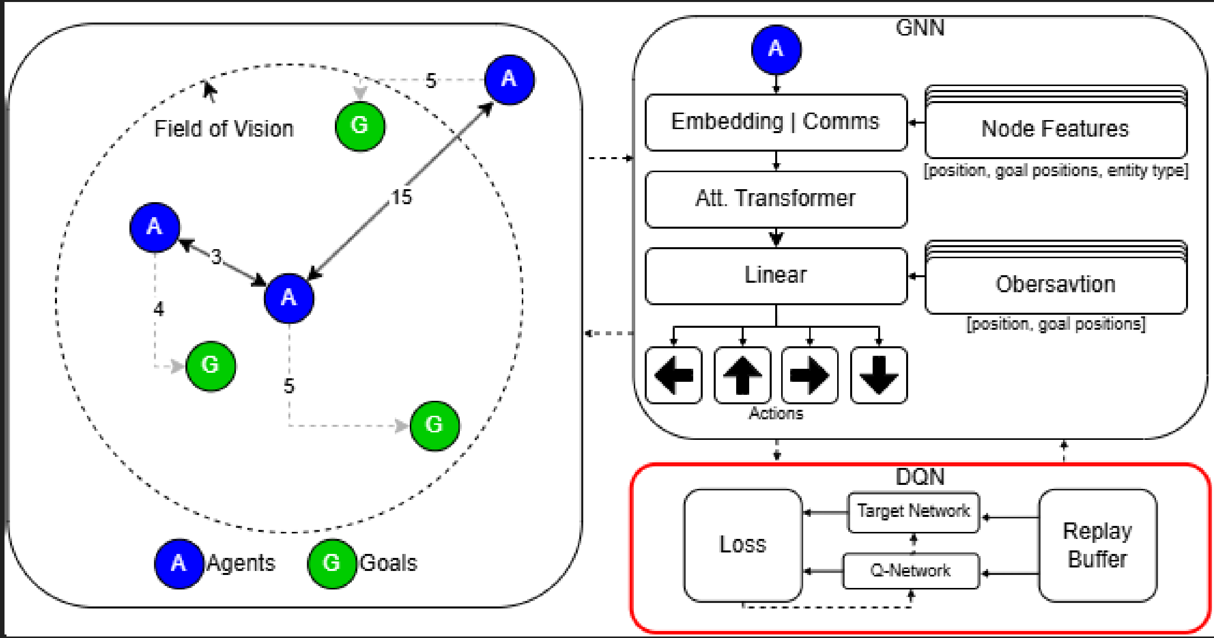


Network Architecture

Graph Neural Network:
- Entity embedding layer
- Node feature processing

Transformer:
- Multi-head attention
- Computes edge weights
- Feature prioritization

Deep Q-Learning:
- Off policy algorithm
- Prioritized experience replay buffer



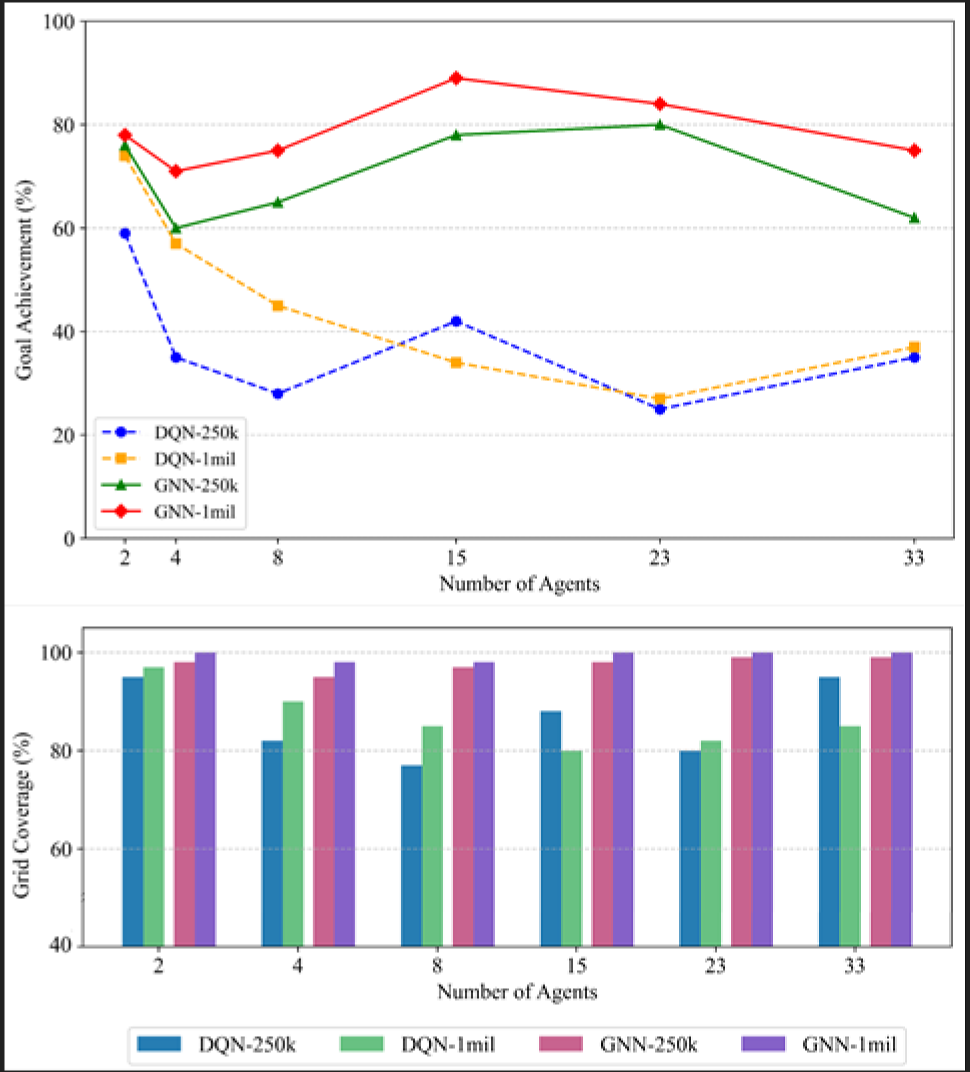
Experimental Setup

- Environment configurations:
  - From 2 agents, 10 objectives
  - To 33 agents, 170 objectives
- Performance metrics:
  - Objective visitation percentage
  - Grid coverage percentage
  - Time steps per session

Agents	Objectives	Grid Size	Time Steps/Session
2	10	10x10	150
4	20	20x20	150
8	45	30x30	175
15	75	40x40	200
23	120	50x50	250
33	170	60x60	300

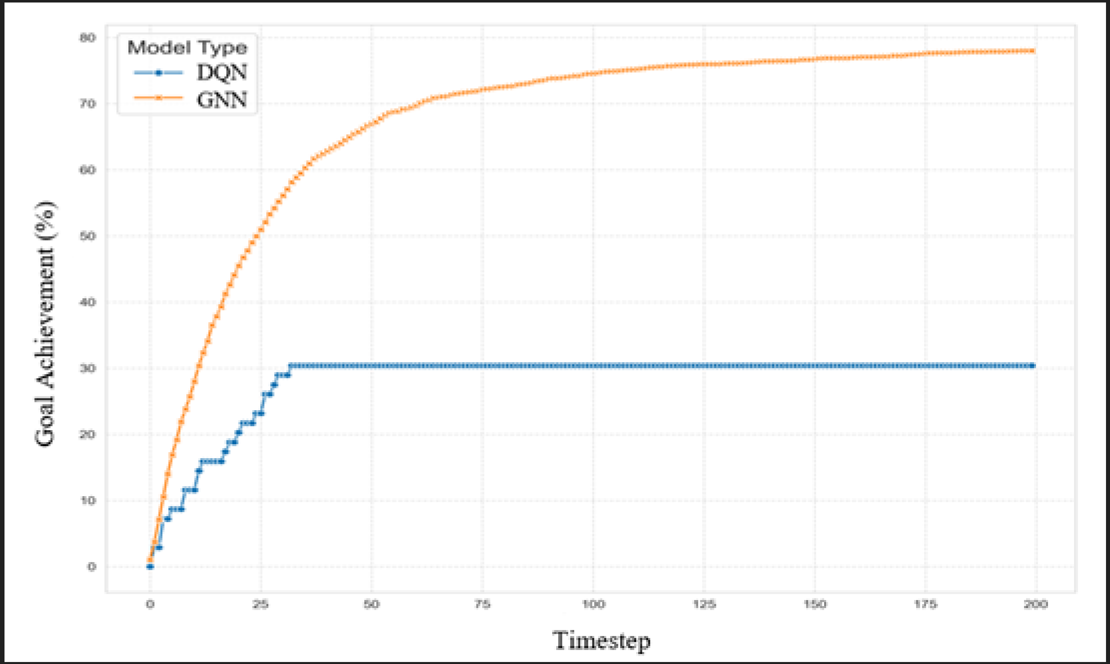
Performance

- GNN consistently outperforms baseline DQN
- Performance gap widens with scale
- 90% objective completion vs. 42% baseline (15-agent, 40×40 grid)
- Near-complete grid coverage in all configurations



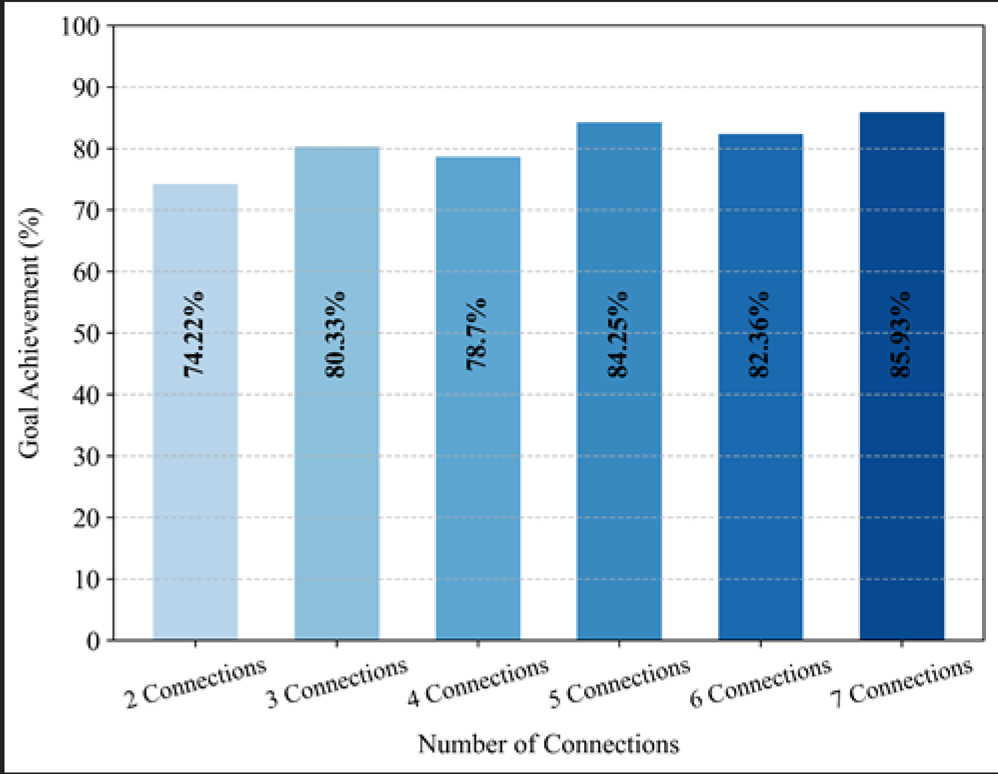
Temporal Performance

- 65% of objectives visited within the first 50 timesteps
- ~75% collected by the halfway point (~100 steps)
- Baseline DQN achieves only ~30% overall in the same environment
- Speed of collection is critical for energy-constrained real-world missions



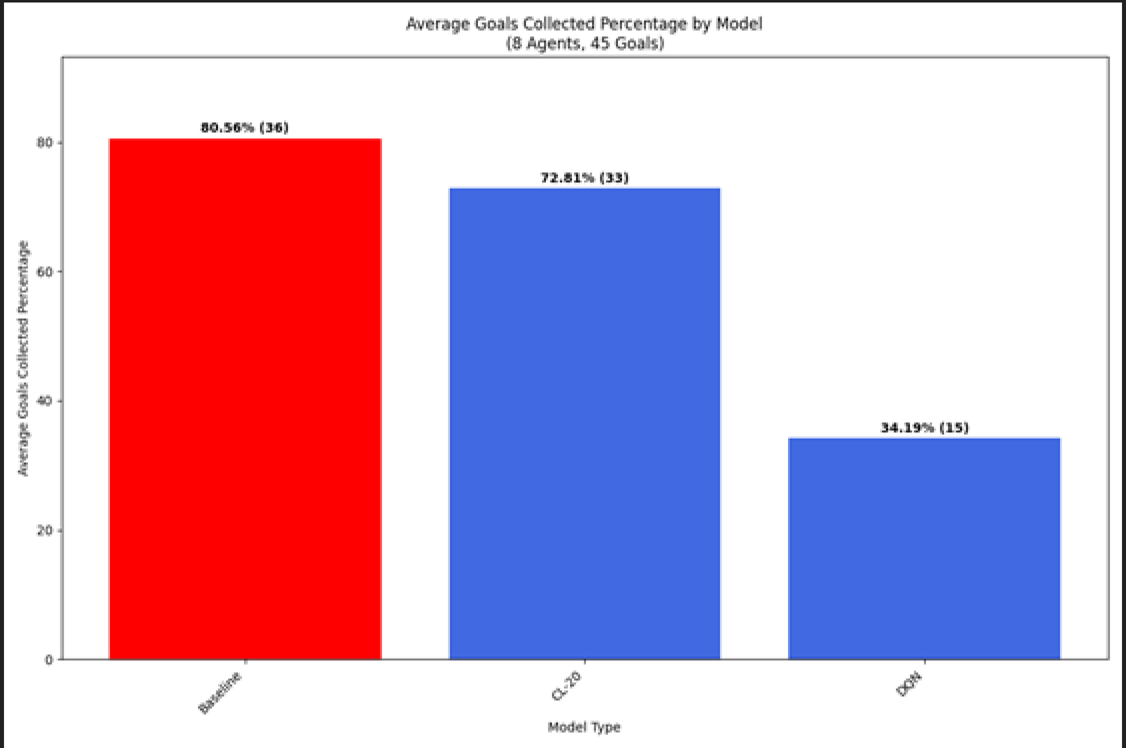
Ablation Study: Connection Limits

- Impact of varying agent communication connections
- Performance across different connection limits
- Diminishing returns beyond 5 connections
- Finding optimal communication-performance balance



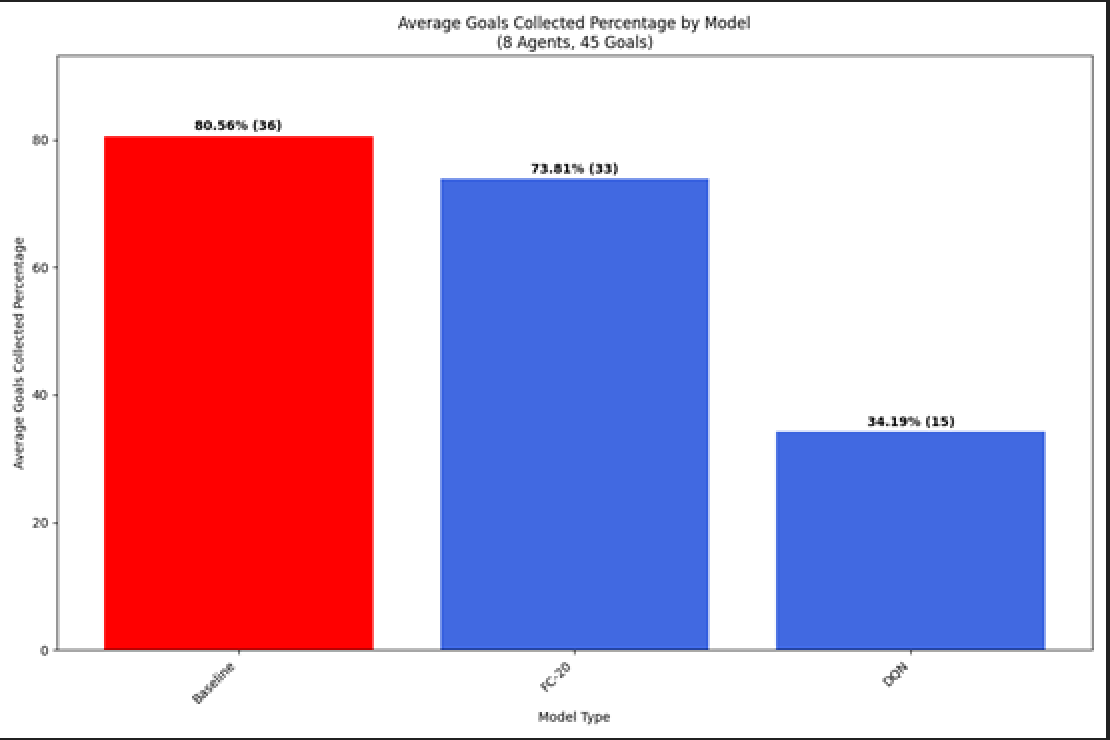
Ablation Study: Communication Latency

- Simulating real-world communication delays
- 20% chance of receiving outdated information
- System resilience to communication delays
- Implications for real-world deployment



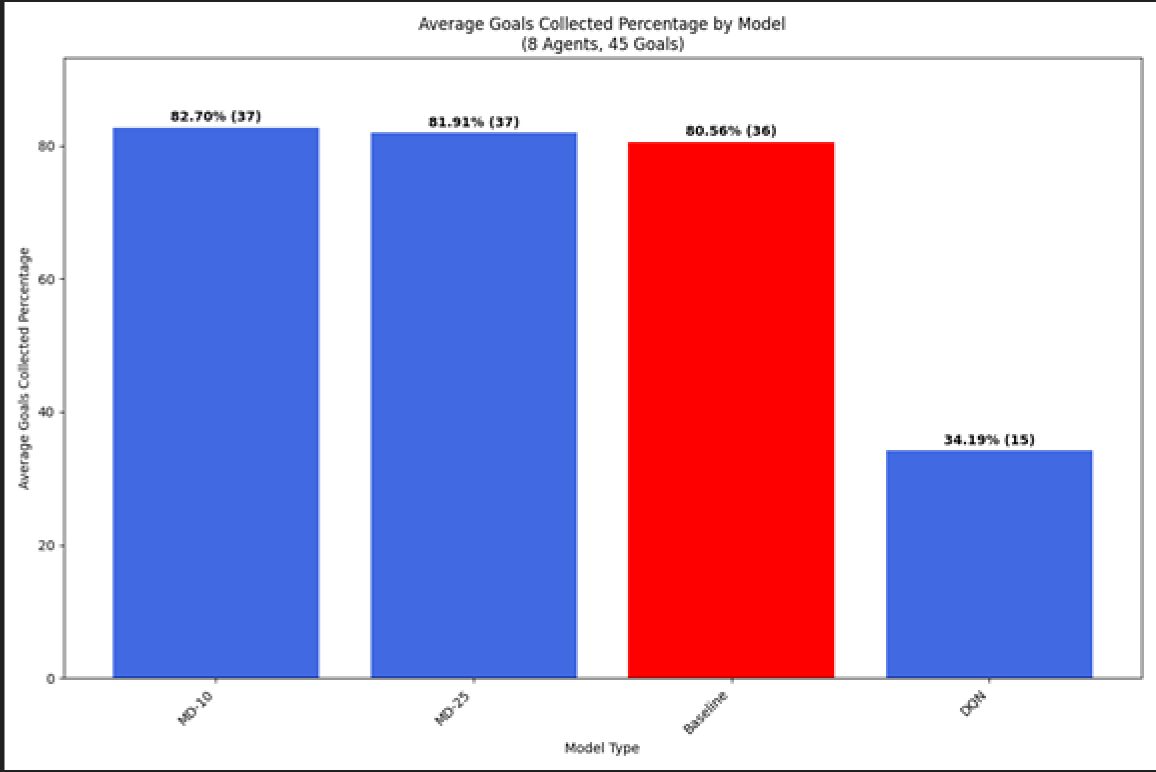
Ablation Study: Feature Corruption

- Testing resilience to corrupted information
- 20% chance of Gaussian noise in messages
- Robust information processing
- Attention mechanism prioritizes reliable data



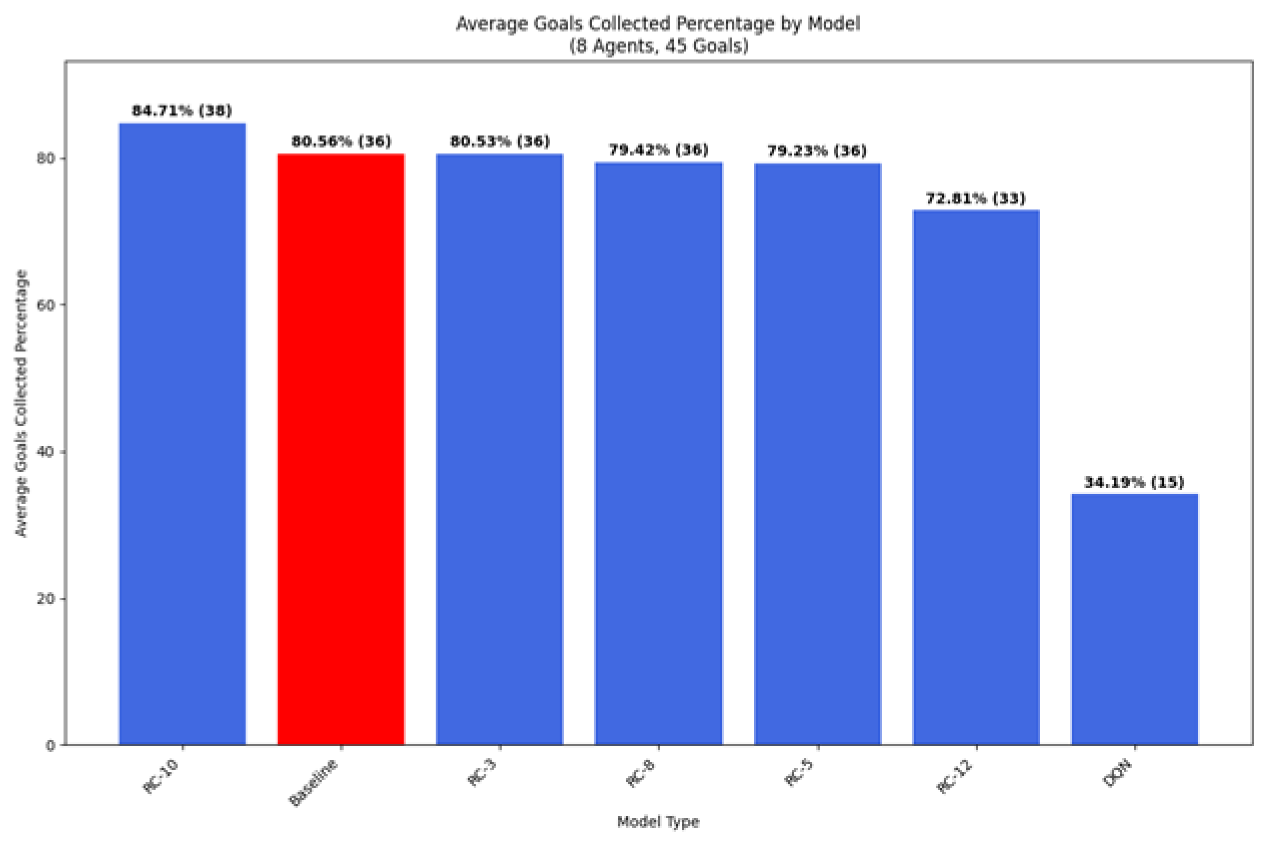
Ablation Study: Message Dropping

- Simulating complete communication failures
- Results show virtually identical performance to baseline
- Connection to earlier findings on communication redundancy
- Implications for robust system design



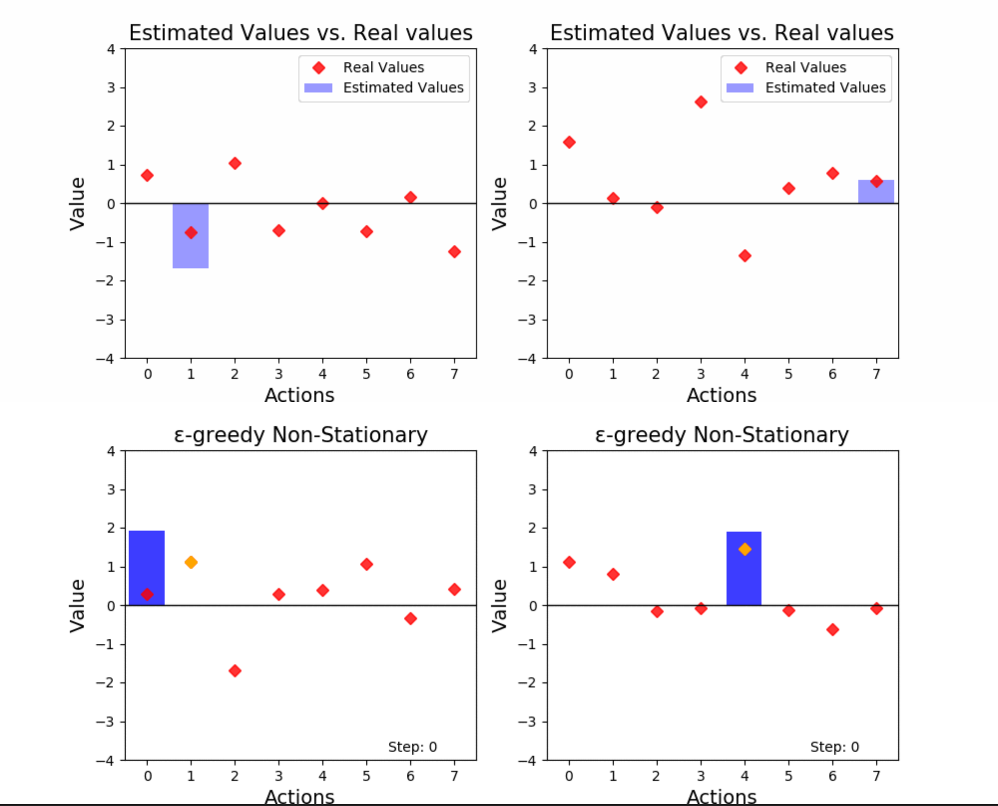
Ablation Study: Communication Frequency

- Less frequent communication can outperform continuous updates
- Peak performance at 10-step intervals (84.71% vs. 80.56% baseline)
- Performance drops below baseline beyond ~12-step intervals
- Suggests constant updates may interrupt local decision-making



Non-Stationarity Problem

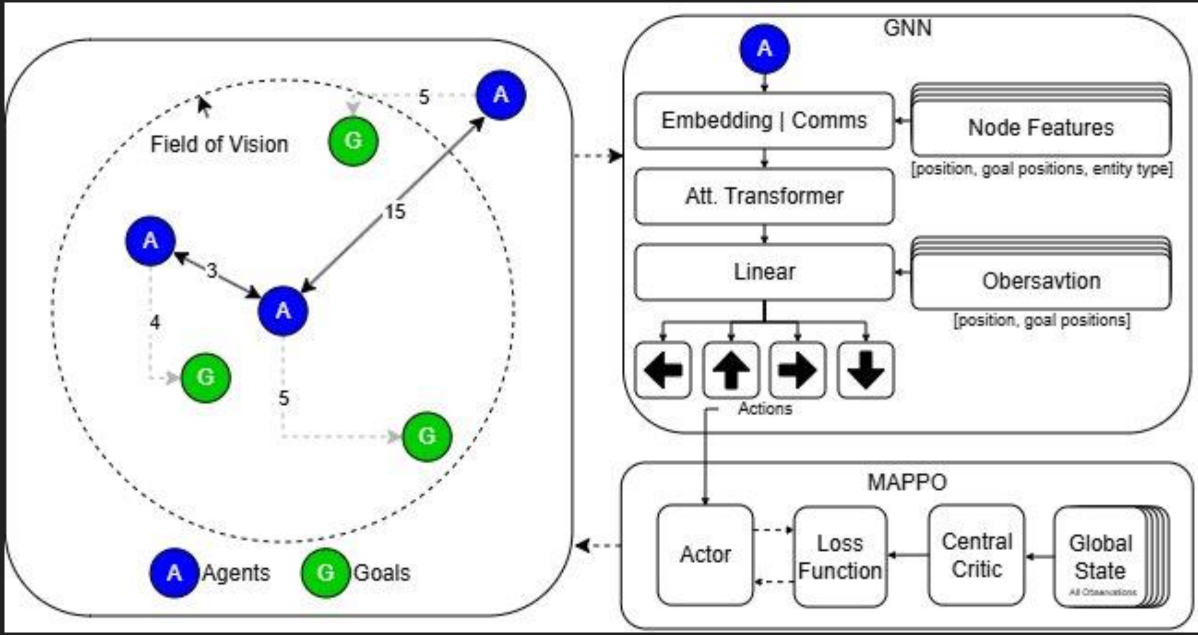
- DQN assumes a stationary environment, invalid when all agents are simultaneously updating
- From any one drone's perspective, the world constantly shifts
- The replay buffer compounds this: stored experiences go stale too quickly
- Problem scales with fleet size
- Known in multi-agent RL as the **non-stationarity problem**



Understanding Reinforcement Learning Hands-on: Non-Stationary
Alejandro Aristizabal

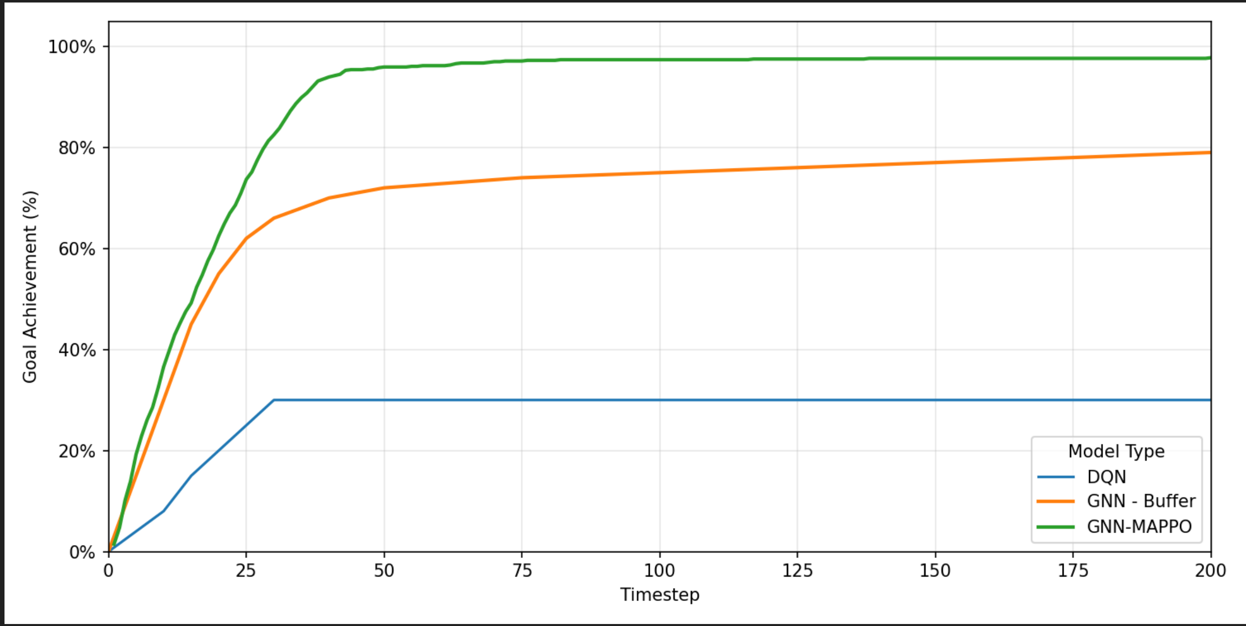
Multi-Agent Proximal Policy Optimization

- Replaces DQN with an on-policy algorithm, no stale experiences by design
- Each drone still acts on local observations only
- A **centralized critic** during training evaluates global state
- GNN architecture is **completely unchanged**
- Early results: **98-100%** average objective visitation



Multi-Agent Proximal Policy Optimization

- Replaces DQN with an on-policy algorithm, no stale experiences by design
- Each drone still acts on local observations only
- A **centralized critic** during training evaluates global state
- GNN architecture is **completely unchanged**
- Early results: **98–100%**



Conclusion & Future Work

- ML framework integrating GNNs, transformers, and deep reinforcement learning
- Superior performance:
  - 90% goal collection (vs. 42% baseline)
  - Near-complete grid coverage
  - 67% reduction in steps per episode (200 vs. 600)
- Preliminary MAPPO results: 98–100% objective visitation, suggesting the performance ceiling is higher still
- Key innovations:
  - Adaptive graph construction
  - Edge-feature-enhanced attention
  - Communication-resilient architecture
- Future work
  - Validate MAPPO results at scale
  - Incorporate raw sensor inputs (camera, LiDAR) to remove reliance on coordinate observations

Acknowledgments

- AI-SENDS Lab, Clemson University
  - Dr. Abolfazl Razi
  - Dr. Long Cheng
  - Dr. Jim Martin
  - Dr. Manveen Kaur
  - Niloufar Mehrabi
- MIT Lincoln Laboratory
  - Dr. Rahul Amin
  - Group 65
  - 2024 Interns

Q&A