Exploring optimization strategies with the prisoner's dilemma

Jay Chandrasekaran, Jaxon Hornsey, Ali Naqvi, Daniela Espinosa-Perdomo
Dept. of Computer of Science
University of Windsor, Windsor, Canada
{thennam, hornseyj, naqvi6, espinos} @uwindsor.ca

***Abstract*— The prisoners' dilemma is a popular game used in game theory to show why rational individuals may not choose to 'cooperate'. The dominant strategy in this paradox can be seen to be 'defection' rather than 'cooperation'. In the iterated prisoner's dilemma (IPD), the players play the game repeatedly, and base their moves on prior decisions made by their opponent. Unlike prisoner's dilemma, in IPD, the desire for long-term reward provides good incentive for players to cooperate. The players can adopt several strategies to maximize their gain based on their opponent's past move. For the current project, the existing online code for Axelrod's IPD was used to run simulations and generate sample data. Different optimization methods like brute-force search, hill climbing search and tabu search were applied to find effective strategies for IPD. Our research concluded with the most successful strategies being 'Tit-for-tat' (TFT) and its variant 'Tit-for-2-tat' (TF2T). 'Grudger' also trailed being behind emerging as a good strategy.**

I. INTRODUCTION

A game-theoretic paradox, "Prisoners' Dilemma", is a situation modelled by cooperation and conflict. In this paradox, two individuals who are unable to communicate with each other must choose to either cooperate with each other or not. The reward system for everyone varies based on their decisions and the highest reward is given if both choose to cooperate.

		Prisoner 2	
		Cooperate	Defect
Prisoner 1	Cooperate	3, 3	0, 5
	Defect	5, 0	1, 1

Figure 1: Sample model of the Prisoner's Dilemma (IPD Payoff Matrix)

The purpose of this report is to analyze and compare a variety of optimization methods for the "Iterated Prisoners' Dilemma" (IPD) in order to find the best possible strategy. The performance of each method will be monitored and tweaked with factors such as population size and strategies. A variety of optimization methods will be used such as brute-force, hill climbing, and tabu search. These strategies have been tested in a simulated environment and modelled for proper analysis.

II. RELEVANT LITERATURE REVIEW

The purpose of the paper is to review some of the methods that can be used to optimize the prisoner's dilemma game. This prisoner's dilemma game, as previously mentioned, is a very commonly used tool in the modeling as well as the formalization of some of the most complex interactions amongst groups. Every player is trying to find the best possible way which they can be applied so that they can be in a position to maximize the long-term rewards. This game has been in application ever since the 1950s as a strong framework for the aspect of the development of cooperation [2]. This topic has sustained its continued interest in topics like the biological, mathematical, social, and environmental sciences. The iterated model of the prisoner's dilemma (IPD) did draw a lot of attention in the years of 1980s, following the "The Evolution of Cooperation" publication, and it has remained a topic of original study ever since then [1]. In the early years, the prisoner's dilemma was only possible as a game with human players; however, a political scientist, Axelrod, who was interested in artificial intelligence, introduced computer tournaments that led to the development of a theoretical framework. This in return would help in the development of the IPD. His experiments had requested researchers to develop strategies for performance using these strategies. In both his tournaments, he found that tit for tat (TFT) was the winning strategy [3]. However, the study came with many loopholes requiring further research that led to the development of other intelligent strategies such as Pavlov and Gradual. Soon, researchers discovered that the strategies were not only useful in solving theoretical problems but also in solving complex real-world problems. This discovery led to the implementation of dynamic evolutionary methods in the fields. Real-world applications had researchers identify factors that would increase cooperation. This strategy was implemented in communities where subjects could interact freely and in where subjects were isolated; they found that even in a game of low chances, cooperation can emerge. Evolutionary approaches offer many insights when trying to understand IPD because these strategies can learn to adapt and control the population by adjusting their actions [4]. Such strategies can then be used to build even more robust frameworks. The study of the prisoner's dilemma is an open study where new variants and structures of strategies continue to be developed. The prisoner's dilemma serves a wide range of use, for example, the study of cancer cells as well as in social situations where people can be driven by reward.

III. EXPERIMENTAL SETUP AND METHODOLOGY

The environment that was used for the experimentation of this paper used Python as its programming language. For all experiments, the popular Axelrod library was utilized [6]. Each experiment uses an optimization method to find out the best strategy for different scenarios. The Axelrod library contains strategies that will be implemented in the simulations [6]. As mentioned before, this paper is to analyze and compare the results of winning strategies from different optimization methods and the effect of their parameters.

The methodology for this research was comparing different types of optimization strategies: the tabu search, the hill-climbing strategy as well as exhaustive search also known as Brute Force. The hill-climbing method is a search strategy that continuously moves in the direction of the value that is increasing within a study in order to find the top of the hill (the very best solution to the prisoner dilemma). The hill climber method ends whenever it hits the topmost value where there is no counterpart with a higher value. The main features of the hill-climbing method are things such as: the greedy approach; the hill-climbing strategy goes to the direction that optimizes on any cost that might be involved; generate the variant and test it. It will generate the best possible result, this strategy does not take into account/ remember the previous states. The tabu search method of optimization is based on the memory structures, where the information of the recently evaluated players is stored. Brute force generates all possibilities and tests them to find the optimal choice; this option however takes the most amount of time and computer power. The efficiency is O(n*m). The tabu search is mostly used in the optimization of the problem with the best possible strategy. This algorithm presents a rapid convergence for searching to come up with the best solution for a prisoner dilemma situation.

A. *Exhaustive Search Setup*

7 different strategies will be used throughout this experiment:

- Defector (given in the Appendix)
- Tit-for-Tat (TFT) (given in the Appendix)
- Tit-for-2-Tats (TF2T) (given in the Appendix)
- Random (given in the Appendix)
- Suspicious-Tit-For-Tat (STFT) (given in the Appendix)
- Cooperator (given in the Appendix)
- Grudger (given in the Appendix)

The first experiment was conducted using brute force optimization in mind. The library contains a tournament function that sets up a round-robin tournament with experimental variables such as noise (the probability that a player's action will be swapped with another strategy), turns to play against each other, and repetitions of the tournament to smoothen out stochastic effects [6]. A round-robin tournament with 7 players each using one of the seven strategies mentioned before was created. The round-robin allows us to match each strategy against another strategy while keeping track of points. After all, players have played their rounds, the player with the most points wins. For some strategies, the number of turns will inherently give it either an advantage or disadvantage. This experiment will explore the best strategy for our given scenarios:

Round Robin with 5 turns and no noise
Round Robin with 15 turns and no noise
Round Robin with 250 turns and no noise

These scenarios have been specifically picked to demonstrate the difference in best strategies due to changing scenarios. The tournaments will use varying turns and no noise to solely compare the strategies against one another. To ensure accurate results the tournament was run 10 times with a different seed in each case. The purpose of changing the seed allows stochastic strategies to change their moves [6].

B. *Hill Climbing and Tabu Search Setup*

1) *Hill Climbing Search:* Hill climbing is a type of local search for optimization where we start with an arbitrary solution and make small incremental changes repeatedly until we reach our goal state. We place all the strategies we have from our complete set of strategies(S) into an array. We can pick any random strategy and start the tournament. Here we start by picking the first pair of strategies from the strategies array and play a tournament with the third strategy. Out of this trio, the strategy that accumulates the least points (based on pay-off matrix) is the loser and the other two winning strategies advance to play next tournament with the next strategy. At any point in the game, the best two strategies from the preceding list of strategies would be playing the current strategy. When the last trio of strategies complete the tournament, the strategy (or strategies in case of tie) with the most points is deemed to be the best strategy from the set of strategies we started with.

a) Hill Climbing Experiment Results Strategy Set (S): {Defector, Random, Cooperator, Grudger, TFT, Alternator, Bully, TF2T}

<u>Round 1.</u>
Participating Trio: [Defector, Random, Cooperator]
Simulation output:
(Defector, Random: 0.5) 5 [(D, C), (D, C), (D, D), (D, C), (D, D)]
(Random: 0.5, Defector) 5 [(D, D), (C, D), (D, D), (D, D), (C, D)]

(Defector, Cooperator) 5 [(D, C), (D, C), (D, C), (D, C), (D, C)]
(Cooperator, Defector) 5 [(C, D), (C, D), (C, D), (C, D), (C, D)]
(Random: 0.5, Cooperator) 5 [(D, C), (C, C), (C, C), (C, C), (D, C)]
(Cooperator, Random: 0.5) 5 [(C, C), (C, D), (C, D), (C, C), (C, D)]
Losing strategy: Cooperator
Strategies advancing to next round: Defector, Random

<u>Round 2.</u>
Participating Trio: [Defector, Random, Grudger]
Simulation output:
(Defector, Random: 0.5) 5 [(D, C), (D, C), (D, C), (D, C), (D, D)]
(Random: 0.5, Defector) 5 [(C, D), (D, D), (D, D), (C, D), (D, D)]
(Defector, Grudger) 5 [(D, C), (D, D), (D, D), (D, D), (D, D)]
(Grudger, Defector) 5 [(C, D), (D, D), (D, D), (D, D), (D, D)]
(Random: 0.5, Grudger) 5 [(C, C), (D, C), (D, D), (C, D), (C, D)]
(Grudger, Random: 0.5) 5 [(C, C), (C, D), (D, D), (D, D), (D, C)]
Losing strategy: Random
Strategies advancing to next round: Defector, Grudger

<u>Round 3.</u>
Participating Trio: [Defector, Grudger, TFT]
Simulation output:
(Defector, TFT) 5 [(D, C), (D, D), (D, D), (D, D), (D, D)]
(TFT, Defector) 5 [(C, D), (D, D), (D, D), (D, D), (D, D)]
(Defector, Grudger) 5 [(D, C), (D, D), (D, D), (D, D), (D, D)]
(Grudger, Defector) 5 [(C, D), (D, D), (D, D), (D, D), (D, D)]
(TFT, Grudger) 5 [(C, C), (C, C), (C, C), (C, C), (C, C)]
Losing strategy: Defector
Strategies advancing to next round: TFT, Grudger

<u>Round 4.</u>
Participating Trio: [Defector, Grudger, TFT]
Simulation output:
(Defector, TFT) 5 [(D, C), (D, D), (D, D), (D, D), (D, D)]
(TFT, Defector) 5 [(C, D), (D, D), (D, D), (D, D), (D, D)]
(Defector, Grudger) 5 [(D, C), (D, D), (D, D), (D, D), (D, D)]
(Grudger, Defector) 5 [(C, D), (D, D), (D, D), (D, D), (D, D)]
(TFT, Grudger) 5 [(C, C), (C, C), (C, C), (C, C), (C, C)]
Losing strategy: Defector
Strategies advancing to next round: TitForTat, Grudger

<u>Round 5.</u>
Participating Trio: [Bully, Grudger, TFT]
Simulation output:
(Bully, TFT) 5 [(D, C), (D, D), (C, D), (C, C), (D, C)]
(TFT, Bully) 5 [(C, D), (D, D), (D, C), (C, C), (C, D)]
(Bully, Grudger) 5 [(D, C), (D, D), (C, D), (C, D), (C, D)]
(Grudger, Bully) 5 [(C, D), (D, D), (D, C), (D, C), (D, C)]
(TFT, Grudger) 5 [(C, C), (C, C), (C, C), (C, C), (C, C)]
(Grudger, TFT) 5 [(C, C), (C, C), (C, C), (C, C), (C, C)]
Losing strategy: Bully
Strategies advancing to next round: TFT, Grudger

<u>Final Round.</u>
Participating Trio: [TF2T, Grudger, TFT]
Simulation output:
(TF2T, TFT) 5 [(C, C), (C, C), (C, C), (C, C), (C, C)]
(TFT, TF2T) 5 [(C, C), (C, C), (C, C), (C, C), (C, C)]
(TF2T, Grudger) 5 [(C, C), (C, C), (C, C), (C, C), (C, C)]
(Grudger, TF2T) 5 [(C, C), (C, C), (C, C), (C, C), (C, C)]
(TFT, Grudger) 5 [(C, C), (C, C), (C, C), (C, C), (C, C)]
(Grudger, TFT) 5 [(C, C), (C, C), (C, C), (C, C), (C, C)]
Winning strategies: TFT, TF2T, Grudger

Please note that the score for strategies were calculated based on 'IPD Payoff Matrix' presented in Figure 1.

2) *Tabu Search:* Tabu search is a metaheuristic search employing a local search method used for optimization. The inferior solutions can be placed in a forbidden list, known as tabu list so that revisiting those solutions can be avoided. In our experiments we apply tabu search along with the hill climbing search. We allocate a random list of strategies (S1, S2, ...Sn) to our subsets (SS1, SS2) from the complete set of strategies(S) we have. Then, we apply the Hill Climbing search (as detailed previously) to the first subset of strategies (SS1) in first level(L1). The highest scoring strategy from S1 is selected and promoted to play in the next level(L2). The worst performing strategy from SS1 is placed in a tabu list (TL). After this, we pick the next subset of strategies (SS2) and proceed to perform Hill Climbing search which would be the current level(L1). When we pick a current strategy, we go through the tabu list (TL) to know if the strategy is inferior based on prior validation done during Hill Climbing search with prior subset (SS1). If the current strategy is in tabu list (TL), the strategy is skipped. If not, the strategy is picked to participate in the tournament. At the end of Hill Climbing search, we will have the best strategy from the current subset (SS2). This strategy is promoted to play in the next level (L2). We can apply Hill Climbing search to select the best from the list of strategies promoted from prior levels. The winning strategies proceed to higher levels to compete and at the end result in a set of highly successful strategies. When the last trio of strategies complete the tournament, the strategy (or strategies in case of tie) with the most points is deemed to be the best strategy from the set of strategies we started with.

b) Tabu Search Experiment Results Strategy Set (S): {Defector, Random, Cooperator, Grudger, TFT, Alternator, Bully, TF2T}

Strategy Subset1 (SS1): {Defector, Random, Cooperator, Grudger}
Strategy Subset2 (SS2): {Cooperator, Grudger, TFT, Alternator, Bully}

<u>Level 1(SS1): Round 1.</u>
Participating Trio: [Defector, Random, Cooperator]

Tabu List (TL): empty
Simulation output:
(Defector, Random: 0.5) 5 [(D, C), (D, D), (D, D), (D, D), (D, C)]
(Random: 0.5, Defector) 5 [(C, D), (D, D), (C, D), (C, D), (C, D)]
(Defector, Cooperator) 5 [(D, C), (D, C), (D, C), (D, C), (D, C)]
(Cooperator, Defector) 5 [(C, D), (C, D), (C, D), (C, D), (C, D)]
(Random: 0.5, Cooperator) 5 [(D, C), (C, C), (D, C), (C, C), (C, C)]
(Cooperator, Random: 0.5) 5 [(C, C), (C, D), (C, C), (C, C), (C, D)]
Losing strategy: Cooperator
Strategy added to Tabu List (TL): Cooperator
Strategies advancing to next round: Defector, Random

<u>Level 1(SS1): Round 2.</u>
Participating Trio: [Defector, Random, Grudger]
Tabu List (TL): Cooperator
Simulation output:
(Defector, Random: 0.5) 5 [(D, C), (D, C), (D, C), (D, C), (D, D)]
(Random: 0.5, Defector) 5 [(C, D), (D, D), (C, D), (C, D), (D, D)]
(Defector, Grudger) 5 [(D, C), (D, D), (D, D), (D, D), (D, D)]
(Grudger, Defector) 5 [(C, D), (D, D), (D, D), (D, D), (D, D)]
(Random: 0.5, Grudger) 5 [(D, C), (C, D), (D, D), (D, D), (D, D)]
(Grudger, Random: 0.5) 5 [(C, D), (D, D), (D, C), (D, D), (D, C)]
Losing strategy: Random
Strategy added to Tabu List (TL): none
Winning strategies of Level-1: Defector, Grudger

<u>Level 1(SS2: Round 1.</u>
Participating Trio: [Cooperator, Grudger, TFT]
Tabu List (TL): Cooperator
Since Cooperator is in Tabu List, the strategy is skipped.
Losing strategy: Cooperator
Strategy added to Tabu List (TL): none
Strategies advancing to next round: Grudger, TFT

<u>Level-1(SS2): Round-2.</u>
Participating Trio: [Grudger, TFT, Alternator]
Tabu List (TL): Cooperator
Simulation output:
(Alternator, TFT) 5 [(C, C), (D, C), (C, D), (D, C), (C, D)]
(TFT, Alternator) 5 [(C, C), (C, D), (D, C), (C, D), (D, C)]
(Alternator, Grudger) 5 [(C, C), (D, C), (C, D), (D, D), (C, D)]
(Grudger, Alternator) 5 [(C, C), (C, D), (D, C), (D, D), (D, C)]
(TFT, Grudger) 5 [(C, C), (C, C), (C, C), (C, C), (C, C)]
(Grudger, TFT) 5 [(C, C), (C, C), (C, C), (C, C), (C, C)]
Losing strategy: Alternator
Strategy added to Tabu List (TL): none
Strategies advancing to next round: Grudger, TitForTat

<u>Level-1(SS2): Round 3.</u>
Participating Trio: [Bully, Grudger, TFT]
Tabu List (TL): Cooperator
Simulation output:
(Bully, TFT) 5 [(D, C), (D, D), (C, D), (C, C), (D, C)]
(TFT, Bully) 5 [(C, D), (D, D), (D, C), (C, C), (C, D)]
(Bully, Grudger) 5 [(D, C), (D, D), (C, D), (C, D), (C, D)]
(Grudger, Bully) 5 [(C, D), (D, D), (D, C), (D, C), (D, C)]
(TFT, Grudger) 5 [(C, C), (C, C), (C, C), (C, C), (C, C)]
(Grudger, TFT) 5 [(C, C), (C, C), (C, C), (C, C), (C, C)]
Losing strategy: Bully
Strategy added to Tabu List (TL): none
Winning strategies of Level-1: Defector, Grudger

<u>Level 2 (SS1 vs SS2 from Level 1): Final Round.</u>
Participating Trio: [Defector, Grudger, TitForTat]
Tabu List (TL): Cooperator
Simulation output:
(Defector, TFT) 5 [(D, C), (D, D), (D, D), (D, D), (D, D)]
(TFT, Defector) 5 [(C, D), (D, D), (D, D), (D, D), (D, D)]
(Defector, Grudger) 5 [(D, C), (D, D), (D, D), (D, D), (D, D)]
(Grudger, Defector) 5 [(C, D), (D, D), (D, D), (D, D), (D, D)]
(TFT, Grudger) 5 [(C, C), (C, C), (C, C), (C, C), (C, C)]
(Grudger, TFT) 5 [(C, C), (C, C), (C, C), (C, C), (C, C)]
Losing strategy: Defector
Strategy added to Tabu List (TL): none
Winning strategies: Grudger, TFT

Please note that the score for strategies were calculated based on 'IPD Payoff Matrix' presented in Figure 1.

IV. DISCUSSION

A. Results for Exhaustive Search

For the round-robin with 5 turns and no noise, there are 2 main winners. The defector strategy was the tournament winner 50% of the time and STFT won 40%. This is logical because a defector player would be able to take advantage of all cooperative strategies as there aren't enough rounds for the defector strategy to be punished. Interestingly the average score of STFT was higher than the defector. Since few rounds are played there is a high variance in average scores.

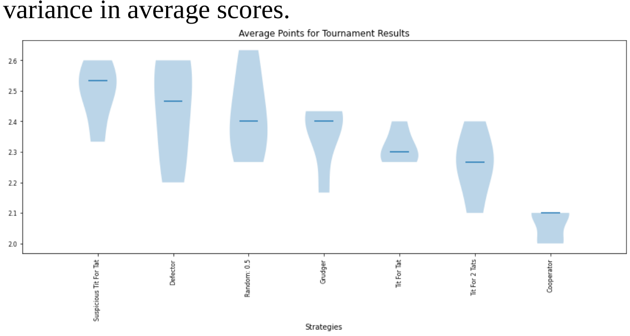

Figure 2: average points for strategies with a tournament of 5 turns

For the round-robin with 15 turns and no noise, there are again 2 main winners. The TFT strategy won the tournament 60 % of the time and TF2T won 40 % of the time with less variance than tit for tat. As more rounds are added, the cooperative strategies are starting to punish defector strategies. With Defector having the worst average score overall. For the round-robin with 250 turns and no noise after 10 simulations, there is 1 clear winner. TF2T seems to be the most dominant with a tournament win rate of 100%.

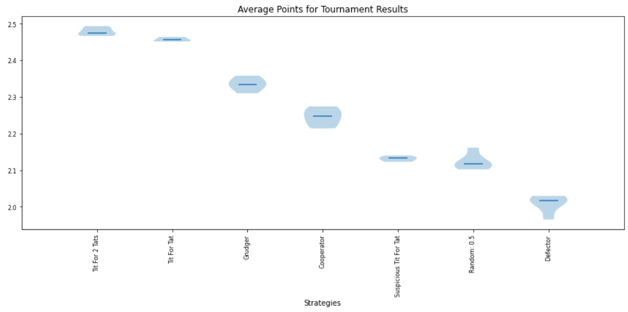

Figure 3: average points for strategies with the tournament of 250 turns and no noise

Comparing figure 2 to figure 3 we can see as rounds are increased the variation decreases and we are receiving a true representation of what strategies are good. A pattern is starting to emerge, as more rounds are added the most effective strategies have the same property of being cooperative. This pattern can be seen visualized by figure 4. The cooperating strategies such as TFT, TF2T, and Cooperator are shown to have better scores with each player on average (the lighter the box the higher the score) when they are playing against each other. In summary when rounds are unknown it seems likely that the best strategies are: TFT, TF2T and Grudger.

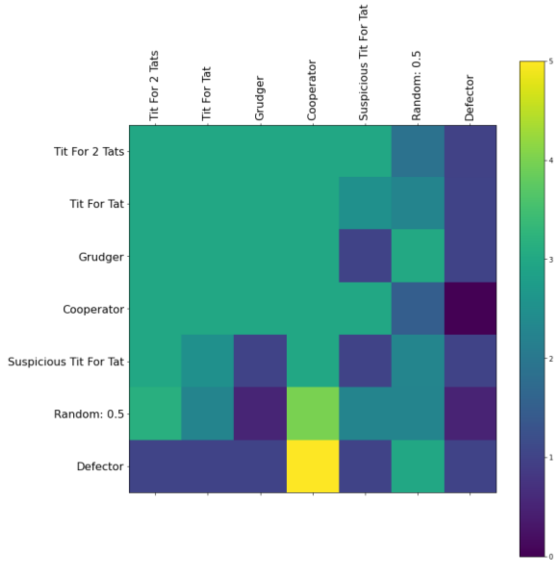

Figure 4: Average score between other strategies

B. *Results for Hill Climbing*

We picked Defector, Random, Cooperator, Grudger, TFT, Alternator, Bully and TF2T in our population of strategies. A number of tournaments are conducted to select the best strategies. In each round, the competing strategies played each other 5 times in one order, and 5 more times by switching the order of play. This is done to provide equal chance to competing strategies. It also prevents the strategies from gaining or losing edge just because they were just used by player taking first turn or vice versa.

TFT, Grudger and TF2T reached the final round and earned equal points resulting in three-way tie. TitFor2Tats and TFT performed consistently well throughout several rounds of the tournament. Cooperator performed worst against bad strategies like Defector and Bully. At the same time, Cooperator performed as good as the competing strategies when the strategies were good like TFT or Grudger. The performance of Random strategy remained inconsistent and thus competing strategies scored inconsistently too. Strategies that cooperated a lot, and at the same time frequently defect when opponent defect succeeded better.

C. *Results for Tabu Search*

We picked Defector, Random, Cooperator, Grudger, TFT, Alternator, Bully and TF2T in our population of strategies. A number of tournaments are conducted to select the best strategies similar to hill climbing. Here we created two subset of strategies that are randomly picked from the population of strategies. Also, a tabu list is maintained that is designed to host inferior strategies. The idea is to skip playing strategies that are already proved to be bad.

Both subsets of strategies take part in phase-1 of tournaments. Subset-1 contains Defector, Random, Cooperator and Grudger. Cooperator strategy performed very badly in subset-1 and was placed in tabu list. Other subset-1 continues to play in tournaments. The winning strategies of subset-1 are Defector and Grudger. Subset-2 contains Cooperator, Grudger, TFT, Alternator and Bully. While arranging tournaments with susbset-2, since Cooperator is part of tabu list, it gets skipped. Rest of the strategies take part in tournament the winning strategies from subset-2 happen to be Grudger and TFT.

The winning strategies of phase-1 move forward to phase-2. In phase-2 final, Defector, Grudger, TFT participated in the tournament. Defector lost in the tournament. Grudger and TFT emerged as the winning strategies. The pattern of behavior of strategies in tabu search are similar to the one observed in hill climbing experiment. The skipping of inferior strategy in tabu list made the process little faster. Here too, strategies that cooperated a lot while returning the favor succeeded well.

V. CONCLUSION

With the modification and application of different optimization techniques on the IPD, we can see that the exhaustive search technique has proved to be highly effective at demonstrating good strategies over a series of simulations. It can further be seen those friendly strategies such as the Tit-for-Tat (TFT) and Tit-for-2-tats (TF2T) showed their dominance against other strategies. Overall, the common theme between all the experiments is the results of friendly strategies succeeding.

VI. FUTURE WORK

After working with the Axelrod library and analyzing the results, we can see that there is a lot of room for future work. One idea involves adding more complex strategies to our simulation to see if our best strategies would still be able to compete. This would allow the best strategies to prove themselves in different simulation environments. Another idea worth looking into would be implementing a genetic algorithm to maximize our optimization techniques. Maximizing our optimization parameters would solidify the accuracy of our test cases. Lastly, we could further analyze the relationships between different strategies, to better understand the effect of friendly strategies to boost each other's scores.

The strategies and methodology used in this experiment can be adopted to find best performing stocks in Wall Street or Toronto Stock Exchange (TSX).

VII. REFERENCE

[1] “Impact of informers on the evolution of cooperation in prisoner's dilemma game”. Chaos, Solitons & Fractals, 149, 2021 111069.Fev Deng, Z. H.Deng, Wang, Z. R.,Wang, H. B. Wang & Y. Huang. https://reader.elsevier.com/reader/sd/pii/S0960077921004239?token=A210B2A82ED711C499DC5620AC0BD56796E1E3F4BE6025F3FCF6E4D2507E587D0F5B753F988139800509516CFC8DDD28&originRegion=us-east-1&originCreation=20220307050256 (accessed Mar. 6, 2022)

[2] "Good influence transmission structure strengthens cooperation in prisoner's dilemma games." The European Physical Journal B 91.12 (2018): 1-12. Liu, Penghui, and Jing Liu https://www.deepdyve.com/lp/springer-journals/good-influence-transmission-structure-strengthens-cooperation-in-zXCMalzMpS (accessed Mar. 6,2022)

[3] “The Next Era of IPD Research: A Systematic Literature Review of The IPD Research Trends 2017-2020”. In IOP Conference Series: Materials Science and Engineering (Vol. 1218, No. 1, p. 012039). IOP Publishing. Arar, A. J., & Poirier, E. (2022) https://iopscience.iop.org/article/10.1088/1757-899X/1218/1/012039/pdf (accessed Mar. 6,2022)

[4] “IPD-inspired framework for measuring stakeholder integration in public-private partnerships. Construction Innovation”. Malaeb, Z., & Hamzeh, F. R. (2021) https://doi.org/10.1108/CI-07-2020-0123 (accessed Mar. 6,2022)

[5] M. Jurisiˇ c, D. Kermek, and M. Konecki, “A review of iterated prisoner’s dilemma strategies,” *CiteSeerX*, Jul. 16, 2012. https://www.egr.msu.edu/~kdeb/papers/k2006002.pdf (accessed Mar. 05, 2022).

[6] A. Developers, “Moran Process — Axelrod 4.12.0 documentation,” *Axelrod*, Apr. 01, 2016. https://axelrod.readthedocs.io/en/stable/index.html (accessed Mar. 06, 2022).

VIII. APPENDIX

<u>Defector</u>
This strategy will always defect against its opponent.
<u>Tit-for-Tat (TFT)</u>
Starts by cooperating the mimic previous action of opponent.
<u>Tit-for-2-Tats (TF2T)</u>
Starts with cooperating and defects after two defects by opponents.
<u>Random</u>
Randomly chooses between cooperating and defecting.
<u>Suspicious-Tit-For-Tat (STFT)</u>
Starts with defect then mimic the previous action by opponent.
<u>Cooperator</u>
Always cooperates.
<u>Grudger</u>
Starts with cooperating but will then defect if opponent defects.
<u>Alternator</u>
Alternates between cooperate and defect.
<u>Bully</u>
Starts by defecting and does the opposite of opponent’s previous move