Reasoning about Uncertainty and the Effect of Collusion Amongst Spies

Name: Simon McGee

Student Number: 19800483

# Introduction

## Background

Resistance is a hidden role game of 5-10 players in which each player is designated a role as either a resistance or a spy. The goal of resistance is to ensure missions go ahead and succeed. The goal of spies is to either prevent missions from going ahead by voting against them or by ensuring that enough spies are on the mission to sabotage it causing it to fail.

Resistance players are initially unaware of the spies within the group however spies are provided with information as to who the other spies are and therefore know the type, either resistance or spy, of all players in the game. As moves are made simultaneously the resistance and spies are both playing a game of imperfect information. It is however the case that spies have more information at the start of play than the resistance. The resistance members must try to deduce the players that are spies and prevent them from being included in missions to prevent sabotage.

## Research Question

The goal of this project was initially to determine whether an agent using inference rules (Inference Agent) would outperform a deterministic agent (Deterministic Agent) both as a member of the resistance and as a spy in games of 5-10 players.

In the process of building Deterministic Agent it was found that collusive behaviour could be built into the spies before the game allowing them to agree on which spies on a mission would sabotage it without requiring communication. This minimises the footprint of suspicion with multiple spies by ensuring only the minimum betrayals are undertaken. At this point the project question was extended from building an agent using inferential rules to include determining how collusion can change the win loss ratios.

# Literature Review

Methods for playing The Resistance using hard coded deductive logic provide a solid foundation for implementing a base set of rules that are fail safe and should exist in conjunction with other methods. An example of this logical play is to decide that in the event that the number of agents on the mission is equal to the number of betrayals all of the agents on the mission are spies. Likewise, as a spy if we get to round three and are on the mission we should betray the mission if we are the only spy otherwise the spies cannot prevail. This type of logic is deduced through playing the game. These pre-existing go-bies labelled ‘Expert Rules’ are discussed extensively in Taylor (2014). The majority of information in The Resistance does not provide definitive evidence meaning other techniques are required when we have less certainty.

Various search methods covered in the unit so far are feasible for use in creating an agent to play The Resistance. As there is a finite number of possible moves and rounds, A\* and Minimax are possible options. However analysis shows there is a large state space for even the five-player game estimated by Serrino *et al* (2019) as 10^56. As the game type moves towards ten players this space becomes significantly larger as the combinations of teams, votes and outcomes for each mission increase. As I am looking at the five to ten player state space it would be difficult to create efficient search. For this reason, it was decided that other methods would prove easier to implement with better results.

Neural networks seem to be an obvious choice for AI agents given their popularity, the access to libraries and with recent research showing that they work well specifically for The Resistance (Serrino *et al*2019). However, there is not much we do not know about the game play for The Resistance running information through a neural network seemed like it would be a lot of work for something that could be modelled using simpler methods. Further to this, access to compute resources and the learning curve in setting it all up given the time constraints was deemed prohibitive. Monte Carlo Tree Search seems to be the one of the most popular methods for playing hidden role and hidden information games (Reinhardt 2020; Taylor 2014; Whitehouse 2014). Given the amount of time required for this project setting this up was also deemed infeasible.

Fuzzy logic is based on human reasoning and the use of imprecise information to generate decisions. This fits particularly well with understanding roles in The Resistance as we have the belief that each agent is a member of the set ‘Spy’ or ‘Resistance’ to some degree but are usually not certain of what their membership actually is. The selected method uses Inferential methods for determining whether an agent is a spy based on past actions (votes and mission proposals) or actions they have been associated with (successful/failed missions). This uses a general method for reasoning about uncertainty in which we associate actions or association with actions with our belief that an agent is or isn’t a spy.

## Collusion

Collusion is common in games of strategy particularly for gamblers profit from the house. In popular culture there are numerous books, films and examples of a team of players using collusive tactics to beat the house, the most famous of which is likely the MIT Black Jack Team. Research has shown that this is an effective model for winning (Ullman 2015; Zehnder 2012). Further to this is a significant body of on detecting collusion in games (Mazrooei *et al* 2013; Yan 2010).

Research exists within the realm of generating collusion within agent models. However, the use of collusive tactics within simultaneous games is an approach that does not appear to have much literature as it side steps the idea of fairness within the game. This places it outside the bounds of most research which focus on optimising play against fair opponents. Similarities do exist in the literature on the generation of predator prey models in which the predators/prey are able to cooperate to hunt/avoid prey however this exists more within the bounds of multi-agent learning rather than preconfigured cooperation (Stone and Veloso 2000).

There is some research into collusion using imperfect information games in AI however this focuses on having imperfect information at each stage (Farina *et al* 2018). For example, working towards a common goal of wins while not knowing what hands the colluding players holding poker. This is a significantly more complex problem than the type of collusion being implemented for this project which relies on the perfect information of the spies as to who the spies are, when they are on a mission and a using having or not having the highest player number as the trigger to betray.

# Selected and Design of Methodology

The Resistance is a game of determining roles given uncertainty attribution of actions and uncertainty in reasoning about known actions. As a resistance player we are required to make inference about whether we believe an agent is a fellow resistance member or a spy based on their actions and perceived actions.

Two agents, Deterministic Agent and Inference Agent were developed for this project. The first agent, Deterministic Agent was developed as a baseline using preconfigured rules determined from game play and the expert rules found in Taylor (2014). These rules are considered best practice, even within Inference Agent so are left there largely unchanged with rules based on inference rather than definitive evidence coming into play only when the criteria for triggering them are not met.

Inference Agent starts off with a single belief value for each agent in the game. This metric is used as a relative measure of how much we trust each player. Generally speaking when an agent is on a mission which succeeds or votes and proposes for missions that are deemed low risk Inference Agent will increase the level of trust we have in that agent. Likewise, when an agent acts in a manner that we believe is counter to roles of each player Inference Agent decreases the level of trust it has in that agent. Exceptions do exist such as in the case we have definitive knowledge that a player is a spy we will not increase our level of trust even if a mission they are on succeeds.

At the start of a game each player is given three metrics upon which we base our trust: Mission Distrust, Vote Distrust and Proposal Distrust. Each player is also provided five penalty values with which to adjust the trust of other players depending on their actions. These are for being on a failed mission, proposing a failed mission, voting for a failed mission, voting for missions that are seen to be high risk and proposing high risk missions. Penalties are added (distrust) or subtracted (trust) from the distrust level depending on game events.

The penalties while stand-alone metrics are tempered by our confidence in the event and the players existing distrust level. For example, when a player is on a mission that has three agents and one betrayal occurs 1/3 is used as a multiplier as we assume only a 1/3 chance that they betrayed the mission. This is further multiplied by the agents existing distrust level. In the case that the agent has a higher existing distrust level the trust adjustment of the association will impact them more than if they had a lower initial distrust level. In the event that there are two betrayals on a three-agent mission this is factored in by using 2/3 as a multiplier and so forth.

The sum of the trust levels are Inference Agent’s overall Distrust Level for that agent. The trust levels were split for two reasons, neither of which has an influence on the agent as it stands as the implementation of the functionality was unachievable within the time frame. First was so that players of different types could be grouped by determining a play type. This would mean that players that vote in what is perceived as a responsible manner but are associated with failed missions could be differentiated from those that vote poorly and are associated with failed missions. Further to this, penalties associated with each metric could be then be tuned based on game play in various scenarios.

Trust and penalty parameters were originally randomly selected and then hand tuned provided the desired outcomes. However, there is undoubtedly a set of values that will maximise the success of the agents. It was initially thought that this could be optimised by using a simple hill climbing search or possibly by using genetic algorithms to pit agents against each other. Unfortunately, time was not on the side of this plan so it was shelved. Names of some of the classes and modules refer to ‘genetic’ as this was the plan however the project should not be understood to include genetic algorithms.

Inference Agent’s trust of other agent is used to determine its confidence in a mission succeeding given the proposed agents. This allows Inference Agent to determine which missions it should propose as a resistance member to maximise the chance of success by selecting the agents with the lowest distrust level. Likewise, Inference Agent will cast votes based on its assessment of whether the mission will succeed. When Inference Agent notices players that we consider more trustworthy assigning agents it considers less trustworthy, it has the option to update trust values based on this new information.

As a spy the Inference Agent uses this same technique to their advantage as they can understand which missions will appear most acceptable to resistance agents. It is advantageous for spies to know which missions will ‘appear’ to succeed whilst having perfect information about the composition of spies on the mission. With this information the spy is able to put forward a mission that initially appears like it will succeed but will likely be betrayed. In knowing this Inference Agent when acting as a spy that knows they are untrustworthy makes assessments of resistance that appear untrustworthy and suggests them for missions in order to bluff the resistance into suspecting them.

The rules for collusion were first developed in Deterministic Agent and brought into Inference Agent as it has always been an extension to the original agent. Collusion is included in the agent and is switched off by default. Deterministic Agent and Inference Agent have functions that turn collusion on and off either prior to or during play. In the event that collusion is enacted for resistance agents nothing is affected. The rules for collusion follow very simple logic. When acting with collusion the agent will get the list of spies on the mission. If the agent had been assigned the maximum player number of the spies on the mission the agent would betray the mission. In the event that they had any other number they would not. This guaranteed that the mission would fail but the footprint would be as small as possible. With collusion on it became effectively impossible for the resistance to win a game unless they went through three games without a spy on the mission by chance.

# Validation of Agent Performance

Various scenarios were created to test the effectiveness of agents across game types with 5-10 players. First was to bench mark the win/loss for the agent acting as both spy and resistance to know how well they play against themselves. Tests were then developed with agents assigned a role based on their type e.g. all spies are Inference Agent and all resistance Deterministic Agent to assess the changes based on specific roles. The last tests check how a single agent acting as either a spy or as resistance affect the outcome of a game consisting of entirely different agents. To do this the Game class was refactored into AllocatedAgentClass which allowed these specific game types to take place. Validation was undertaken by running agents through 10,000 games under each test type for games of 5-10 players.

## Beating Random

Games played solely with the Random Agent show that it is incapable of providing a strategy that will provide victory in a consistent manner for the spies. This is due to their inability to use the perfect information they hold as part of a strategy and the inconsistent manner in which they will betray work to prevent mission success.

As shown in Figure 1, Deterministic Agent was able to match the relatively high resistance wins of the Random Agent and when working as spies significantly reduce the number of wins for the resistance.

Figure 1. Victories for Random Agents

The effect of the Deterministic Agent on play by the Random Agent is particularly clear when we look at the effect that adding a single Deterministic Agent spy on the game. Figure 2 shows that a single Deterministic Agent acting as a spy amongst otherwise Random Agents has a significant effect on resistance victories. This appears to be due to the increased aggression Deterministic Agent when acting as a spy and the inability of Random Agents to react to this. The most significant effect is in the games of five and six players where the Deterministic Agent is more likely to be selected for missions. As number of player increases the likelihood of selection goes down, so too does the ability of the agent to affect change. The effect of adding a single Deterministic Agent to the resistance was negligible due to the already high win rate and the limited ability to shape mission selection.

Figure 2. Effect of a Single Deterministic Agent on Random Play

## Beating Deterministic

The set rules of the Deterministic Agent prevent it from making inference about a player given past activities that are not completely incriminating. The Bayesian Agent improves on this by modifying the trust level of the agent and selecting missions based on how likely other Agents are to be spies.

The goal of the Inference Agent was to be able to improve upon the win ratio as both resistance and spies in relation to how the Deterministic Agent did against itself. As seen in Figure 3, this was mostly successful. The superior win rate of the Deterministic Spies when playing Deterministic Resistance has not been analysed fully but is likely due to the Inference Agent’s playing a more careful game as spies in order to lower their suspicion profile. As the number of players goes up the overall suspicion profile drops so it is believed that this allows the Bayesian Spies to play more aggressively which is shown by their increased wins in games of 7-10 players.

Figure 3. Deterministic vs Inference Victory Comparison

The risk aversion shown by the Inference Agent when playing as a spy against the Deterministic resistance is also evident in how the Inference Agent plays itself. It is likely that the weighting of the risk aversion parameters need to be adjusted for the number of players in order to optimise the agent.

## Collusive Play

The inclusion of collusion amongst Deterministic and Bayesian Agents allows for increased aggression in games where multiple spies are on the same mission by reducing the chance of multiple spies betraying the mission and therefore raising their suspicion to zero. The effect is most notable against the Deterministic Agents which require either spies outing themselves to all players or to individual players over successive missions in order to succeed.

Figure 4. Deterministic Resistance vs Colluding and Non-Colluding Spies

Using collusion against Bayesian Agents has mixed though largely positive results. Once again there appears to be some issues in the 5 and 6 player games. It is possible that this is caused by the same wariness in game play mentioned in Bayesian Agents above however it could be the increased confidence of the Bayesian Agents that outs them quickly to the resistance. Once again it is in the games with increased numbers of players that the spies are able to act more boldly while still maintaining a relatively low suspicion or at least keeping spies with minimal suspicion in reserve to betray later missions.

Figure 5. Inference Resistance vs Colluding and Non-Colluding Spies

# Bibliography

[Cowling, P.I, Powley, E.J, Whitehouse D, (2015) ‘Emergent Bluffing in Inference with Monte Carlo Tree Search’, IEEE Transactions on Computational Intelligence and AI in Games, 4(2) pp.120–143](https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7317927)

Dutta, P.K. (1999). Strategies and Games: Theory and Practice. Cambridge, MA, The MIT Press. Chapters 14-15 for Repeated Games. Chapter 20 for Games with Incomplete Information.

[Farina, G, Celli, A, Gatti, N, Sandholm, T, (2018) ‘Ex ante coordination and collusion in zero-sum multiplayer extensive form games’ Advances in Neural Information Processing Systems, Vol 31](https://proceedings.neurips.cc/paper/2018/file/c17028c9b6e0c5deaad29665d582284a-Paper.pdf)

[Mazrooei, P, Archibald, C, Bowling, M, ‘Automating Collusion Detection in Sequential Games’ Proceedings of the 27th AAAI Conference on Artificial Intelligence](https://www.aaai.org/ocs/index.php/AAAI/AAAI13/paper/download/6448/7235)

Morovat, K. (2013), ‘Fuzzy Reasoning’, Presentation, University of Arkansas.

[Reinhardt, J. (2020) ‘Competing in a Complex Hidden Role Game with Information Set Monte Carlo Tree Search’ arXiv:2005.07156v1 [cs.AI]](https://arxiv.org/pdf/2005.07156.pdf)

[Serrino, J, Kleiman-Weiner, M, Parkes, D.C, Tenenbaum, J.B,  *et al* (2019) ‘Finding Friend and Foe in Multi-Agent Games’ arXiv:1906.02330v1 [cs.LG]](https://arxiv.org/pdf/1906.02330.pdf)

[Stone, P. and Veloso, M (2000) ‘Multiagent Systems: A Survey from a Machine Learning Perspective’, Autonomous Robots, Vol 8, pp.345-383](https://link.springer.com/content/pdf/10.1023/A%3A1008942012299.pdf)

[Szita, I, Chaslot, G, Spronck, P, (2009) ‘Monte Carlo Tree Search in Settlers of Catan’, Advances in Computer Games pp.21-32](https://link.springer.com/content/pdf/10.1007%2F978-3-642-12993-3.pdf)

[Taylor, Daniel P. (2014) ‘INVESTIGATING APPROACHES TO AI FOR TRUST-BASED, MULTI-AGENT BOARD GAMES WITH IMPERFECT INFORMATION WITH DON ESKRIDGE’S “THE RESISTANCE”’](https://teaching.csse.uwa.edu.au/units/CITS3001/Resistance.pdf)

[Ullman, R.J, (2015) ‘Optimal Poker Strategies Under Collusion’ PhD Thesis, Princeton University](http://bobbyullman.com/files/poker/thesis-fall15.pdf)

[Whitehouse, Daniel (2014) ‘Monte Carlo Tree Search for games with Hidden Information and Uncertainty’, PhD thesis, University of York](https://core.ac.uk/download/pdf/30267707.pdf)

[Yan, J. (2010) ‘Collusion Detection in Online Bridge’ Proceedings of the 24th AAAI Conference on Artificial Intelligence](https://www.cs.huji.ac.il/~jeff/aaai10/02/AAAI10-265.pdf)

Young, H. Peyton (2007) ‘The Possible and Impossible in Multi-Agent Learning’, Artifical Intelligence (AIJ), Vol 171, pp. 429-433

[Zehnder, B. (2012) ‘Collusion in Online Poker Pays Off’ Bachelor’s Thesis’, Swiss Federal Institute of Technology](https://pub.tik.ee.ethz.ch/students/2012-FS/BA-2012-03.pdf)