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 [?] 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 [?]. Most solutions (such as brute force) are intractable for all but toy problems. Solutions which have found some tractability are

- Alternating Maximisation: This is effectively coordinate ascent for determining a joint policy; the optimal policy for one agent is found by keeping the others' actions fixed. This process is iterated across the entire team.
- Approximation as Bayesian Games: A Dec-POMDP can be approximated as a series of Bayesian (or Markov) Games. This is accomplished by giving each agent the same payoff in each game and allowing the payoff vector to transform Markovially at each time-step. By determining the solution to each of the games in turn, an approximate solution to the Dec-POMDP
- Selecting sub-tree policies: This considers standard tree search methods such as breadth first search or depth first search. This is used with the aim to reduce the number of possible policies which are searched over.

Whilst the above solutions are show lower complexity than the brute force approach, they are still limited in their tractability to large scale problems, perhaps with many agents or larger (even continuous) action spaces. This intractability largely occurs due to the fact that Dec-POMDP methods attempt to find optimal actions across the entire team at each time-step. This is in contrast to, for instance, swarm methods in which actions are chosen on a local basis and so computation is not affected by the size of the team.

Approximate solutions to Dec-POMDP have been proposed, perhaps most notable of which is the proposal of MacDec-POMDP [?] 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 [?], 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 [?] 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.