Literature Review - Informal

Aamal Hussain

November 28, 2019

Contents

Introduction

The areas of interest regarding Multi Agent Systems fall into two distinct categories (which have some slight overlap). The first is denoted as 'Decision Making', though the terms 'Planning' and 'Distributed Control' may sometimes be used as substitutes. It is important to note that, in my case, I do not include Planning Formalisation techniques such as PDDL. Instead, I focus on the particular methods which involve interaction in the world. The second, I will refer to as 'Multi Agent Dynamics', though sometimes I will refer to this as 'Stability Analysis'.

Scope

Both of the aforementioned topics have been applied to all variants of multi agent systems and have sufficient room for further exploration. The typical variants of multi agent systems that I will consider are

- Swarms
- Multi Agent Reinforcement Learning (MARL)
- Decentralised Partially Observed Markov Decision Processes (Dec-POMDP)
- Game Theoretic Approaches
- Hard Coded

In the above, 'Game Theoretic Approaches' cover a wide spectrum, including zero-sum (minimax) games, bayesian games, and many more. The final category, 'Hard Coded', refers to any approach which does not fit neatly into any of these categories. The name is chosen since they often refer to methods which revolve around a series of if-else statements.

I will first consider Decision Making as this is most closely related to the research proposal, before I consider Multi Agent Dynamics, which has strong implications for Safe and Trusted Multi Agent Systems.

Objectives

The aim of the following sections is not to provide an exhaustive list of all work done in the aforementioned areas. To attempt to do this would be an exercise in futility. Instead, it is to identify research directions which lie within the broad scope of Multi-Agent Systems (MAS). With these directions, we may be able to narrow down the remaining literature review and hone in on particular problems and, in fact, we may find that we address multiple of these over the next four years (while we, of course add more). As such, the end of each chapter will provide a list of research directions which I have identified from the preceding review.

Chapter 1

Decision Making

Decision making refers to the generalised problem of considering how a multi-agent system should interact in the world to achieve their goals. This subsumes both the cases where the agents' goals are aligned (cooperative) or in conflict with one another (competitive).

The following sections will provide an overview of the literature aimed towards coordinating such systems as followed by interesting sub-fields in each approach which present avenues for research related to Safe and Trusted AI (STAI). These, of course, are not exhaustive and more will be added as the review progresses.

1.1 Swarms

Swarm based systems comprise of multiple homogeneous agents which are able to organise themselves through a formation using a series of simple local interactions with their neighbours [1]. Whilst the individual agents are generally simplistic, the collective behaviour may exhibit complex phenomena emulating systems observed in biological organisations such as bee or ant colonies [2]. Hybrid algorithms such as in [3] show an accelerated performance in reaching globally optimal solutions in search-based tasks. The advantage of many swarm algorithms is that they are based on local interactions and so are incredibly scalable [4].

The reduction of the complexity in interactions between agents also allows for the robots to perform other calculations on board. In [5], Pini et al. leverage this by considering adaptive task partitioning across swarms. This allows a swarm, in a decentralised manner, to deliberate whether to partition a task into its sub-tasks or to perform the task in its whole. As of now (to the best of my knowledge) the problem of partitioning general tasks into its N sub-tasks is unexplored. This, however, highlights another advantage of swarm systems; they are readily divided into sub-groups (as in [6]) to perform a divide-et-impera (divide and conquer) approach to solving problems [5]).

Furthermore, swarm systems may be designed in a leaderless manner and so do not require the use of a central controller [1]. This presents the advantage that the system can rapidly adapt to the loss of agents or separation of groups throughout the task. However, the assumptions made regarding the homogeneity of individual agents and the simplicity of their local interactions result in significant limitations placed on the complexity of the tasks that swarm systems can accomplish.

1.1.1 Co-evolution and Self-Healing

Recently, there has been an increased interest in introducing heterogeneity into the swarm systems to improve their real world applicability. An example of this which have been shown strong real world success can be found in [7]. Here, the authors consider two swarm teams, referred to as 'foot bots' and 'eye bots' who work in unison to explore an environment and solve a navigation task.

This area of research is sparsely populated and warrants further exploration. This is since the use of co-evolutionary teams can improve the robustness of the swarm optimisation. This is since it will be possible for a team to automatically determine when robots in the other team are not exhibiting expected behaviour and ensure that the other team self corrects. This process is referred to as 'Self-Healing' in [8]. Here, the authors allow a user to define the goals of a swarm system. From this, a 'trust' metric can be defined which measures the deviation of each agent from the expected behaviour. The self-healing process occurs by limiting communication of all agents

with 'untrusted' agents and encouraging communication with 'trusted' agents. The unison of self-healing with co-evolution may present the opportunity for heterogenous swarms to maintain their evolutionary stability, even in the face of environmental disturbances.

1.1.2 Fault Detection

The above sections have considered the fact that swarm systems are robust to losses in the group. However, any MAS system must first be able to recognise that an agent has undergone some failure.

To this end, Tarapore et al. [9] develop a robust fault detection approach in which the swarm itself, in a decentralised manner, is capable of assessing deviation from 'normal' behaviour, even when the behaviour of the swarm itself is altered (perhaps by a remote operator). The authors achieve this by requiring that the agents themselves sense and characterise their own behaviour. This characterisation is formulated as a binary feature vector which is then communicated to the agent's neighbours. These neighbours will reach a consensus over whether the agent should be treated as faulty based on their collective behaviour. The results presented in [9] show extremely promising results and suggest that their method is, in fact, able to determine faults with high accuracy in the presence of various fault types (including sensor and actuator failures), although poor performance is seen in actuation failures in some instances. It should be noted that this method requires that each robot transmit their feature vector to the nearest neighbours. In environments where communication may be severely limited, this may present further errors. Furthermore, it is unlikely that, when a robot is damaged, only one of its components will be affected. Therefore, it is important to determine the effect on performance in the face of multiple agent failures and in communication losses. This exploration may open the possibility of improving the state of the art in terms of failure detection in swarm systems. (Of course, this conclusion is based off two papers so further review is required).

1.1.3 Verification

Both of Alessio's papers [10, 11] fit in here but they require further reading

1.1.4 Directions

From the above consideration of swarms, the following research directions have been identified which, in my view, concern themselves with STAI.

- Healing through co-evolution: The application of heterogenous robot swarms towards ensuring that emergent phenomenon and swarm behaviour are as expected by the user.
- Fault Detection in limited communication: Considering the ability of a swarm to, in a decentralised approach, consider which robots in the team have failed, even in cases of no communication or multiple failures.

1.2 Dec-POMDPs

The use of POMDPs in multi-agent settings is formalised as decentralised POMDP (Dec-POMDP) which aims for a team of agents to maximise a common utility. However, it has been found that determining the exact solution to Dec-POMDP problems is NEXP [12] and so is intractable for all but toy problems. A number of methods have been presented to attempt to solve Dec-POMDPs. Oliehoek gives a review of these in [13]. Most solutions (such as brute force) are intractable for all but toy problems.

Approximate solutions to Dec-POMDP have been proposed, perhaps most notable of which is the proposal of MacDec-POMDP [14] by Amato et al. Here, macro actions (actions which extend over multiple time steps) are used, as opposed to low-level actions which are re-evaluated at each time step. This allows an exact solution to be found as it does not need to be evaluated at each time step. This method assumes that, once macro-actions are distributed, the policies (sequence of state-action pairs) are known. Since this is not the case, Amato also proposes the use of a Dec-POSMDP [15], where 'SMDP' refers to 'Semi-Markov Decision Process, in which a high level model is defined without the underlying Dec-POMDP's actions and observations.

However, this is largely applicable in passive settings where common payoffs can be determined by an offline planner. They also require a significant amount of data with which to allow the system to learn the underlying models and payoff structures. This limits the applicability of the system when communication is limited and the system is presented with environments that it has not seen before. Recent work in MDPs [16] has considered learning in the face of Significant Rare Events (SREs) which the system has not yet observed. Currently, it is required that a model of such SREs is known and so it would be interesting to consider the application of Dec-POMDPs in situations where the SRE model is incomplete or erroneous and assess the robustness of the Dec-POMDP framework against such events.

1.2.1 Directions

The area of Dec-POMDPs still requires more review from me, especially for solutions which are not developed by Amato as he seems to largely dominate the field. Based on this initial review of the area, a potential direction for research is

- Significant Rare Events: Consider Dec-POMDP capability to remain robust to SREs which have a limited or erroneous model.
- Agent Failures: The Dec-POMDP model uses a centralised planner which acts offline. It therefore assumes that the agents will be able to carry out their assigned tasks. It would be interesting to examine the possibility of, either adaptive planning, or planning with contingencies in the Dec-POMDP framework.

1.3 Game Theoretic Approaches

Game theoretic models are generally the go-to method for understanding multi-agent systems. As such they fit into all of the categories in this chapter (except, perhaps, swarms) since Dec-POMDP and MARL methods have both used game theory to support their frameworks. In fact, Dec-POMDP is a subset of Partially Observed Stochastic Games (POSG), in which all agents use the same payoff. Game theory can, therefore, be used in an applied capacity to direct task allocation across heterogenous teams.

1.3.1 Market Based Methods

Garapati et al. [17] define a market based method as the setting where agents "follow their own interests and establish the mechanism of a market for distributing the tasks". Auctioning is the most widely used sub-field of market approaches and so I will use them interchangeably.

Whilst there are different variants to autioning, the general procedure is that an auctioneer who has knowledge of a task (or multiple tasks) will set up an auction for said task. Agents can then make bids on these tasks and, once the auction is complete, the highest bid will win the task. In the specific application to robotics, a robot's bid will often reflect the costs, suitability or utility their undertaking the task [18]. This immediately highlights a few points. The first is that the method is not too heavily reliant upon a single processor to determine some joint policy. Tasks are allocated on a case-by-case basis and the utilities are calculated by the agent themselves. The only centralised process is the auctioneer's assessment of the winner which then relays this information back to them. The downside of this is that the system is heavily reliant upon strong communication channels, without which tasks may not be assigned, incorrect utilities may be communicated and, in general, sub-optimal solutions reached. Furthermore, the requirement that the agents themselves determine the cost of their actions assumes that they have the computational capability to do so. Furthermore, the bids placed by each agent need to be a strong representation of their capability to perform a task which may be hard to estimate without expert knowledge. However, market based methods are well suited to explanation through argumentation (similar to [19].

With well chosen payoffs, market based approaches work extremely well. For instance, in [20], the authors show that a free market approach (where agents try to maximise their own profits) can lead to a strong collaborative effort across teams. Similarly, in [21], Thomas et al. apply the auctioning scheme presented in [22] towards a robot construction team. However, it is important to note that these are both passive settings; tasks were assigned before the team were in the field and, in the case of [22], the system would repeat the bidding process if a robot failed. While both show strong performance, it cannot be said that either would be applicable in dangerous environments

in which dynamic reassignment must happen within strict time constraints. Stancliff et al. [23] suggest that a more robust method to planning would be to account for failures a priori, a philosophy which is exemplified in [24] who consider the robot's reliability and relevance to a task as well as 'history relevance' which considers the relationship between pairs of robots with the aim of producing more effective teams.

There has also been some interesting work in probabilistic verification of market based approaches. Most notable to me is [25] which considers the case of conflict avoidance. Though their method focuses on collision avoidance, it highlights the need for verification of conflict resolution and goal achievement in market based approaches with different payoff structures. Sirigineedi et al. [26] make a step in this direction by considering the verification of cooperative surveillance along a route network. From my understanding, this means that they were able to verify that their agents were able to traverse along the network without interference. However, this, as always, requires further analysis to truly understand.

1.3.2 Directions

The particular considerations which have jumped out to me from the above analysis are as follows:

- Verification of goal achievement under different payoffs: can we ensure that self-interested agents will, in fact, show cooperative emergent properties? A similar question arises in terms of valuations (how much cost each agent incurs).
- A priori consideration of reliability: Can we learn and take into account the fact that robots may fail throughout the progress of a mission when we assign tasks?

Stochastic Games

1.4 Hard Coded

1.5 MARL

Reinforcement learning extends the Markov Decision Process problem by considering the case where the payoff model is not known. This, of course, is the case for most real world environments. As such, MARL algorithms can perhaps be considered to be more applicable than Dec-POMDP models. Fortunately, MARL has picked up a lot of traction in research recently, with a large body dedicated towards solving the many problems it presents.

The largest problem in MARL is the non-stationarity of the environment [27]. In single-agent settings, it is assumed that the environment is Markovian. However, this must be lifted in the Multi Agent setting since other agents in the environment will be learning concurrently. This learning will be based on their own history of interactions which extend beyond the previous state. As such, we must now consider that the policy for any one agent will depend on the policy of all other agents. As such, a big concern in this area is regarding convergence guarantees and the stability of the learner system. Approaches to this will be discussed in the next chapter.

Agent Modelling

Returning to the problem of non-stationarity, solutions have been presented in which the agent models the learning of other agents. A noteworthy example of this is found in [28] in which the agent performs a one-step lookahead of the other agents' learning and optimises with respect to this expected return. They show that this leads to stable learning and can even lead to emergent cooperation from competition. However, the method requires that both agents have exact knowledge of the others' value functions in order to perform the one step lookahead. Furthermore, it has only been considered for the case of a two agent adversarial game and so the scalability of the system to multiple agents is not yet understood. Another method presented by Mao et al. [29] uses a centralised critic to collect the actions and observations of all agents and allows it to model the joint policy of teammates. This is shown to generate cooperative behaviour across four agents and so is more applicable to real world settings. However, its disadvantage over the method presented in [28] is that the critic is centralised. In real world settings, this requires the presence of an agent (perhaps a laptop) which is able to handle the computational

load of determining a joint policy across all agents and must then communicate the Q-values of all agents back to them. This is both a taxing both in terms of computation and time.

Hong et al [30] present a similar system for modelling teammate policies by tasking a CNN with determining the policy features of other agents and then embedding these as features in its own DQN. This shows strong performance in settings where other agents dynamically change their policies. The concerns with this, however, are that, as the number of agents in the field increase, the CNN in each agent must perform another approximation. This places strong requirements on the performance of the CNN since errors in estimation will accumulate as the number of agents increases. Similarly, the complexity of the DQN will increase as more feature vectors are added.

Finally, all of the above methods are not robust to evolving numbers of agents. The problem of agent modelling is an important one to ensure stable learning and to understand the evolution of the system. It also presents a strong challenge and is open to exploration. To put it in context the methods described in this section are all from 2018-19, so its all very new.

1.5.1 Directions

I still have a lot of reading to do regarding MARL, which, in turn, will identify new directions. However, on initial assessment I put forward

- Modelling evolving teammates: The purpose of this is to more strictly ensure the stability of the learning process. However, the particular problem I suggest is to consider the modelling in a decentralised manner and with the consideration of evolving numbers of agents in teams.

1.6 MARL Approaches

The task of MARL is to determine an optimal joint policy for all agents across the game. This joint policy may be the concatenation of all the individual policy or it may just be options for each agent to take. In either case, optimality is defined through the standard notions of Nash equilibria and so, in this section, I will try to consider the broad spectrum of methods which attempt to achieve this Nash equilibria. I will start with the more foundational methods which are, thankfully, all reviewed by Schwartz in [31]. I will use the chapters of the book to help guide this review. I am not referencing the individual papers immediately, though the references can be found in the book.

Learning in Two Player Matrix Games

The most fundamental method to learning in Matrix games is the simplex algorithm. This is a popular method of linear programming (in which constraints are linear). This will be important in considering more current methods. A similar consideration is given to the infinitesimal gradient ascent algorithm, in which the step size converges to zero. This method guarantees that, in the infite horizon limit, the payoffs will converge to the Nash equilibrium payoff. Note that this does not necessarily mean that both agents will converge to a single Nash equilibrium. This is a particular problem in games where there are multiple Nash equilibria. However, in practice it is difficult to choose a convergence rate of the step size and, without an appropriate choice the strategy may oscillate as shown in the book. To address this, a modified approach is presented by Bowling and Veloso which incorporates the notion of Win or Learn Fast (WoLF) to produce WoLF-IGA. WoLF is a notion we will come across often in MARL and is shown by the authors to converge to always to a NE. The concern with WoLF methods, however, is that it requires explicit knowledge of the payoff matrix (which is not so much of a problem for model based methods) and the opponent's strategy (which is more of a problem in real-world methods). Finally, the Policy Hill Climbing method (PHC) is shown to converge to an optimal mixed strategy if the other agents are stationary (i.e. are not learning). However, it is shown that, when this is not the case, the algorithm again oscillates. The WoLF-PHC adaptation of this method is shown to converge to a NE strategy for both players with minimal oscillation.

The above methods are all centralised techniques, in which a controller determines the optimal joint policy for both agents. However, in real scenarios it is often preferable that each agent learns their own strategy, a task which must be completed without information of the other agent's strategy. The methods presented towards this problem are: linear reward-inaction (L_{R-I}) which guarantees convergence to NEs in games which contain pure NEs, linear reward-penalty (L_{R-P}) which can guarantee convergence to mixed strategies given the appropriate

parameters, lagging anchor algorithm which also converges to mixed strategies, and the author's own proposal of the L_{R-I} lagging anchor algorithm which can converge to both pure and mixed NEs.

Learning in Multiplayer Stochastic Games

Stochastic Games (or Markov Games) form a basis for MARL settings. However, in this case the agents must learn about the equilibrium strategies by playing the game, which means they do not have a priori knowledge of the reward or transition functions. Schwarz considers two properties which should be used for evaluating MARL algorithm: rationality and convergence. The latter simply states that the method should converge to some equilibrium whereas the former suggests that the method should learn the best response to stationary opponents. A similar set of conditions is considered by Conitzer and Sandholm in [32], whose algorithm we will consider shortly. Schwarz then presents a review of MARL methods (as of September 2014)

Dec-POMDP and MARL

As Dec-POMDPs are the theoretical formulation of MARL problems, it stands to reason that other methods for solving Dec-POMDPs should provide insights into improving MARL. Fortunately Oliehock in [13] presents a number of existing methods towards solving Dec-POMDPs. The methods which I feel may be applicable are

- Alternating Maximisation: This is effectively coordinate ascent for determining a joint policy.
- Approximation as Bayesian Games: Perhaps solving the repeated Bayesian Game through MARL would be more efficient. It does, however, force us to ask how best to sub-divide a Dec-POMDP into a series of Bayesian Games
- Selecting sub-tree policies: Could DL be used to determine which sub-tree policies are optimal? In order to make this work, we would also need to consider how to select the feature space of sub-trees and how to collect this information.

Chapter 2

Multi Agent Control

2.1 Stochastic and Robust Model Predictive Control

Francesco: I've moved this to the top for your attention - it seems to be the most interesting problem with the scope to improve the theoretical results as well as applications in complex domains which are of significant interest these days.

The idea of Stochastic Control is best described in the research statement of the Control and Analysis of Stochastic Systems (CASS) group at Penn State:

> "...to model, predict and control uncertain engineering systems where the interplay between dynamics and uncertainty (stochasticity) is important."

When operating in the real world, systems are subject to uncertainty and environment perturbations. Despite this, for their safe operation, they must provide certain guarantees (such as collision avoidance) and remain stable. It is with this in mind that we must design control mechanisms for the system which satisfy these requirements, even in the face of uncertainty. Methods in stochastic control have found their way into controlling power distribution, chemical processes and (of course) robotics.

Model Predictive Control is a long standing paradigm in AI which looks specifically at the problem of operating real-world agents safely in the face of environmental disturbances. The overarching idea begins with the assumpion that we have a model of the environment. As an example, in the case of autonomous vehicles, we have a model of how adjusting the angle of the front wheels will affect the heading of the car. We then perform a finite horizon look ahead, in which we estimate the environment state for a few time steps ahead, and generate a policy for this horizon. The agent performs the immediate action generated by the policy, and then we repeat the process, after taking measurements of the environment to determine the error in the system. Throughout this process, the controller (a.k.a. policy) must remain stable, but also satisfy constraints. From our previous example, a likely constraint would be that the controller will never result in the car going on the pavement, where it would present a real hazard to pedestrians. It is through these constraints, and the requirement of stability that the system is required to remain safe throughout operation. To that end, mathematical proofs of these properties are provided in the literature ensuring that we can trust in the system's performance.

Deterministic MPC assumes that the environment is completely deterministic and, therefore, assumes knowledge of the exact nature of the disturbances. This somewhat naive assumption simplifies computation and so appears often in the literature, as in [33]. However, it does not provide strong guarantees outside of games and simulations. Robust MPC brings this to a higher level of abstraction in which system perturbations, though deterministic, lie on a bounded set. Therefore, we no longer assume the exact nature of the environment and can guard against worst case scenarios. This is extremely effective in closed environments or for simple tasks, but falters in more complex environments.

Stochastic Model Predictive Control (SMPC) lifts the assumption made in MPC, namely that perturbations are deterministic and lie on a bounded set [34]. To accomplish this, we define chance constraints, for which probabilistic guarantees must be determined. Furthermore, optimality is defined in terms of the minimisation of the expectation of a probabilistic cost function. This formulation brings about controllers which are more applicable and robust when deployed. However, the field is in its infancy and presents a number of theoretical challenges. These are best described by Mesbah in [34]

- The arbitrary form of the feedback control laws
- The non-convexity and intractability of the chance constraints
- The complexity of the uncertainty propagations
- Establishing stability of the control problem

Since the review was written in 2016, there have been a number of approaches towards solving some of these problems, most notably the intractability of the chance contraints. For instance, in [35], Paulson and Mesbah propose the use of joint chance constraints in considering time varying stochastic disturbances as well as model uncertainty. This is shown to be strongly suited to non-linear systems, which itself is an open problem in the MPC sphere.

2.1.1 Directions

The problem of MPC, both in the stochastic, and the robust sense, are rife for work in theoretical and applied problems. Control in autonomous systems is also one of the more applicable problems related to STAI since it requires that we place mathematical guarantees on the performance and stability of the system as well as guarantees on the satisfaction of constraints. This is, of course, an ever present problem in autonomous systems which operate in the real world. I begin by listing the more theoretical areas for exploration in MPC before then considering their application domains.

- Significant exploration is required with regards to the optimisation problem. Especially for non-linear (or coupled systems), solving the optimisation problem is a struggle. As such, approximate methods (such as sampling) should be explored and their effect on the system guarantees presented.
- Uncertainty propagation in SMPC is also an important problem since we must be able to determine the uncertainty in the environment model in order to generate control laws. We should consider a comparison of formal techniques, such as Bayesian inference (e.g. Markov models) or particle filters, against modern deep learning methods for propagation.
- Adaptive MPC is particularly important when considering failures in a system. In his talk, Mark Cannon considers the platooning problem (in which multiple autonomous vehicles move together whilst maintaining a specified distance from one another). Consider, then, the case where one vehicle undergoes a fault (burst tyre etc). The system model must incorporate this, and hence requires adaptation. It would again be interesting to consider the use of neural networks in this manner. However, specifically we should consider whether the use of a function approximator in the form of an NN might result in a violation of the constraints.
- Distributed predictive control. This is what, in my view, we could work towards. This is a difficult problem due to the coupled nature of the system dynamics. We will need to consider: the effect that each agent has on the other, the effect of the environment on all agents, estimating the future behaviour of each agent, and ensuring that all agents do not violate the constraints. It is here that I believe game-theoretic notions (especially differential games) will come to our aid. Similarly, mean-field games should be considered in the general N-agent problem, as well as in the case of multiple disturbances, but I will elaborate on this in the applied problems section.

Multi Agent Dynamics

Multi Agent Dynamics considers the problem of mathematically modelling learning in Multi Agent Systems (MAS). This model then serves to be able to predict the evolution of a learning system as well as to understand the trajectory of learning. Typically, this looks at considering whether or not the method is likely to converge towards a Nash equilibrium. This is generally a difficult problem to solve [36] for all but toy problems. To extend this applicability into real world settings requires the study of stable equilbrium points; [37] shows that the stable equilbrium and Nash equilibrium (NE) are not necessarily the same and, in fact, argue that stable points are more

informative than NEs. Stability provides some guarantees against the stochastic nature of the environment since a stable equilibrium will always be returned to even after perturbations. This extremely important in Safe and Trusted AI as it provides guarantees against undesired behaviour in real world environments.

2.2 Stability of Learning Agents

Stability may be looked at from the view point of two perspectives. The first is from an optimisation point of view. This considers the dynamics of the learning model, allowing us to better choose our parameters and design our models so that they may converge to a stable result. The second is from the view point of the state-action space of a learnt model. This allows us to determine, before the MAS is deployed, which set of state-action pairs will lead to unstable behaviour. This knowledge allows us to consider which state-action pairs should be avoided. In both cases, stability analysis allows us to build multi agent systems which will learn and act in the way that we expect them to.

In [37], Letcher et al. model gradient descent learning of generative-adversarial-networks (GANs) as a two-player differentiable game. A differentiable game is one in which the loss function is twice differentiable. Using this formulation, they are able to analyse the system from new perspectives by considering the current state-of-the-art understanding of differentiable game theory. Whilst, at first glance, this may seem like a purely theoretical exercise, they go on to show that the insights gained allow them to develop a new multi-objective optimisation technique for GANs which shows stronger convergence properties, most notably of which is that it guarantees that the method finds a stable equilibrium (and avoids saddles) between the two players' loss functions.

Jin and Lavaei [38] consider the policy of a reinforcement learning agent as a non-linear, time varying feedback controller. Using this notion, they then consider the bounded-input-bounded-output stability of the system. They do this by analysing the ratio between the total output and total input energy (called the L2 gain). If the L2 gain remains finite then the system may be considered to be stable. They then apply these considerations on real-world applications including multi-agent flight formation and obtain stability certificates (essentially confirming that the system will remain stable under certain conditions) for the learned controller.

Berkenkamp et al. [39] consider a similar problem from a different definition of stability. Specifically, they look at stability from the point of view of Lyapunov functions. A system is said to be stable if the applying the policy will result in stricly lower evaluations of the function. In other words, a system is stable if its corresponding Lyapunov function is decreasing towards a minimum point. The authors use this idea to define a 'region of attraction' in which the system is stable in the sense of Lyapunov. The goal of Safe Lyapunov Learning, a method which they develop from these insights, is to learn a policy without leaving this region of attraction. They do this by taking measurements within the set and using this to learn about the system dynamics, thereby increasing the safe set. With this, we now have a guarantee that policy optimisation will not result in unsafe behaviour, even in the presence of stochasticity and exploration.

As we can see from the discussion thus far, there are many definitions and perpectives of stability, each lending to a new understanding of learning systems. To add another to the mix, Milchtaich in [40], presents a notion of static stability. This means that it is based solely on the incentives of the players and does not require a consideration of the dynamics of the system. Milchtaich's definition of a stable system is one in which, when perturbed, it is more beneficial for an agent to move back towards the equilibrium than it is for them to move away. This is perhaps the most fundamental definition of a stable equilibrium and, as such, Milchtaich shows that it is applicable to all strategic games, within certain assumptions (finite set of player and continuous strategy space) and considers probabilistic perturbations from the original state. This is particular applicable to Multi Robot Reinforcement Learning (or any MARL in continuous strategy spaces) since these systems require random exploration of the strategy space and the dynamics are not known a priori. However, the lack of dynamics means that we do not consider the evolution of learning.

2.3 Learning Stable Agents

The above discussion has considered how notions of stability and control are important to Multi-Agent Learning. It stands to reason, therefore, that Multi-Agent Learning and, more specifically, game theory, can be used to develop controllers which drive systems to stable states. This idea is explored in [41]. Here, a zero-sum game

is considered in which the players are a controller and an adversarial environment. The design of the controller must be such that it is able to drive the system to zero error. To illustrate, consider the problem of designing a controller for a re-entry vehicle, as in [42], in which vortices seek to destabilise the agent. This will allow us to build stable agents in a much more efficient manner since we can simulate the adversarial environment and hypothetical scenarios the agent may encounter without actually encountering them. The same notion is explored by Bardi et al [43].

Mylvaganam et al, in [44], consider the N-robot collision avoidance problem, similarly from the point of view of differential game theory. They develop a robust feedback system for the robots which they show to be able to drive the system towards predefined targets whilst providing guarantees of interference from other agents (or lack thereof). In [45], Mylvaganam also considers a game theoretic control of multi-agent systems in a distributed manner. Here, agents only consider their own payoff structure and have limited communication with one another. The author shows that the an approximate equilibrium can be found using algebraic methods and illustrate the capabilities of the technique using a collision avoidance example. For the sake of brevity, I will not include all of the numerous strides that Mylvaganam has introduced to the area. However, I must conclude with those presented in [46]. Here, the author presents approximate solutions to a number of differential games, including linear-quadratic differential games (in which system dynamics are linear functions whilst payoff functions are quadratic), Stackelberg differential games, where a hierarchy is induced across the players (a notion was suggested in the research proposal) and mean-field games, which is discussed in Section 1.6. The importance of the linear-quadratic differential game is the stability of the solution; solutions for the Nash equilibria are admissable iff they are locally exponentially stable (which the author often shows with the aid of Lyapunov functions). Approximate solutions to the NE are developed which are more feasible to calculate online. The author then shows that this is not simply a theoretical exercise by applying the novel methods towards multi-agent collision problems and designs dynamic control laws which guarantee that each agent will reach their desired state whilst avoiding collision with the other agents. Similarly, the Stackelberg game is applied to the problem of optimal monitoring by a multi robot system.

2.3.1 Directions

I will briefly summarise the research directions which I develop in this section, to save ourselves some time, and to preserve sanity. **I'll do this later. Sanity is overrated.**

The above discussion has illustrated a few points. The first, and perhaps most important, is that stability is extremely applicable to Stochastic Optimal Control. This is because of the fact that learning agents, especially in real world environments, are subject to uncertainty and perturbations as well as epsilon-random exploration. As such, it is important that the system is able to return to a stable state after such perturbations. This leads to the need to develop learning systems as in [37] which are attracted to stable equilibria. Similarly, it allows us to consider existing algorithms and whether they lead to stable equilibria. For instance, Letcher et al show in [47] that the Learning with Opponent Learning Awareness (LOLA) algorithm [28] does not converge to stable fixed points.

The second point is that the notion of stability has multiple definitions. Each one is appropriate in different conditions and, as such, it is important to explore stability from various viewpoints to consider how to best understand the evolution of a learning system.

Another factor to consider is suggested in [40]; multiplayer games do not necessarily contain stable fixed points. The existence of stable equilibria is therefore an open question and is well worth consideration. This would help guide the choice of loss functions and strategy spaces. This is likely to be an extremely difficult problem to generalise but would have a dramatic impact on the safety certification of MARL systems.

The problem of finding stable solutions through multi agent dynamics also opens the door for a large degree of application domains which are currently quite sparse. I would recommend looking at the contents page of [48] which illustrates a number of applications of dynamic and differential game theory in a number of domains including: pursuit-evasion, systems control and economic modelling. My particular interest, as always, lies within robotic control problems. This appears to be the most direct and relevant application of stable MAS systems. An example of this might be the problem of Cooperative Moving Path Following Control considered by [49]. Here, multiple agents must track moving targets according to a pre-defined path without interfering with one another. It is clear then that each player's control is dependent on the other and so can be modelled as a game theoretic

problem. Our task, then, is to determine control rules which lead to stable coordination across the team. Similarly, we can model consider modelling a controller and an adversarial environment as a zero-sum game as examined in [41].

Another application is proposed by Letcher et al in [37] - that of Generative adversarial networks (GANs) which can be considered as a two player minimax game with differentiable loss functions. As pointed out by Mylvaganam, it may also be interesting to consider the differential game problem from the point of view of different information structures. In real-world systems, it is unlikely that all agents will have access to the information of other agent' cost functions and current positions. It could, however, be interesting to explore the use of localisation methods to determine and track this information through observation (e.g. through Kalman or Bayesian methods). Finally, most of the relevant work in stochastic optimal control considers the case where system dynamics are linear. A whole new set of problems arise in the more realistic case where we must consider the non-linearity of system dynamics. Such a consideration would also allow us to extend these technique to the more difficult (and important) problems in control.

The following papers [50, 51, 52, 53, 39, 38, 37] are the ones that I found particularly relevant to this study. However, they will require some further analysis before I write about them here.

Bibliography

[1] M. S. Couceiro, R. P. Rocha, and F. M. L. Martins, "Towards a predictive model of an evolutionary swarm robotics algorithm," in *2015 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, 5 2015, pp. 2090–2096. [Online]. Available: http://ieeexplore.ieee.org/document/7257142/

[2] D. Sethi and A. Singhal, "Comparative analysis of a recommender system based on ant colony optimization and artificial bee colony optimization algorithms," in *8th International Conference on Computing, Communications and Networking Technologies, ICCCNT 2017*. Institute of Electrical and Electronics Engineers Inc., 12 2017.

[3] Y. Gao, "An improved hybrid group intelligent algorithm based on artificial bee colony and particle swarm optimization," in *Proceedings - 2018 International Conference on Virtual Reality and Intelligent Systems, ICVRIS 2018*. Institute of Electrical and Electronics Engineers Inc., 11 2018, pp. 160–163.

[4] Y. Rizk, M. Awad, and E. W. Tunstel, "Decision Making in Multiagent Systems: A Survey," pp. 514–529, 9 2018.

[5] G. Pini, A. Brutschy, M. Frison, A. Roli, M. Dorigo, and M. Birattari, "Task partitioning in swarms of robots: An adaptive method for strategy selection," *Swarm Intelligence*, vol. 5, no. 3-4, pp. 283–304, 2011.

[6] P. Zahadat and T. Schmickl, "Division of labor in a swarm of autonomous underwater robots by improved partitioning social inhibition," *Adaptive Behavior*, vol. 24, no. 2, pp. 87–101, 2016.

[7] F. Ducatelle, G. A. D. Caro, C. Pinciroli, Luca, M. Gambardella, F. Ducatelle, . Dalle, G. A. D. Caro, C. Pinciroli, and L. M. Gambardella, "Self-organized Cooperation between Robotic Swarms," Tech. Rep. [Online]. Available: http://www.swarmanoid.org

[8] R. Liu, F. Jia, W. Luo, M. Chandarana, C. Nam, M. Lewis, and K. Sycara, "Trust-Aware Behavior Reflection for Robot Swarm Self-Healing *," Tech. Rep. [Online]. Available: www.ifaamas.org

[9] D. Tarapore, J. Timmis, and A. L. Christensen, "Fault Detection in a Swarm of Physical Robots Based on Behavioral Outlier Detection," *IEEE Transactions on Robotics*, pp. 1–7, 8 2019.

[10] P. Kouvaros, A. Lomuscio, E. Pirovano, and H. Punchihewa, "Formal Verification of Open Multi-Agent Systems," Tech. Rep., 2019. [Online]. Available: www.ifaamas.org

[11] A. Lomuscio and E. Pirovano, "A Counter Abstraction Tech-nique for the Verification of Probabilistic Swarm Systems," Tech. Rep., 2019. [Online]. Available: www.ifaamas.org

[12] B. Eker, E. Ozkucur, C. Mericli, T. Mericli, and H. L. Akin, "A finite horizon DEC-POMDP approach to multi-robot task learning," in *2011 5th International Conference on Application of Information and Communication Technologies (AICT)*. IEEE, 10 2011, pp. 1–5. [Online]. Available: http://ieeexplore.ieee.org/document/6111001/

[13] F. A. Oliehoek, "Decentralized POMDPs," Tech. Rep.

[14] C. Amato, G. Konidaris, G. Cruz, C. A. Maynor, J. P. How, and L. P. Kaelbling, "Planning for decentralized control of multiple robots under uncertainty," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 5 2015, pp. 1241–1248. [Online]. Available: http://ieeexplore.ieee.org/document/7139350/

[15] C. Amato, "Decision-Making Under Uncertainty in Multi-Agent and Multi-Robot Systems: Planning and Learning," Tech. Rep., 2017. [Online]. Available: https://youtu.be/34xHxXrnPHw,

[16] R. Klima, D. Bloembergen, M. Kaisers, and K. Tuyls, "Robust Temporal Difference Learning for Critical Domains," Tech. Rep., 2019. [Online]. Available: www.ifaamas.org

[17] K. Garapati, J. J. Roldán, M. Garzón, J. del Cerro, and A. Barrientos, "A game of drones: Game theoretic approaches for multi-robot task allocation in security missions," in *Advances in Intelligent Systems and Computing*. Springer Verlag, 2018, vol. 693, pp. 855–866.

[18] M. Bernardine Dias, R. Zlot, N. Kalra, and A. Stentz, "Market-based multirobot coordination: A survey and analysis," pp. 1257–1270, 7 2006.

[19] H. Jung, "Distributed constraint satisfaction as a computational model of negotiation via argumentation," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 2239. Springer Verlag, 2001, p. 767.

[20] M. B. Dias and A. Stentz, "A Free Market Architecture for Distributed Control of a Multirobot System," 1 2000.

[21] G. Thomas, A. M. Howard, A. B. Williams, and A. Moore-Alston, "Multi-robot task allocation in lunar mission construction scenarios," in *Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics*, vol. 1, 2005, pp. 518–523.

[22] B. P. Gerkey and M. J. Matarić, "Sold!: Auction methods for multirobot coordination," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 758–768, 10 2002.

[23] S. B. Stancliff, J. Dolan, and A. Trebi-Ollennu, "Planning to fail - Reliability needs to be considered a priori in multirobot task allocation," in *Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics*, 2009, pp. 2362–2367.

[24] J. Chen, X. Yang, Y. Hu, Y. Han, D. Li, and G. Zhang, "A Multi-Robots Task Allocation Algorithm Based on Relevance and Ability With Group Collaboration," *Article in International Journal of Intelligent Engineering and Systems*, 2010. [Online]. Available: https://www.researchgate.net/publication/267968475

[25] L. Pallottino, A. Bicchi, and E. Frazzoli, "Probabilistic verification of decentralized multi-agent control strategies: A Case Study in Conflict Avoidance," in *Proceedings of the American Control Conference*, 2007, pp. 170–175.

[26] G. Sirigineedi, A. Tsourdos, B. A. White, and P. Silson, "Decentralised cooperative aerial surveillance for harbour security: A formal verification approach," in *2010 IEEE Globecom Workshops, GC'10*, 2010, pp. 1831–1835.

[27] P. Hernandez-Leal, B. Kartal, and M. E. Taylor, "A Survey and Critique of Multiagent Deep Reinforcement Learning $," Tech. Rep.

[28] J. Foerster, R. Y. Chen, O. Maruan Al-Shedivat, S. Whiteson, P. Abbeel, I. Mordatch OpenAI, and M. Al-Shedivat, "Learning with Opponent-Learning Awareness," Tech. Rep., 2018. [Online]. Available: www.ifaamas.org

[29] H. Mao, Z. Zhang, Z. Xiao, Z. Gong, and Z. . Gong, "Modelling the Dynamic Joint Policy of Teammates with Attention Multi-agent DDPG," Tech. Rep. [Online]. Available: www.ifaamas.org

[30] Z.-W. Hong, S.-Y. Su, T.-Y. Shann, Y.-H. Chang, and C.-Y. Lee, "A Deep Policy Inference Q-Network for Multi-Agent Systems," Tech. Rep., 2018. [Online]. Available: www.ifaamas.org

[31] H. M. Schwartz, *Multi-agent machine learning : a reinforcement approach.*

[32] V. Conitzer and T. Sandholm, "AWESOME: A General Multiagent Learning Algorithm that Converges in Self-Play and Learns a Best Response Against Stationary Opponents *," Tech. Rep.

[33] U. Rosolia and F. Borrelli, "Learning model predictive control for iterative tasks. A data-driven control framework," *IEEE Transactions on Automatic Control*, vol. 63, no. 7, pp. 1883–1896, jul 2018.

[34] A. Mesbah, "Stochastic model predictive control: An overview and perspectives for future research," pp. 30–44, dec 2016.

[35] J. A. Paulson and A. Mesbah, "An efficient method for stochastic optimal control with joint chance constraints for nonlinear systems," *International Journal of Robust and Nonlinear Control*, vol. 29, no. 15, pp. 5017–5037, oct 2019. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/rnc.3999

[36] Y. Shoham and K. Leyton-Brown, "Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations," Tech. Rep. [Online]. Available: http://www.masfoundations.org.

[37] A. Letcher, D. Balduzzi, S. Racanì, J. Martens, J. Foerster, K. Tuyls, and T. Graepel, "Differentiable Game Mechanics," Tech. Rep., 2019. [Online]. Available: https://github.com/deepmind/symplectic-gradient-adjustment.

[38] M. Jin and J. Lavaei, "Stability-certified reinforcement learning: A control-theoretic perspective," 2018. [Online]. Available: http://arxiv.org/abs/1810.11505

[39] F. Berkenkamp, M. Turchetta, A. P. Schoellig, and A. Krause, "Safe model-based reinforcement learning with stability guarantees," in *Advances in Neural Information Processing Systems*, vol. 2017-Decem, 2017, pp. 909–919.

[40] I. Milchtaich, "Static Stability in Games," Tech. Rep., 2007.

[41] J. R. Marden and J. S. Shamma, "Annual Review of Control, Robotics, and Autonomous Systems Game Theory and Control," 2018. [Online]. Available: https://doi.org/10.1146/annurev-control-060117-

[42] M. H. Breitner and H. J. Pesch, "Reentry Trajectory Optimization under Atmospheric Uncertainty as a Differential Game," in *Advances in Dynamic Games and Applications*. Birkhäuser Boston, 1994, pp. 70–86.

[43] M. Bardi and C. Sartori, "Differential games and totally risk-averse optimal control of systems with small disturbances," in *Lecture Notes in Control and Information Sciences*, vol. 156. Publ by Springer-Verlag Berlin, 1991, pp. 91–99.

[44] T. Mylvaganam and M. Sassano, "Autonomous collision avoidance for wheeled mobile robots using a differential game approach," *European Journal of Control*, vol. 40, pp. 53–61, 2017. [Online]. Available: https://doi.org/10.1016/j.ejcon.2017.11.005

[45] T. Mylvaganam, "A Game Theoretic Approach to Distributed Control of Homogeneous Multi-Agent Systems," Tech. Rep.

[46] ——, "APPROXIMATE FEEDBACK SOLUTIONS FOR DIFFERENTIAL GAMES THEORY AND APPLICATIONS," Tech. Rep., 2014.

[47] A. Letcher, J. Foerster, D. Balduzzi, T. Rocktäschel, and S. Whiteson, "STABLE OPPONENT SHAPING IN DIFFERENTIABLE GAMES," Tech. Rep.

[48] R. P. Hamalainen and H. K. H. K. Ehtamo, *Differential games : developments in modelling and computation : proceedings of the Fourth International Symposium on Differential Games and Applications, August 9-10, 1990, Helsinki University of Technology, Finland.* Springer-Verlag, 1991.

[49] M. F. Reis, R. P. Jain, A. P. Aguiar, and J. B. de Sousa, "Robust Cooperative Moving Path Following Control for Marine Robotic Vehicles," *Frontiers in Robotics and AI*, vol. 6, 11 2019. [Online]. Available: https://www.frontiersin.org/article/10.3389/frobt.2019.00121/full

[50] J. P. Bailey and G. Gidel, "Finite Regret and Cycles with Fixed Step-Size via Alternating Gradient Descent-Ascent," pp. 1–15, 2019.

[51] J. P. Bailey and G. Piliouras, "Multi-Agent Learning in Net-work Zero-Sum Games is a Hamiltonian System," Tech. Rep., 2019. [Online]. Available: www.ifaamas.org

[52] V. Boone and G. Piliouras, "From Darwin to Poincar\'e and von Neumann: Recurrence and Cycles in Evolutionary and Algorithmic Game Theory," 2019. [Online]. Available: http://arxiv.org/abs/1910.01334

[53] L. Dickens, K. Broda, and A. Russo, "The Dynamics of Multi-Agent Reinforcement Learning," Tech. Rep.