COMP0036 – Project Plan:
Multiagent Reinforcement Learning for Noised Communication in Fully Cooperative MPEs

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

BSc Computer Science, UCL

November 17, 2022

1. Project Overview

Aims:

To devise a centralized learning and decentralized execution Multiagent Reinforcement Learning algorithm with the hope for agents to learn communication protocols to effectively communicate over noisy channels, such that agents can achieve coordination tasks in selected Multi-Particle Environments. The algorithm takes into account the following assumptions and constraints:

Assumption and constraints

- Tasks to solve are fully corporative with no antagonistic agents.
- Individual agents have partial observability of the environment.
- Agents can fully communicate with any other nearby agents in its observable space with no constraints on who it can communicate with
- The communication channel is noisy but with no limits in bandwidth.
- The algorithm does not assume a differentiable communication channel between agents, the channel can be discrete.
- All messages would have a pre-set encoding which agents wouldn't need to learn about.
- Learned policies can only make use information local to each agent at test time
- The noise added to the communication channel is unknown to the agents.

Objectives:

1.1. Review on Reinforcement Learning, Deep Reinforcement Learning concepts.
1.2. Research and understand underpinnings of MARL in the context of coordination through communication.
1.3. Elevate the understanding to the context of communicating in noisy channels.
1.4. Conduct in-depth literature review on related topics and methods
1.5. Devise my algorithmic solution for this problem mathematically and in pseudocode.
1.6. Implement the proposed pseudocode in Python.
1.7. Train and test the Python implementation on the listed MPE environments.

1.8. Evaluate success of implemented algorithms with other state-of-the-art MARL algorithms.

2. Expected deliverables

The expected deliverable would start with an overview of the problem to be solved, explaining its motivation and context.

This is followed by a broad 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 that are trained and tested in OpenAI Gym's toy environments:

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

The report would then have a greater focus on cooperative MARL in the context of coordination through communication with a literature review over an array of algorithmic approaches on similar problems.

I would then propose my algorithm for solving the proposed problem. This would include mathematical formulations as well as pseudocode and would also be backed by my Python 3.x implementation of the algorithm.

This implementation would be trained and tested on a selective of benchmark MPEs [1] as shown below with modification in adding additional noises to agent's communication channels:

- Simple Speak Listener
- Simple Reference

And is then followed by an in-depth evaluation on its performance presented in the form of figures and graphs. This would be done by comparing results against the state-of-the-art algorithms for discrete communication listed below which would also be trained on the same MPEs:

- DiffDiscrete [2]
- RIAL [3]

The metrics used for comparison would be the mean episodic reward in training and in testing. The results would be obtained on varied number of cooperative agents with optimal (to my best ability) network hyper-parameters for each environment.

In conclusion, I will perform analysis on strengths and weaknesses of my proposed algorithm based on the evaluation and offer future work to be done as well as areas for improvements.

3. Work plan

- Project start to mid-Nov (4 weeks)
    - Complete reviewing, implementing and testing on Reinforcement and Deep Reinforcement Learning algorithms
- Mid-Nov to mid-Dec (4 weeks)
    - Complete Project plan and Literature review on proposed topic
    - Apply modification on selected MPEs and test the environments.
    - Come up with framework for the algorithm
- Dec 19th to Early-Jan – Christmas Break (3 weeks)
    - Continue working on developing the algorithm
- Early-Jan to 18th Jan (2 weeks)
    - Work on completing the interim report
- **18th Jan - Interim Report Due**
- 18th Jan – Mid Mar (7 weeks)
    - Finalise mathematical formulation and pseudocode for my algorithm
    - Finish implementation of pseudocode in Python.

- Obtain training and testing results of the implementation in the modified MPEs.

➢ Mid-Mar to Late-Mar (2 weeks)
  - Implement listed state-of-the-art algorithms.
  - Obtain training and testing results for these algorithms in the modified MPEs.

➢ Late-Mar to 26th April (5 weeks)
  - Work on completing Final Project report

➢ **26th April - Project Submission**

References:

[1] R. Lowe, Y. Wu, A. Tamar, J. Harb, P. Abbeel, and I. Mordatch, "Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments." arXiv, 2017. doi: 10.48550/ARXIV.1706.02275.

[2] Freed, B., Sartoretti, G., Hu, J., & Choset, H. (2020). Communication Learning via Backpropagation in Discrete Channels with Unknown Noise. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 34, Issue 05, pp. 7160–7168). Association for the Advancement of Artificial Intelligence (AAAI). https://doi.org/10.1609/aaai.v34i05.6205

[3] Foerster, J. N., Assael, Y. M., de Freitas, N., & Whiteson, S. (2016). Learning to Communicate with Deep Multi-Agent Reinforcement Learning (Version 2). arXiv. https://doi.org/10.48550/ARXIV.1605.0667