COMP0036 – Project Plan:
Multiagent Reinforcement Learning algorithms for fully cooperative coordination tasks

Jeffrey Li, supervised by **Professor Mirco Musole**

BSc Computer Science, UCL

November 7, 2022

1. Aims and objectives

Aims:

To study and compare a wide range of popular Multiagent Reinforcement Learning (MARL) algorithms in the context of fully cooperative agent coordination with the assumption that agents are able to freely communicate with each other, but their communication channel can be restricted depending on the environment that they are in. (*See Section 2*)

Ultimately, I would ambitiously propose my own/improve on existing algorithms for such cooperative tasks.

Objectives:

(*See Section 2. Expected deliverables for additional detail*)

1.1. Review on Reinforcement Learning, Deep Reinforcement Learning concepts
1.2. Research and understand underpinnings of MARL through toy experiments
1.3. Implement the listed MARL algorithms.
1.4. Train and test implemented algorithms on the listed cooperative environments.
1.5. Evaluate success of implemented algorithms with each other and with the baseline Independent Q Learning algorithm and perform in-depth analysis based on the results.
1.6. Devise a new algorithm/improve on the implemented MARL algorithms for multiagent cooperation.

2. Expected deliverables

The expected deliverable would start with a general literature survey summarizing core concepts of Reinforcement Learning, Deep Reinforcement Learning and Multiagent Reinforcement Learning for the more general readers. The literature survey would be backed by Python 3.x implementations for the following RL and Deep RL algorithms and are trained and tested in OpenAI Gym's toy environments:

- Value Iteration
- Policy Iteration
- Q Learning
- Deep Q Learning

The project report would then have a focus on cooperative MARL and would also be supported by Python 3.x implementations of the following MARL algorithms:

- Independent Q Learning [1]

Value Decomposition methods:

- Value-Decomposition Networks [2]
- QMIX [3]

Policy Gradient methods:

- MAPPO [4]

These implementations would be trained for a fixed 10,000 episodes and tested on a selective of benchmark multiagent environments for cooperative tasks as shown below, followed by an in-depth analysis and evaluation:

- Hanabi [5]
- Mult Particle Environments [6]
    - Simple Spread
    - Simple World Comm
    - Simple Reference
- PRESSUREPLATE [7]

The evaluation of these algorithms would be done by comparing results against each other and the Independent Q Learning baseline in terms of mean episodic reward in training and in testing. The results would be obtained on varied number of cooperative agents and objectives depending on the environment with fixed network hyper-parameters.

As a conclusion, I will analyse strengths and weaknesses of these algorithms based on the evaluation and ambitiously propose an improved algorithm based on the investigated algorithms or give rise to a new approach for cooperative MARL, which I hope to implement and evaluate in the future projects.

3. Work plan

When a MARL algorithm is mentioned, it means the fully implementation, testing and documentation of corresponding algorithm

- Project start to mid-Nov (6 weeks)
    - Complete implementation and testing on RL and DRL algorithms
    - Complete and get Project plan approved by supervisor
- Mid-Nov to mid-Dec (4 weeks)
    - Independent Q Learning and Value Decomposition Networks
- Dec 19th to Early-Jan – Christmas Break (3 weeks)
    - QMIX
- Early-Jan to 18th Jan (2 weeks)
    - Work on completing the interim report
- **18th Jan - Interim Report Due**
- 18th Jan – Late-Feb (5 weeks)
    - MAPPO
- Late-Feb to mid-Mar (3 weeks)
    - Devise my own cooperative MARL algorithm
- Mid-Mar to 26th April (6 weeks)
    - Work on completing Project report
- **26th April - Project Submission**

References:

[1] https://web.media.mit.edu/~cynthiab/Readings/tan-MAS-reinfLearn.pdf

[2] https://arxiv.org/abs/1706.05296

[3] https://arxiv.org/abs/1803.11485

[4] https://arxiv.org/abs/2103.01955

[5] https://pettingzoo.farama.org/environments/classic/hanabi/

[6] https://pettingzoo.farama.org/environments/mpe/

[7] https://github.com/uoe-agents/pressureplate