
Master-Slave Key Detection and Exploitation in Iterated Prisoner’s Dilemma

Peter A. Hamiton
1522 Copeland Rd.
Catonsville, MD 21228-1137

PETE5@UMBC.EDU

Keywords: game theory, master/slave team strategy, multi-agent systems, Iterated Prisoner’s Dilemma, master/slave key, key exploitation, code breaking

1. Introduction

Team strategies, especially Master/Slave, do quite well in both noise free and noisy game playing tournaments (Delahaye & Mathieu, 1993; Delahaye, 1995), primarily through the transmission and detection of an identification code that allows team members to discriminate between other team members and tournament opponents. Previous work has shown that error correction codes (ECCs) can be applied to an ID code used in noisy Iterated Prisoner’s Dilemma (IPD) tournaments, allowing team members to recognize codes corrupted by noise (Rogers et al., 2007). This research draws primarily from that work and examines the viability of a key detection scheme that will allow a “Spy” agent to accurately detect the transmission of team’s ID code and thereby allow the Spy to infiltrate the team and take advantage of the cooperation among team members.

While both noise free and noisy environments are of interest, this research focuses primarily on the noise free IPD, allowing for a clear demonstration of the viability of key exploitation without the hindrances entailed in handling transmission noise.

2. The Spy

The Spy’s functionality is split into two main parts, key detection and key exploitation, which generally correspond to non-deception and deception in the infiltration process. An overall state diagram of the Spy is provided in Figure 1.

3. Key Detection

To successfully detect possible keys, a sample of opening moves must be available for the Spy to pick from. At the start of each round, the Spy records the open-

ing move sequence of its opponent, which is 15 moves, or bits, long. A maximum key length of 15 was chosen because it is the recommended length for the key and ECC scheme in the original work. Note that the Spy need not understand the ECC method used to verify the key in the noisy IPD. Since the ECC is unique to the original key, it effectively becomes the second part of the “new” key (original key + ECC).

The Spy records the number of occurrences for each unique key, eliminating those keys that correspond to “All Cooperate” or “All Defect” strategies. This pruning is done for two reasons: (1) “All Cooperate” and “All Defect” strategies are actually quite common in IPD tournaments (“All Defect” moreso since it is the Nash equilibrium for Prisoner’s Dilemma), and (2) the complexity of a team key must be greater than a simple sequence of 0s and 1s, (i.e. “Cooperates” and “Defects” respectively) to adequately distinguish team members.

The Spy examines the distribution of keys and picks out those that have occurred at least γ times, where γ is a key viability threshold set at the beginning of the tournament. For this research, $\gamma = 3$. The Spy then selects the key with the largest occurrence count and adopts it as its opening move sequence. Note that while there is no key whose occurrence count exceeds the viability threshold, the Spy defaults to the CC-TFT strategy (i.e. two “Cooperates” to solidify cooperation and then Tit-for-Tat to propagate it), a generic strategy which is recommended in the original work.

3.1. Key Exploitation

Once the Spy has selected a key, it changes its behavior to mimic that of a Slave in the infiltrated team. For this work, the definition of a Slave agent defined in the original work is used for simplicity (i.e. upon

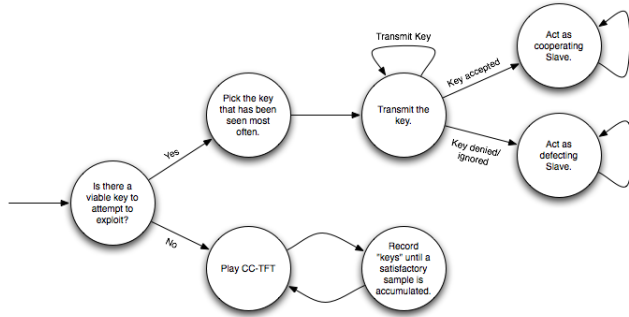


Figure 1. The general state diagram of the Spy agent splits the majority of the Spy’s actions into two categories: deception and non-deception. While the Spy constructs its dictionary of possible team keys, it employs a modified TFT strategy that actually does quite well in general IPD play (see Figure 3). Once the Spy selects a key, it starts by transmitting the key and then checking the opponent. If the opponent did not send their key, or if it does not cooperate, the Spy acts like a “jilted” Slave agent and defaults to perpetual defection.

Slave recognition, both Slaves cooperate to boost their collective score, and, upon non-recognition, the Slave defects to prevent exploitation). Since an agent, in general, does not know if it is playing in a noisy environment, the Spy will start a match by transmitting its key and recording the opening move sequence of its opponent. If the opponent’s opening move matches the Spy’s key, the Spy assumes it is playing another Slave and adopts the “All Cooperate” strategy to boost its performance and avoid “betrayal” detection by the opposing Slave.

If the Spy is detected and/or betrayed by the Slave, or if the opposing agent haphazardly transmitted the key unknowingly, the Spy will detect the anomalous behavior (i.e. anything but “All Cooperate”) and will, like the Slave, switch to the “All Defect” strategy. Note that, since there is only one Master per team, the Spy will never recognize the Master, since it is too hard to pick out the Master’s key from the set of stored opening move sequences. The Spy will therefore never have to sacrifice itself for the Master but will still reap the benefits of cooperating with the Slaves. This simple difference typically ensures that the Spy performs, in the overall tournament, better than most if not all of the Slaves in the infiltrated team.

4. Experiments and Performance

Several experiments were conducted to verify the viability of the key detection and exploitation process. For each experiment, each data point is averaged over

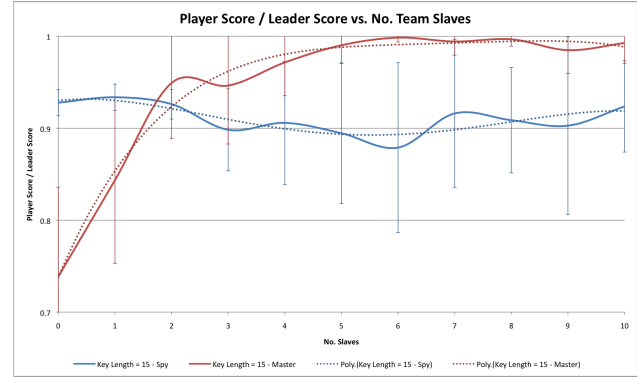


Figure 2. The performance of both the Master and the Spy plotted according to increasing team size (i.e. increasing number of Slaves). The adjusted scores for both agents are computed by dividing the agent’s overall score in the tournament by that of the tournament winner. Trend lines are graphed with the actual data points to depict the general shape of the performance curves.

10 distinct tournament runs where the keys for the Master/Slave team are randomly generated for each tournament (note that “All Cooperate” and “All Defect” keys are excluded). Figures 2 and 3 depict the results of multiple noise free IPD tournaments for the Master and Spy agent scores. In Figure 2, the adjusted score for both agents is computed relative to the score of the tournament winner. Since the Master wins often and with a larger margin for large Master/Slave teams, the difference between its score and the second place agent’s score, which is usually the Spy, grows as the team size increases. This does not mean the Spy is performing worse as the Master/Slave team size increases. On the contrary, the Spy improves its performance since there are more Slaves to take advantage of. Note that in general, if $S \gg \gamma$, where S is the number of Slaves in the infiltrated team and γ is the key viability threshold, the Spy will do well since there are plenty of remaining Slaves to take advantage of.

Figure 3 plots the actual ranking of the Master and Spy relative to the number of actual players in the tournament. The higher the agent ranking, the better the performance. Rankings are also normalized by the total number of players in a tournament. For example, for a tournament with 30 players, an agent ranked 30 (the maximum ranking possible), receives a normalized ranking of 1 (30/30). Another player with a ranking of 20 (i.e. borderline top third ranking), would receive a normalized ranking of approximately 0.667.

As shown in Figure 3, the Spy demonstrably improves its performance by exploiting the team key transmission process. For teams with approximately 10 agents,

the Spy even occasionally ousts the Master from first place to win the tournament.

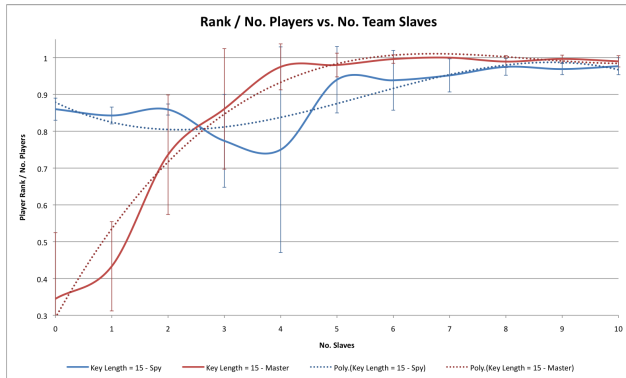


Figure 3. The performance of both the Master and the Spy plotted according to increasing team size (i.e. increasing number of Slaves). The adjusted scores for both agents are computed by dividing the agent’s overall score in the tournament by that of the tournament winner. Trend lines are graphed with the actual data points to depict the general shape of the performance curves.

5. Future Work

Due to the number of parameters involved in the Master/Slave/Spy dynamic (e.g. team size, key length, recognition strategies, etc.), there is plenty of subsequent work that can be done in the key exploitation process.

5.1. Expanding to the Noisy IPD

The importance of noise in IPD lies in the level of accuracy team members have in recognizing and coordinating with other team members. The more noise, the harder it is to verify the identifying key code. In light of the design choices made for the Spy, several modifications must be made. First, noise effectively ruins the Spy’s ability to accurately pick out repeated keys, since one flipped bit will result in a different key. A possible solution to this problem involves the use of code edit distance. The Spy could, for each new opening move sequence, compute the edit distance between the new prospective key and each key in its set of recorded keys. Since the original work indicates that the best ECC can only account for up to three flipped bits, if the edit distance between keys is less than or equal to three, it is possible the new key is a repeat of a previous opening move sequence.

5.2. Optimizing γ

Varying the key viability threshold, γ , varies the time the Spy spends analyzing the pool of opponents in search of a possible team key. Ideally, γ should be defined in terms of the number of tournament players. A small γ for a large tournament will cause the Spy to rapidly switch keys and, potentially, default to “All Defect” for most matches if the selected move sequence is not an actual key. Conversely, a large γ will cause the Spy to play against most of the Slaves on a team before verifying the key, leaving the Spy with few remaining matches to exploit. Studying the impact of tournament size and the value of γ on Spy performance would be an interesting extension of the current work.

6. Conclusion

This research presents a key detection and exploitation scheme that successfully allows a Spy agent to infiltrate a Master/Slave team, defeating most Slaves in an IPD tournament and, occasionally, the Master as well. Spy performance has been tested against team size and has been shown to dramatically improve with the use of key exploitation. Additional future work has been given to propose solutions to the problems posed by the noisy IPD.

References

- Delahaye, J. P. (1995). Logique, informatique et paradoxes. *Bibliothèque Pour la Science*.
- Delahaye, J. P., & Mathieu, P. (1993). L’altruisme perfectionné. *Pour la Science*, 102–107.
- Rogers, A., Dash, R. K., Ramchurn, S. D., Vytelingum, P., & Jennings, N. R. (2007). Coordinating team players within a noisy iterated prisoner’s dilemma tournament. *Theoretical Computer Science*, 243–259.