Independent Project Proposal

Towards Optimally Decentralized Multi-Robot Collision Avoidance via Deep Reinforcement Learning

Yifan Zuo - 20876522

WHAT PROBLEM WILL STUDY

The problem we will study is how robots plan for their path in dynamic environments and decentralized scenarios where each robot generates its paths without observing other robots' states and intents using deep reinforcement learning. How do we train the policy using a gradient-based reinforcement learning algorithm?

Can the learned policy be generalized to new scenarios that do not appear in the whole training period?

WHY DO YOU FIND THE PROBLEM INTERESTING OR IMPORTANT

MAPF system is important for warehouse and intelligent robots system for sorting and mobility-on-demoand services and its real-world application has attracted increase attention from researchers.

While other centralized methods such as CBS, Prioritized Planning, and decentralized methods exist which often require prohibitive computation and weak robustness. Interestingly, the decentralized methods often perform worse than their centralized counterparts. Therefore, we aim to improve the decentralized method to find a time efficient, collision free paths for a large scale robot system.

PRELIMINARY PLAN ON THE METHODOLOGY

Overall structure

The MAPF problem can also be transformed into a POMDP problem, and a reinforcement learning framework can be designed to solve this problem.

Deep Reinforcement learning

- The key idea in deep reinforcement learning for MAPF is finding a policy π (*s, a*) that can maximize expected future return. DQN is one of the popular deep reinforcement learning methods that is currently and widely used in single-agent and fully-observable RL settings. At each time step *t*, the agent obtains the current state *s*t∈*S* by interacting with the environment, and selects an action *a*t∈*A* according to the policy π. The agent aims to maximize the accumulated discounted expected reward *G*t, and *G*t .
- where *r*t is the reward received at time *t*. According to an action value function, the policy π is expressed as *Q*π (*s, a*) in DQN.
- The optimal action value was *Q**(*s, a*) = maxπ*Q*π(*s, a*). The action value function was learned by DQN using neural networks parameterized by θ, represented as *Q*(*s, a*;θ). The ε-greedy strategy was adopted to select actions during training.
- The parameters generally used by the target network were from previous (i − k) itera- tions. The DQN loss function can be defined.
- The MAPF problem can be viewed as a single agent moving in a dynamic multi-agent environment. One of the agents obtains (st , at , rt , st+1) through interaction with the environment at time *t* and then uses Equation to calculate the reinforcement learning loss.

Hot Supervision Contrastive Loss

- transformexpertdata into supervised signals to train policy neural networks that can be trained in combination with deep reinforcement learning methods.