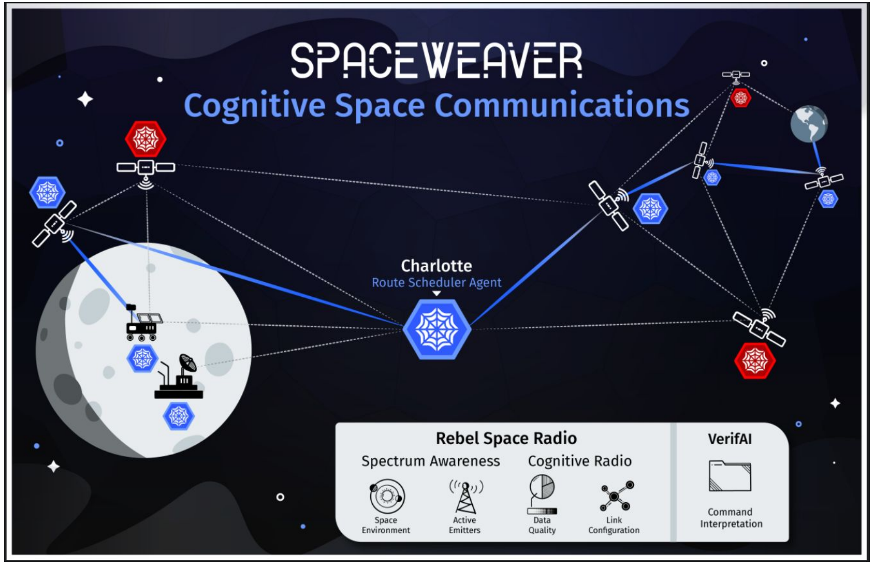
Scaling Collaborative Space Networks with Deep Multi-Agent RL

REBEL SPACE TECHNOLOGIES

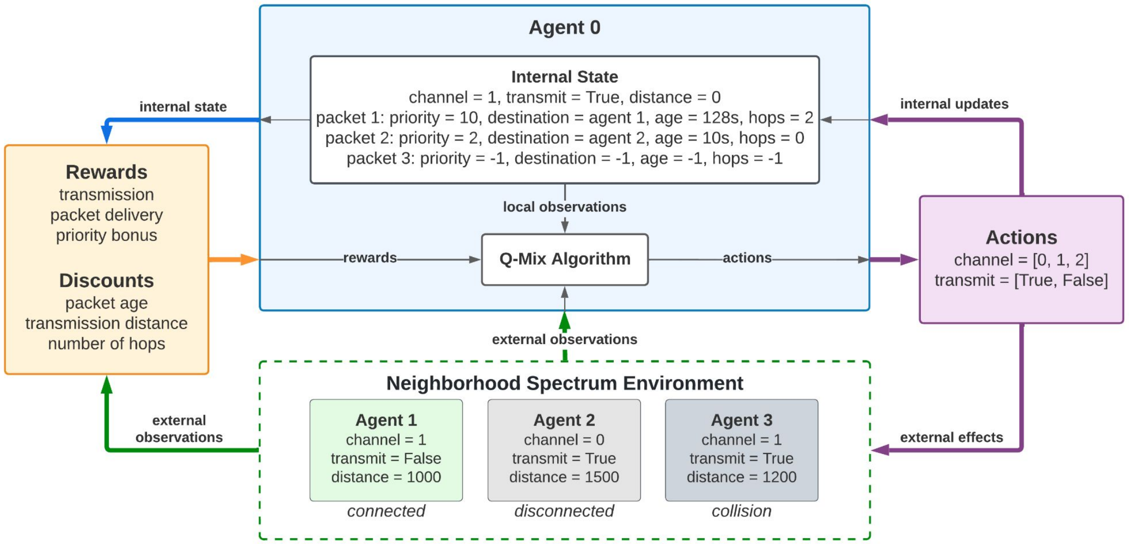
NASA

Scaling applied RL is hard
Scaling applied RL in space is even harder

- Future space communication architectures will require novel methods of coordination inter-system communication
- Ability to facilitate communication across a large number of dynamic systems is a significant constraint on mission performance
- Recent publications have experienced challenges when scaling existing MARL networks without retraining every agent from scratch



A Scalable Model for MARL Applications in Delay Tolerant Networks



- Actions:
  - Transmission channel
  - Transmit/receive
- Observations:
  - States of 3 nearest neighbors
  - States of 3 oldest packets
- Rewards:
  - Successful transmission
  - Packet delivery + bonus
- Discounts:
  - Packet age + passes
  - Transmission distance

Charlotte Simulation Environment

Gymnasium

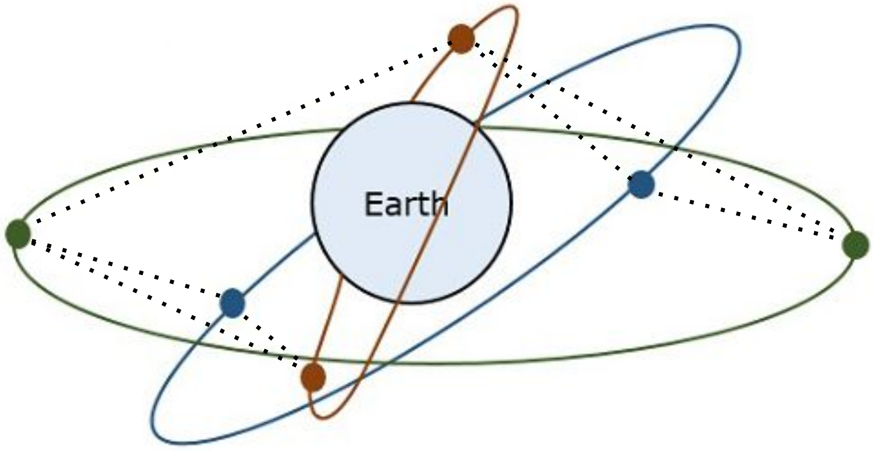
RAY

poliastro
Astrodynamics in Python

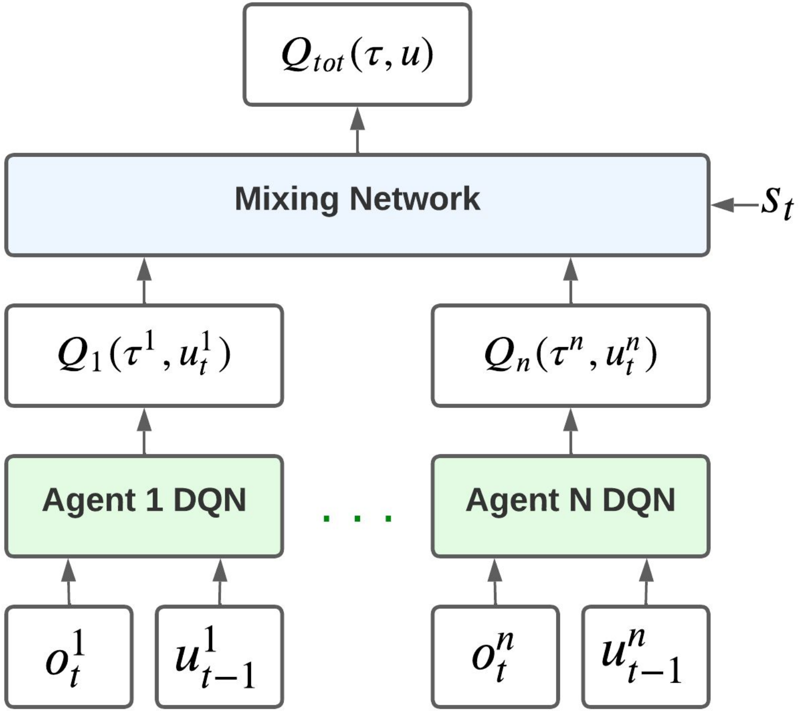
PettingZoo

RLlib

PyTorch



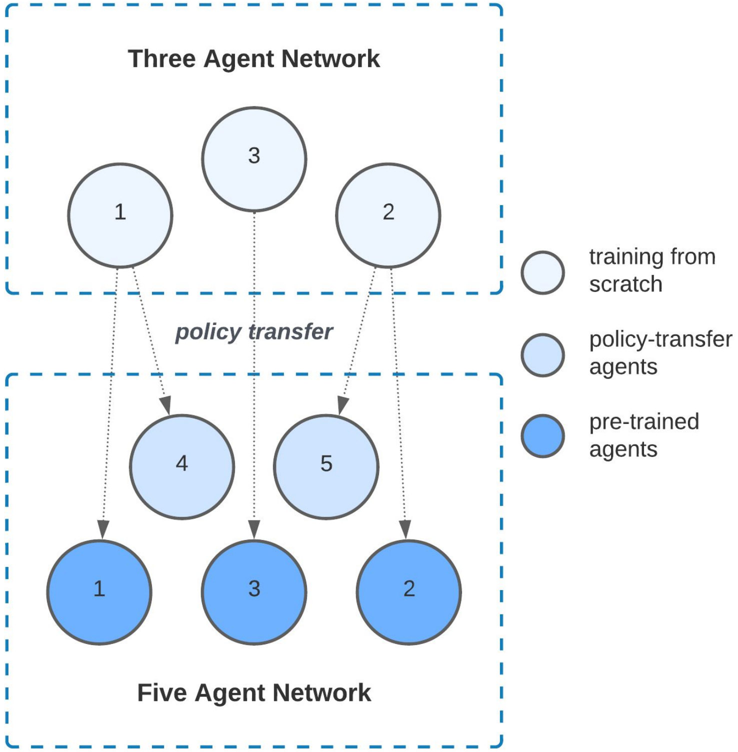
Q-Mix for Deep Multi-Agent RL



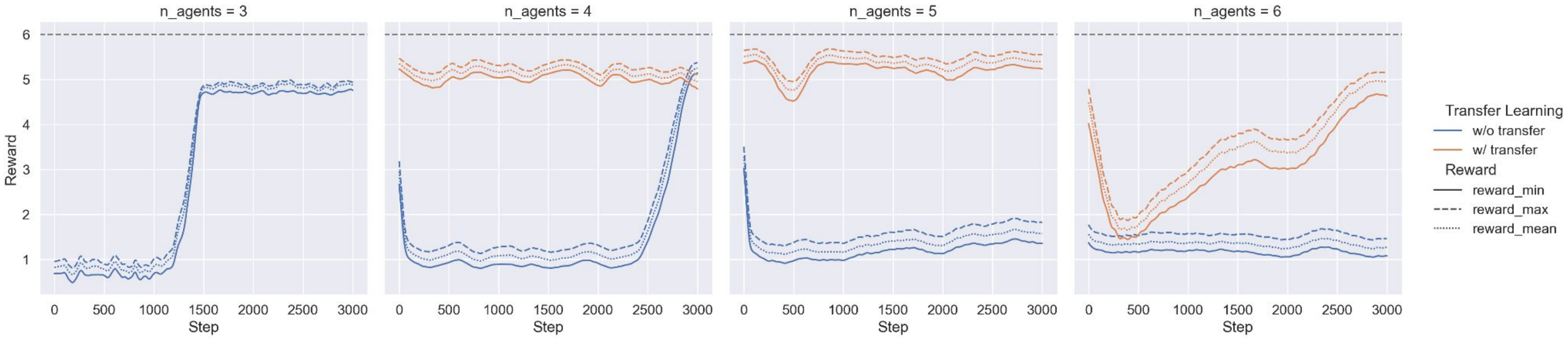
- Centralized training through mixing network, combats non-stationarity of multi-agent environment
- Decentralized execution through individual deep Q-networks
- Each agent has its own observations, actions, and rewards

Multi-Agent Policy Transfer

- Learning from scratch is impractical and typically infeasible in non-primitive multi-agent scenarios
  - Explosion in sample complexity as number of agents increase
  - Reusing previously learned knowledge helps to scale up MARL algorithms
- With our model, individual actions and observations do not depend on the number of agents in the network
  - Individual pre-trained DQN policies can be transferred to new agents
  - Agents are then fine-tuned within this new environment

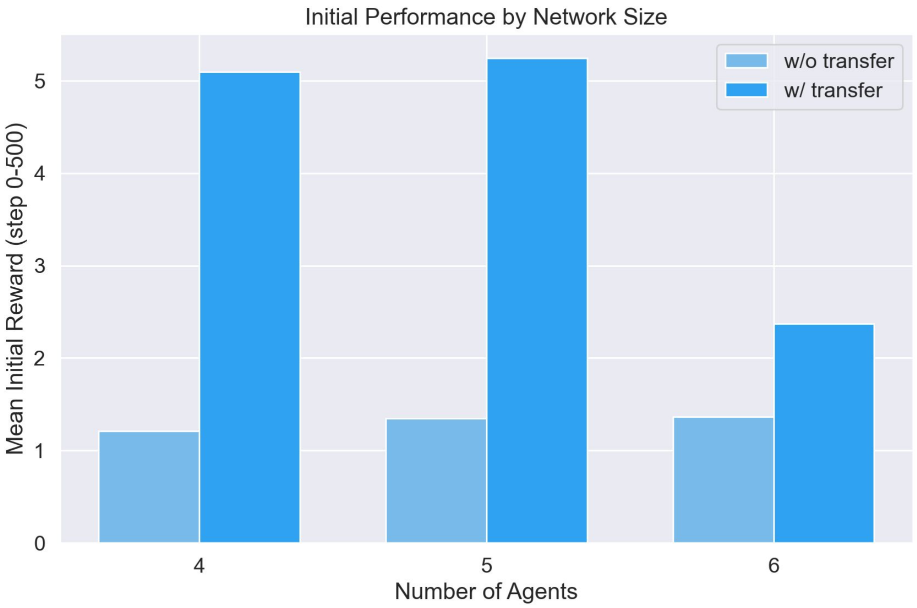


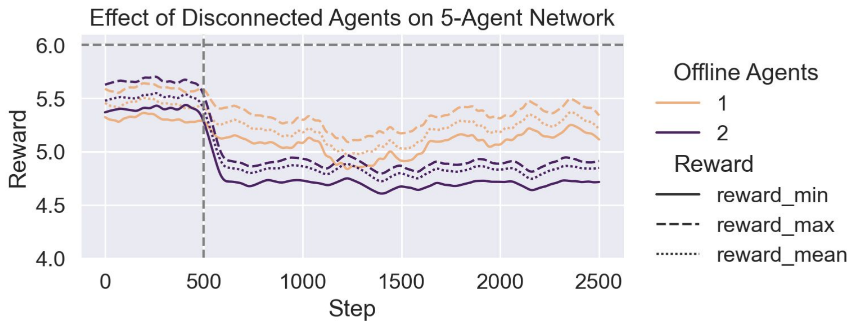
Multi-Agent Policy Transfer



- When training from scratch:
  - 3 agent network converges in 1500 steps, 4 agent network converges in 3000 steps
  - 5 and 6 agent networks do not converge in under 3000 steps
- With pre-trained policies:
  - 4 and 5 agent networks settle immediately on near-optimal solution
  - 6 agent network takes 3000 steps to find near-optimal solution

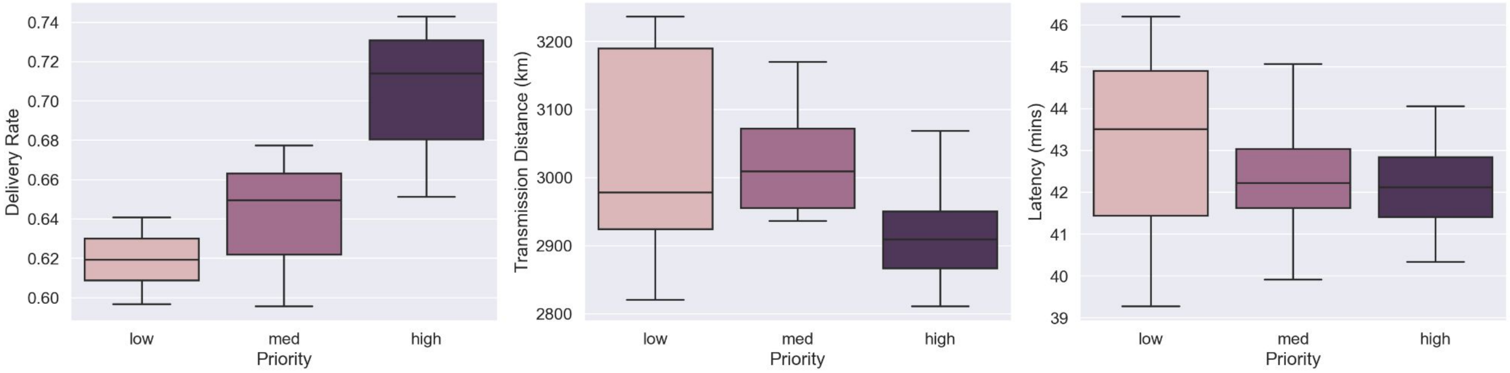
Efficiency and Robustness of Policy Transfer





- Dramatic differences in initial performance of networks with and without transferred pre-trained policies
- Noticeable but tolerable decrease in network performance when 20% (1) and 40% (2) of agents are taken offline

Collaborative Prioritization of Packet Delivery



- Packets are given low, medium, and high priority, where a higher priority corresponds to higher delivery reward
- High priority packets are more likely to be delivered within a full LEO orbit (~100 mins)
- To conserve data quality, high priority packets are sent at shorter transmission distances
- On average, low priority packets have a higher transmission latency

Conclusion

- Developed multi-agent RL simulation environment and training pipeline for delay-tolerant networking
- Presented transfer learning framework for effectively pre-training and scaling multi-agent networks
- Demonstrated efficiency and robustness of network when multiple agents are dynamically added and/or removed
- Established ability for multi-agent networks to collaboratively prioritize packet delivery based on packet importance

Thank You

rebelspacetech.com

REBEL SPACE
TECHNOLOGIES

NASA