Project Proposal

William Bailkoski 2324599

In this project, I aim to explore the topic of Multi-Agent Reinforcement Learning (MARL). MARL is a diverse topic with a wide variety of applications due to its powerful modelling capabilities. More specifically, I plan to look at a significant topic in the field: decentralisation.

Consider a vehicle travelling on a 2-dimensional plane, travelling at a constant speed. It is being monitored by a network of stationary drones, each equipped with a sensor. The drones collectively try to maximise their accuracy in their estimation of the position of the vehicle by transmitting either 'raw' or 'processed' data to a central computer. The transmission of data incurs a time delay, making the data received by the computer slightly outdated. In addition to this, processed data suffers an extra time penalty for the processing (eg: compression, noise-averaging). Alternatively, the raw data is drastically less accurate, and as such a latency-accuracy trade-off is created. It is the responsibility of the central computer to determine how many sensors should transmit processed data for a given time-step. This is the framework posed by [1] which demonstrates the capability of Q-Learning in this scenario and provides us with a case study using 4 homogenous sensors and realistic values for the time-steps and delays.

The problem can be considered as a Markov Decision Process (MDP), defined by a tuple $M = (S, A, r, P, \gamma)$ where S is the state space, A is the action space (ie: $A = \{0,1,2,3,4\}$ "number of sensors sending processed data"), and r is the reward function for a given $s \in S$, $a \in A$. P is the transition kernel such that for every (s,a) there is a probability distribution over S for the next state to transition to, and γ is the discount factor which controls the balance between exploring new options in the state space and exploiting actions that are known to give the highest reward.

The framework can be generalised to a few key components. The central computer requires an estimation algorithm (to determine the error covariance, which becomes the reward) and a Reinforcement Learning (RL) algorithm (to try and determine the optimal policy). However, this model comes with a few limitations.

Firstly, from a practical standpoint, centralised systems often lack robustness due to the single point of failure. Should there be a fault in the central computer, the entire model would cease operation. Secondly, from a theoretical standpoint, centralised systems often lack scalability due to the computational requirements being burdened onto one decision maker. The complexity of the MDP that needs solving grows larger as the State-Action space increases (namely, the action space increases). As such, factors like memory requirements and algorithmic complexity grow with the number of sensors.

A decentralised approach where each sensor becomes an individual agent would combat these issues. In this model, the state space would remain the same, and the action space would become $A_d = \{\text{raw, processed}\} = \{0,1\}$ for each agent in the system. A_d is a direct subset from our centralised action space A and as such we can easily adapt our transition kernel and reward function to create P_d and r_d.

This outlines the first aim of my project. To have agents evaluate the best actions to take individually to maximise the reward for the entire system, the first step is to implement parallel MDPs. Due to the homogeneity of our agents, there is an identical MDP for each agent in the system. This would create an environment where each agent is acting completely independently and could maximise its own

reward. However, our aim is to maximise the cumulative reward of the entire system, and there is a distinct disadvantage: the centralised model uses the information gathered by every agent to estimate the error covariance, whereas the decentralised model only gets one agent's data.

This leads into the second aim of this project. To combat this asymmetry in the information that an agent can use in its decision-making process, we allow the agents to communicate with each other to collaborate towards maximising the reward of the entire system, not just their own. I plan to do this with the development of a message passing protocol. Here, an agent could potentially share details of actions it plans to take or known rewards for a given state that has already been explored, for example. This highlights the main challenges of my project:

1. How can we develop a message passing protocol that effectively communicates the joint action space? This includes consensus mechanisms, and further maintaining stability when converging to a near-optimal policy.

2. In addition to this, can we minimise communication yet ensure that we remain efficient in our exploration of the State-Action space?

Literature Review

There has been a resurgence of interest in MARL due to developments in deep learning. Combined with its powerful modelling capabilities, MARL can be applied to a wide variety of disciplines. Some examples of this include its use in finance [2] and electrical engineering [3]. As such, there has been recent interest in the decentralisation of models and ensuring that they can be efficiently implemented.

There is lots of general literature in the field of MARL. An eclectic collection of many important MARL concepts comes from [4]. There is a brief breakdown of a variety of topics, including ones that hold significance in this project such as Q-learning, Markov team games, and joint action spaces. This provides a great starting point but lacks thorough detail.

The study of complexity in MARL is rich and can provide us with a lot of motivation for our project. [5] demonstrates a clear motive as decentralised MDPs (DEC-MDP) are NEXP-complete in certain scenarios. As such, it is important to ensure we approach this problem with parallel MDPs and an efficient message passing algorithm such as the one outlined by [6]. Alternatively, the algorithm proposed in [7] guarantees a convergence to team-optimality, which is distinctly different to Q-function equilibrium.

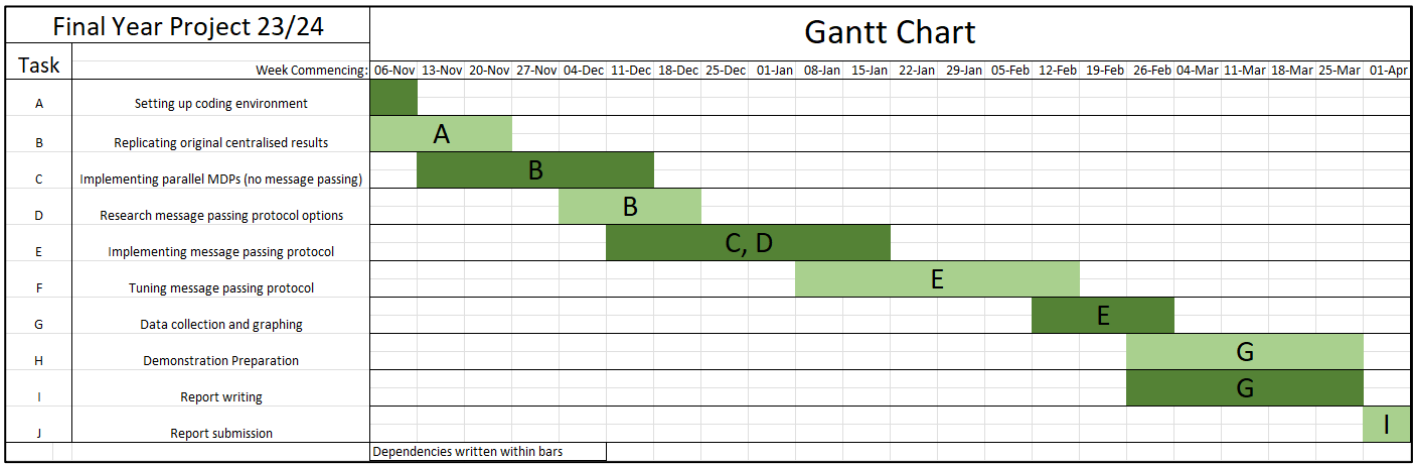
The most important paper to this project is [1], which lays down the framework for the entire project. It is defined generally enough to be applied to various distributed sensor models, beyond surveillance (such as monitoring the temperature of a room). Furthermore, a concrete realisation of the model is provided. This is then evaluated with Q-learning and highlights its advantages over some obvious policy choices. The optimal policy provided in the paper will be a crucial benchmark to evaluate the model that is created in this project.

In terms of modelling, the framework comes with a few limitations. The scenario in which all homogenous sensors can see the entire state space is heavy restrictive compared to a partially observable state space. In addition, there is no transmission delay for the base-station to communicate back to the sensors, which directly contradicts the default time delay that is factored

into all sensor transmissions, regardless of the action taken. However, this provides a foundation for future adaptation, which makes it more desirable to research now.

I believe that the adaptation of this framework into a decentralised model through parallel MDPs and a message passing protocol is what makes this project a relevant, valuable and novel contribution.

Deadline Planning



References

[1] G. P. F. Z. L. Ballotta, “A Reinforcement Learning Approach to Sensing Design in Resource-Constrained Wireless Networked Control Systems,” IEEE, 2022.

[2] J. O. Jae Won Lee, “A Multi-agent Q-learning Framework for Optimizing Stock Trading Systems,” Springer, 2002.

[3] W. W. Haotian Liu, “Federated Reinforcement Learning for Decentralized Voltage Control in Distribution Networks,” IEEE, 2022.

[4] Z. Y. T. B. Kaiqing Zhang, “Multi-Agent Reinforcement Learning: A Selective Overview of Theories and Algorithms,” Springer, 2021.

[5] S. Z. N. I. Daniel S. Bernstein, “The Complexity of Decentralized Control of Markov Decision Processes,” 2000.

[6] U. M. N. E. L. Justin Lidard, “Provably Efficient Multi-Agent Reinforcement Learning with Fully Decentralized Communication,” IEEE, 2022.

[7] G. A. S. Y. Bora Yongacoglu, “Decentralized Learning for Optimality in Stochastic Dynamic Teams and Games With Local Control and Global State Information,” IEEE, 2022.