



Folk Theorem and, in particular, at the
thresholds where cooperation is more
beneficial than defection in the game of a
Prisoners' Dilemma.

Student: Sophie Shapcott

Supervisor(s): Dr Vince Knight and Henry Wilde

Academic Year: 2019/20

School of Mathematics,
Cardiff University

A final year undergraduate project submitted in partial fulfilment of the
requirements for MMORS (Masters in Mathematics, Operational Research
and Statistics) taught programme.

Summary

This project consists of an empirical study into a class of theorems, entitled ‘Folk Theorems’, which are a key part of the repeated games theory. The aims of the project include: an in-depth review of academic literature regarding the topic; the development and execution of a large experiment based on the ‘original’ folk theorem of Friedman [31] with the Iterated Prisoner’s Dilemma; and an analysis of the effects of different tournament characteristics on the p -threshold appearing in the folk theorems. These ideas are extended from a third year assignment in Game Theory, completed by the author.

Firstly, after an introduction to the theory of one-shot and repeated games, a search of the folk theorem literature is provided in Chapter ???. This reveals the vast directions of research in the area for the past fifty years. Many generalisations and refinements of the folk theorem have been analysed since the first written papers in the 1970s. The games for which the notion has been expanded to range from complete information games to games with imperfect private monitoring. However, in certain cases, the strategies used in the proof of these are unstable. Also, in situations where an individual deviator cannot be identified, a much smaller set of payoffs can be achieved, yielding the so-called ‘Anti-Folk Theorems’. More recently, focus has been on the application of the folk theorem to computing and quantum transportation, for example. Finally, it is concluded that, to the best of the author’s knowledge, this is the first study to execute an experiment on the folk theorem of this size.

Following this, a detailed description of how the experiment was set-up and executed is provided in Chapter ??, with justifications to the specific methods and software chosen. After considering the benefits and drawbacks of varying file formats for storing data, it is stated that a relational database, in particular SQLite, would be the most appropriate. This is due to: the existence of libraries in Python enabling the easier access of the data; the general robustness of databases; and its ability to perform out of memory operations. Due to the pre-existing game theoretic libraries, Axelrod and Nashpy, Python was chosen as the

language of implementation and ensuring good software development principles were followed is highlighted as a priority. The volume of data which was planned to be collected meant that remote computing was required and is explained in this chapter. Here, the choice of support enumeration for the calculation of Nash equilibria and the potential issues which may be faced due to degeneracy is also discussed.

The main analysis of the data collected is provided in Chapter 4. An initial analysis discussing the characteristics of the strategies used and the overall summary statistics is detailed, before the p -thresholds are explored. The characteristics of tournaments focused on are: the number of opponents the Defector had and the level of tournament noise included. However, this is concluded as a non-trivial task due to the uncertainty regarding the degeneracy of a large proportion of tournament sets. Also, the inevitability of randomness within the tournaments meant that a lot more data than what was obtained is required. On the other hand, the graphs that were yielded are stated as successful in visualising the folk theorem.

Finally, Chapter 5 details the conclusions of the research before giving recommendations for further work. Amongst these were suggestions on how more characteristics of the tournament set could be studied as well as the potential to predict the p -threshold from the characteristics of the tournaments via regression analysis.

Acknowledgements

First and foremost, I would like to express my sincere thanks and gratitude to my project supervisors Dr Vince Knight and Henry Wilde for their continual advice, support and invaluable ideas throughout the completion of this project. You have always been willing to lend a hand or provide encouragement when it was needed the most. I do not think I could have gotten better supervisors and I apologise for not achieving everything in the plan. You have assisted me through what will probably be the largest project I will ever do!

There are many people who have helped and supported me, both inside and outside university, over the past four years of my undergraduate degree here at Cardiff School of Mathematics. If I was to mention all the names, this would be longer than my whole project! Therefore, this ‘thank you’ is for all those who I have had the pleasure of being taught by and working with during my degree. You have made this journey an interesting and enjoyable one and I will continue to be inspired by the passion all the lecturers have for what they do.

Also, I would like to give special mention to Nikoleta, who, even though is in her final few months of a PhD (probably the most challenging period), found time to have chats with me and make sure I was still operating on this planet. You encouraged me and always had a positive thing to say when I was convinced everything was going to end terribly. So thank you very much Nikoleta, for being such a great friend and best of luck with the completion of your thesis!

Last but most certainly not least, I want to say a huge thank you to my Mum, Dad and Zoe for their continual love and support. To say these have been a challenging four years is an understatement but you were always there for me through it all. If it was not for your belief that I could achieve it I do not think I could have completed this project, let alone the past four years! My Mum, thank you for keeping me fuelled with coffee and biscuits through the long days and for being a shoulder to cry on when necessary (I will pay you back with hair dye, for all those grey ones I caused!). Dad, thank you for making me laugh (and cry)

with your terrible jokes and I promise I will replace all those grains of coffee I had from ‘your’ coffee pot. Finally, my sister Zoe (yes, you are getting a mention because, yes, you did play a key part) thank you for keeping me sane and always making me smile.

To all those who I have forgotten to mention: my sincerest apologies and thank you for all your support.

Contents

Summary	i
Acknowledgements	iii
1 Introduction	1
1.1 An Introduction to Games	1
1.2 Nash Equilibrium for Normal Form Games	4
1.2.1 Brouwer's Fixed Point Theorem	6
1.2.2 Proof of Nash's Theorem (Theorem 1.2)	6
1.3 Repeated Games	8
1.3.1 Finite Repeated Games	8
1.3.2 Infinite Repeated Games	10
1.4 Folk Theorem	12
1.5 Aims of the Project	14
2 Literature Review	16
2.1 First Papers	16
2.2 Games with Complete Information	17
2.3 Games with (Imperfect) Private Monitoring	18
2.4 Games with Communication	19
2.5 Finite Horizon Games	21
2.6 Stochastic and Sequential Games	21
2.7 Anti-Folk Theorems	22
2.8 Evolutionary Stability	23
2.9 Recent Applications	24
2.10 Conclusion	24
3 Methodology and Experiment Setup	25
3.1 Data Collection Algorithm	25
3.2 Databases	30
3.2.1 Types of Database	31

3.2.2	Implementation of the Database	33
3.3	Software Implementation	34
3.4	Remote Computing	35
3.5	Calculating Nash Equilibria	38
3.5.1	Support Enumeration	39
3.5.2	Vertex Enumeration	42
3.5.3	Algorithm Execution Timings	44
3.6	Degeneracy	45
4	Analyses	47
4.1	Initial Analysis	47
4.2	Analysis of the p-Threshold	52
4.2.1	Effects of the Number of Players	55
4.2.2	Effects of Noise	57
4.3	Conclusions and Further Work	59
5	Conclusions and Recommendations	61
5.1	Conclusions	61
5.1.1	Aim 1	62
5.1.2	Aim 2	62
5.1.3	Aim 3	63
5.2	Limitations	63
5.3	Recommendations	65
	References	65
	Appendices	75
A	Computer Report	75
A.1	The Overall Program	75
A.2	Software Development and Remote Computing	77
A.3	Further Details	78

List of Figures

1.1	Graphs to show the row and column players' payoffs against a mixed strategy.	5
1.2	A plot to show the possible payoffs of the game between two players in which A Prisoner's Dilemma is repeated twice.	9
1.3	A plot highlighting the possible individually rational payoffs for the game of A Prisoner's Dilemma.	12
1.4	Original plots obtained which inspired the creation of this project.	15
3.1	An example plot illustrating the p -threshold as indicated by the Folk Theorem	26
3.2	Representation of the algorithm used to collect the data.	26
3.3	DB Browser, a database graphical user interface.	34
3.4	A screenshot of the html report produced when utilising Coverage.	35
3.5	A screenshot of a GitHub pull request which allowed for collaboration between supervisors and author.	36
3.6	A screenshot of the authors connection to Siren via an SSH tunnel.	37
3.7	Representation of how the experiments were run remotely. Note, 'tmux sessions' correspond to emulators of terminals.	37
3.8	Command line code used to retrieve the database from the remote server.	38
3.9	Best response polytope, P , obtained from the payoff matrix given in Equation 3.5.2.	43
3.10	Violinplots of the log timings obtained for the experiment and calculation of Nash equilibria using the three algorithms as implemented in Nashpy.	45
4.1	Example graphs obtained from the experiment.	48
4.2	A plot to show the ratio of stochastic to deterministic strategies randomly chosen throughout the experiment.	50

4.3	A violinplot showing the distribution of the number of equilibria obtained for varying number of players.	51
4.4	Plots to show the p -thresholds for all 1001 sets of tournaments. . .	52
4.5	Plots to show the p -thresholds for all 1001 sets of tournaments. . .	53
4.6	A plot to show the minimum and maximum p -thresholds for those tournaments which had an mean threshold within the range $[0.5, 0.6]$. . .	54
4.7	Example plots of the tournaments where the mean p -threshold was within the range $[0.5, 0.6]$	54
4.8	Plots to show the p -thresholds for all tournaments which led to non-degenerate games.	55
4.9	Plots to show the p -thresholds for all tournaments which led to non-degenerate games.	55
4.10	Violinplots of the thresholds for each number of opponents.	56
4.11	Violinplots of the thresholds for each level of noise.	57
4.12	Observation of one 6-player tournament set through the varying levels of additional noise. There was one stochastic player and 13 out of the 1100 tournaments played yielded potential degenerate games. The opponents here were: <i>Getzler</i> ; <i>Punisher</i> ; <i>Forgiver</i> ; <i>GrudgerAlternator</i> ; and <i>GraaskampKatzen</i>	58
A.1	Representation of how the experiments were run remotely. Note, ‘tmux sessions’ correspond to emulators of terminals.	77

Chapter 1

Introduction

Game Theory is the study of interactive decision making and developing strategies through mathematics [25]. It analyses and gives methods for predicting the choices made by players (those making a decision), whilst also suggesting ways to improve their ‘outcome’ [58]. Here, the abstract notion of utility is what the players wish to maximise (see Chapter 2 in [58] for a detailed discussion on the topic of utility theory or Section 1.3 [88] for a more introductory explanation). One of the earliest pioneers of game theory is mathematician, John von Neumann who, along with economist Oskar Morgenstern, published *The Theory of Games and Economic Behaviour* in 1944 [58]. This book discusses the theory, developed in 1928 and 1940-41, by von Neumann, regarding “games of strategy” and its applications within the subject of economics [84]. Following this, several advancements have been made in the area, including, most notably, John Nash’s papers on the consequently named Nash Equilibria in 1950/51 [62, 61]. Due to the “context-free mathematical toolbox” [58] nature of this subject, it has been applied to many areas, from networks [56, 78] to biology [21, 2]. In this project, the main focus is on a particular class of theorems, within game theory, known as “Folk Theorems” with application to the game of A Prisoner’s Dilemma. These will be defined and discussed in the subsequent sections.

1.1 An Introduction to Games

Consider the following scenario:

Two convicts have been accused of an illegal act. Each of these prisoners, separately, have to decide whether to reveal information (defect) or stay silent (cooperate). If they both cooperate then the convicts are given a short sentence whereas if they both defect then a medium sentence awaits. However, in the

situation of one cooperation and one defection, the prisoner who cooperated has the consequence of a long term sentence, whilst the other is given a deal [46].

This is one of the standard games in game theory known as A Prisoner's Dilemma. It has four distinct outcomes, for the given two player version, which can be represented as a table (see Table ??). Each coordinate (a, b) in the table represents the utility values obtained for each player, where a is the utility value obtained by the row player and b is the utility gained by the column player. These utility values are as given in [7] and are used throughout this project. More formally, the game can be represented as the following matrix:

$$\begin{array}{cc} & \begin{array}{cc} coop & defect \end{array} \\ \begin{array}{c} coop \\ defect \end{array} & \left(\begin{array}{cc} (3, 3) & (0, 5) \\ (5, 0) & (1, 1) \end{array} \right) \end{array}$$

which is known as a *normal form* representation of the game. The following definition is adapted from [58].

In general a *normal form* or *strategic form* game is defined by an ordered triple $G = (N, (S_i)_{i \in N}, (u_i)_{i \in N})$, where:

- $N = \{1, 2, \dots, n\}$ is a finite set of players;
- $S = S_1 \times S_2 \times \dots \times S_n$ is the set of strategies for all players in which each vector $(S_i)_{i \in N}$ is the set of strategies for player i ¹; and
- $u_i : S \rightarrow \mathbb{R}$ is a payoff function which associates each strategy vector, $\mathbf{s} = (s_i)_{i \in N}$, with a utility² $u_i(i \in N)$.

Yet another way of representing this game is as a pair of matrices, A, B , defined as follows:

$$A = \begin{pmatrix} 3 & 0 \\ 5 & 1 \end{pmatrix} \text{ and } B = A^T = \begin{pmatrix} 3 & 5 \\ 0 & 1 \end{pmatrix}$$

This way of defining games allow for the use of calculating payoffs for each player (see Section 1.2).

Before continuing the discussion into the key notions of game theory, it needs to be highlighted that there is an important assumption, which is central to most

¹Since the game of A Prisoner's Dilemma has a finite strategy set for each player $S_i = \{\text{cooperate, defect}\}(i \in N)$, in this project only finite strategy spaces are considered.

²'Utility' is referred to as a player's 'payoff' throughout the remainder of this report.

studies of game theory, entitled *Common Knowledge of Rationality*. This, more formally, is an infinite list of statements which claim:

- The players are rational;
- All players know that the other players are rational;
- All players know that the other players know that they are rational; etc.

Assuming Common Knowledge of Rationality allows for the prediction of rational behaviour through a process entitled *rationalisation* [47]. See section 4.5 in [58] for an alternative explanation of this assumption.

A strategy for player i , s_i , is *strictly dominated* if there exists another strategy for player i , say \bar{s}_i , such that for all strategy vectors $s_{-i} \in S_{-i}$ of the other players,

$$u_i(s_i, s_{-i}) < u_i(\bar{s}_i, s_{-i}).$$

In this case we say that s_i is *strictly dominated* by \bar{s}_i . Here, $s_{-i} = \{s_1, s_2, \dots, s_{i-1}, s_{i+1}, \dots, s_n\}$, i.e. the i th player's strategy has been omitted. The set, S_{-i} , is defined similarly. Looking at the row player's matrix of a Prisoner's Dilemma 1.1 (the first entries in the ordered tuples), it is clear that cooperation is a strictly dominated strategy. Due to the symmetricity of the game, this is also true for the column player. [58]

So far, only the pure strategies, $S_i = \{\text{coop}, \text{defect}\}$, have been discussed, thus the notion of a probability distribution over S_i is now introduced, giving the so-called *mixed strategies* as defined in [58]: Let $G = (N, (S_i)_{i \in N}, (u_i)_{i \in N})$ be a game (with each S_i finite), then a *mixed strategy* for player i is a probability distribution over their strategy set S_i . Define:

$$\Sigma_i := \{\sigma_i : S_i \rightarrow [0, 1] : \sum_{s_i \in S_i} \sigma_i(s_i) = 1\}$$

to be the set of mixed strategies for player i . Hence, observe that the pure strategies are specific cases of mixed strategies, with $\sigma_i = (1, 0)$ for cooperation and $\sigma_i = (0, 1)$ for defection, in the example of a Prisoner's Dilemma.

This leads onto the following definition of a *mixed extension* of a game, taken from [58]: Let G be a finite normal form game as above, with $S = S_1 \times S_2 \times \dots \times S_N$ defining the pure strategy vector set and each pure strategy set, S_i being non-empty and finite. Then the *mixed extension* of G is denoted by

$$\Gamma = (N, (\Sigma_i)_{i \in N}, (U_i)_{i \in N}),$$

and is the game in which, Σ_i is the i th player's strategy set and $U_i : \Sigma \rightarrow \mathbb{R}$ is the corresponding payoff function, where each $\sigma = (\sigma_1, \sigma_2, \dots, \sigma_N) \in \Sigma = \Sigma_1 \times \Sigma_2 \times \dots \times \Sigma_N$ is mapped to the payoff

$$U_i = \mathbb{E}_\sigma(u_i(\sigma)) = \sum_{(s_1, s_2, \dots, s_N) \in S} u_i(s_1, s_2, \dots, s_N) \sigma_1(s_1) \sigma_2(s_2) \cdots \sigma_N(s_N) \quad (1.1)$$

for all players $i \in N$.

1.2 Nash Equilibrium for Normal Form Games

As mentioned above, mathematician, John Nash, introduced the concept of an equilibrium point and proved the existence of mixed strategy Nash Equilibria in all finite games. These notions are central to the study of game theory [58] and hence, in this section, Nash's concepts will be defined and proved in detail.

Firstly, before the definition of a Nash equilibrium, the idea, as given in [58] of a *best response* is introduced: For a game $G = (N, (S_i)_{i \in N}, (u_i)_{i \in N})$, the strategy, s_i , of the i th player is considered a *best response* to the strategy vector s_{-i} if $u_i(s_i, s_{-i}) = \max_{t_i \in S_i} u_i(t_i, s_{-i})$.

This leads onto the main definition of the section:

Definition 1.2.1. Given a game $G = (N, (S_i)_{i \in N}, (u_i)_{i \in N})$, the vector of strategies $s^* = (s_1^*, s_2^*, \dots, s_n^*)$ is a *Nash equilibrium* if, for all players $i \in N$, s_i^* is a best response to $s_i^* \in N$. [58]

In other words, s^* is a Nash equilibrium if and only if no player has any reason to deviate from their current strategy s_i^* .

Recall that, in Section 1.1, for any player in A Prisoner's Dilemma, defection dominated cooperation. This leads to the following observation:

The strategy pair (Defect, Defect), is the unique Nash equilibrium for A Prisoner's Dilemma, with a payoff value of 1 for each player. [58]

This can be visualised as followed: Assume the row player uses the following mixed strategy, $\sigma_r = (x, 1 - x)$, i.e. the probability of cooperating is x and the probability of defecting is $1 - x$. Similarly, assume the column player has the strategy, $\sigma_c = (y, 1 - y)$. The payoff obtained for the row and column player, respectively, is then:

$$A\sigma_c^T = \begin{pmatrix} 3 & 0 \\ 5 & 1 \end{pmatrix} \begin{pmatrix} y \\ 1 - y \end{pmatrix} = \begin{pmatrix} 3y \\ 4y + 1 \end{pmatrix},$$

$$\sigma_r B = \begin{pmatrix} x & 1-x \end{pmatrix} \begin{pmatrix} 3 & 5 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 3x & 4x+1 \end{pmatrix}$$

Plotting these gives the following: From Figure 1.1 it is clear that, regardless of

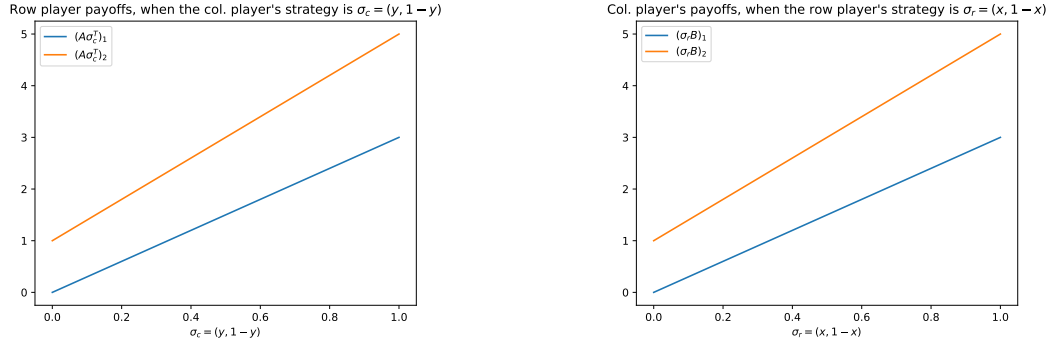


Figure 1.1: Graphs to show the row and column players' payoffs against a mixed strategy.

the strategy played by the opponent, defection is indeed the only rational move for one to play. Thus, both players have no incentive to deviate if and only if both play the strategy $\sigma = (0, 1)$, i.e. defection for every single game of A Prisoner's Dilemma.

On the other hand, consider, for example, a game with no dominated strategies³:

$$A = \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}, \quad B = \begin{pmatrix} -1 & 1 \\ 1 & -1 \end{pmatrix}$$

Are Nash Equilibria guaranteed to exist? This result is given in the next theorem, taken from [61], Nash's second paper on equilibria in games.

Theorem 1.2.1. Every finite game has an equilibrium point.

The proof of Theorem includes the use of a *fixed point theorem* and thus, a short sub-section regarding one such result is given, for completeness, before providing a formal proof of 1.2.

³The game highlighted here is another standard used in game theory entitled *Matching Pennies*. Moreover, it is what is defined as a *zero-sum* game. Interested readers are encouraged to read Example 4.21 of [88] for an introduction to the game and Section 4.12 in [58] for an explanation of zero-sum games.

1.2.1 Brouwer's Fixed Point Theorem

Brouwer's Fixed Point Theorem is a result from the theory of topology. Named after the Dutch mathematician, L.E.J. Brouwer, it was proven in 1912 [19]. However, before stating this notion, a few conditions regarding the properties of sets are recalled.

The following three definitions appear as in [90, 9, 89] respectively:

Definition 1.2.2. A set $X \subseteq \mathbb{R}^d$ is called *convex* if it contains all line segments connecting any two points $\mathbf{x}_1, \mathbf{x}_2 \in X$.

Definition 1.2.3. An *open cover* of a set $S \subset X$, a topological space, is a collection of open sets $A_1, A_2, \dots \subset X$ such that $A_1 \cup A_2 \cup \dots \supset S$, that is, the union of the open sets contain S.

Definition 1.2.4. A subset $S \subseteq X$, a topological space, is called *compact* if, for each open cover of S , there is a finite sub-cover of S.

The presentation of Brouwer's Fixed Point Theorem is now given as in [58]

Theorem 1.2.2. Let $X \subseteq \mathbb{R}^n$ be a non-empty convex and compact set, then each continuous function $f : X \rightarrow X$ has a fixed point.

In other words if X and f satisfy the conditions given above then there exists a point $x \in X$ such that $f(x) = x$.

Since this project is regarding game theory, rather than topology, the proof to the above theorem is omitted. However, the interested reader is referred to [] for an in-depth consideration into the theory of topology.

1.2.2 Proof of Nash's Theorem (Theorem 1.2)

The proof provided here is adapted from the original, by Nash, as presented in [61] (with extra notes adapted from [58]). According to [58], the general idea here is to define a function, which satisfies the conditions required for Theorem 1.2.2, using the payoff functions on the set of mixed strategies. Then by identifying each equilibrium point with a fixed point of the function, the required result is obtained.

Proof. Firstly, a brief restatement of the notation needed is provided for clarity. Let $G = (N, (S_i)_{i \in N}, (u_i)_{i \in N})$ be a finite game with mixed extension $\Gamma = (N, (\Sigma_i)_{i \in N}, (U_i)_{i \in N})$. Here, N denotes the number of players; $S = S_1 \times S_2 \times \dots \times S_N$ is the set of pure strategies for all players, with $(S_i)_{i \in N}$ the pure strategy set for player i ; Σ is defined similarly but relating to mixed strategies; and

$U_i : \Sigma \rightarrow \mathbb{R}$ are the payoff functions as given in equation 1.1. Recall that σ_{-i} is used to denote the strategy choice of all players *except* the i th player.

Let $\sigma = (\sigma_1, \sigma_2, \dots, \sigma_N)$ be a tuple of mixed strategies and $U_{i,t}(\sigma)$ be the i th player's payoff if they changed to their t th pure strategy and all other players continue to use their mixed strategy. Now, define function $f : \Sigma \rightarrow [0, \infty)$ such that

$$f_{i,t}(\sigma) = \max(0, U_{i,t}(\sigma) - U_i(\sigma)) \quad (1.2)$$

and also let

$$\sigma'_i = \frac{\sigma_i + \sum_t f_{i,t}(\sigma) s_i^t}{1 + \sum_t f_{i,t}(\sigma)} \quad (1.3)$$

be a modification of each $\sigma_i \in \sigma$, with $\sigma' = (\sigma'_1, \sigma'_2, \dots, \sigma'_N)$. In words, this modification increases the proportion of the pure strategy t used in σ_i if the payoff gained by the i th player is larger when they replace their mixed strategy by t . Else, it remains the same if doing this decreases their payoff as $f_{i,t}(\sigma) = 0$ in this case. Note, the denominator ensures that the ending vector is still a probability distribution by standardising.

The aim is to apply Theorem 1.2.2 to the mapping $T : \sigma \rightarrow \sigma'$ and show that it's fixed points correspond to Nash equilibria. Thus, firstly compactness and convexity of the set σ is shown along with continuity of the function f .

Observe that each σ_i can be represented by a point in a simplex in a real vector space with the vertices given by the pure strategies, s_i^t . Therefore, it follows that the set Σ_i is convex and compact. Using the result, *If $A \subseteq \mathbb{R}^n$ and $B \subseteq \mathbb{R}^m$ are convex compact sets then the set $A \times B$ is a convex compact subset of \mathbb{R}^{n+m}* , as highlighted in [58] gives the convexity and compactness of the set Σ , the cross product of all Σ_i s.

The continuity of the function f depends upon the continuity of the payoff functions U_i . As given in [58], this is shown by first proving that the U_i are multilinear functions in the variables $(\sigma_i)_{i \in N}$ and then applying the fact that any multilinear function over Σ is a continuous function.⁴ The result then follows.

Hence, by Theorem 1.2.2, the mapping T must have at least one fixed point. The proof is concluded by showing that any fixed points of T are Nash equilibria.

Suppose σ is such that $T(\sigma) = \sigma$. Then, the proportion of s_i^t used in the mixed strategy σ_i must not be altered by T . Therefore, in σ'_i , the sum $\sum_t f_{i,t}(\sigma)$, in the denominator, must equal zero. Otherwise, the total sum of the denominator will be greater than one, decreasing the proportion of s_i^t . This implies that for

⁴For a detailed consideration of the continuity of the payoff functions see [58], pages 148-149.

all pure strategies q , $f_{i,q}(\sigma) = 0$. That is, player i can not improve their payoff by adopting any of the pure strategies. Note, this is true for all i and q by definition of $T(\sigma) = \sigma$ and thus no player is able to improve their payoff. By definition 1.2.1, this is exactly the conditions of a Nash equilibrium.

Now assume σ is a Nash equilibrium. Then, by definition, it must be that $f_{i,q}(\sigma) = 0$ for all pure strategies q for all players, $i \in N$. Note, if $f_{i,q}(\sigma) \neq 0$, then the i th player would benefit from changing their strategy to the pure strategy q , which violates the condition for a Nash equilibrium. From this it follows that $T(\sigma) = \sigma$, that is, σ is a fixed point of T . This concludes the proof.

□

1.3 Repeated Games

The folk theorems studied in this project are a consequence of games which are repeated several times (not just once). Thus, before discussing the main ideas, the theory of both finitely- and infinitely- repeated games is presented.

Firstly, a couple of alterations to the terminology used in previous sections is redefined for conciseness and to be consistent with the literature [1]. What was known as a ‘game’ will become known as a *stage game* to highlight the fact that a one-off game is being considered. Also, what was defined previously as a ‘strategy’ will now be referred to as an *action* to differentiate it from a strategy of a repeated game (see Section 1.3.1).

1.3.1 Finite Repeated Games

According to [50], a T -stage repeated game, $T < \infty$ is when the stage game, G , is played T times, over discrete time intervals. Each i th player has a strategy based on previous ‘rounds’ of the game and the payoff of a repeated game is calculated as the total sum of the stage game payoffs.

Prior to giving the notion of a strategy in a repeated game, the idea of *history*, within the context of repeated games, is provided. The *history*, $H(t)$ of a repeated game is the knowledge of previous actions of all players up until the t th stage game, assumed to be known by player i for all $i \in 1, \dots, N$. Note that, when $t = 0$, $H(0) = (\underbrace{\emptyset, \emptyset, \dots, \emptyset}_{N \text{ times}})$, since no stage games have yet been played.

As given in [49], a *strategy* of a T -stage repeated game is defined to be a mapping

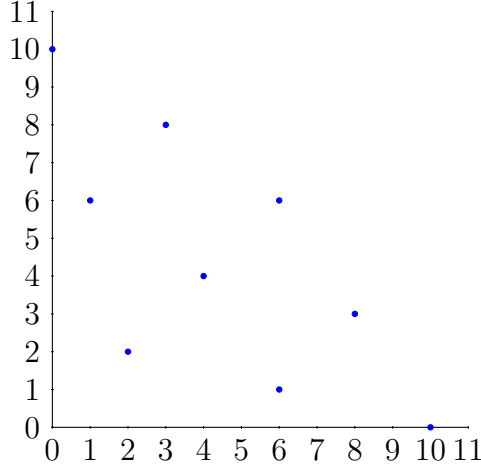


Figure 1.2: A plot to show the possible payoffs of the game between two players in which A Prisoner's Dilemma is repeated twice.

from the complete history so far to an action of the stage game, that is

$$\tau_i : \cup_{t=0}^{T-1} Ht \rightarrow a_i.$$

Here, $H(t)$ is the history of play as defined above and a_i is the i th player's action of the stage game.

Consider, for example, the environment in which the stage game of a Prisoner's Dilemma is repeated each time. This is known as the *Iterated Prisoner's Dilemma* (IPD) and has been a popular topic of research for many years see Chapter ?? . Note that the objective here is to maximise your payoff. The player:

No matter what my opponents play, I will always defect,

commonly known as the 'Defector' has the following strategy mapping:

$$\tau_i : \cup_{t=0}^{T-1} Ht \rightarrow a_i,$$

where $a_i = D$ for all time periods $\tau \geq 0$. Other common IPD strategies include:

Cooperator - *No matter what my opponents play, I will always cooperate;*

Random - *I will either cooperate or defect with a probability of 50%; and*

Tit For Tat - *I will start by cooperating but then will duplicate the most recent decision of my opponent.*

Figure 1.2 shows the possible payoffs obtained in a 2-stage repeated IPD with two players:

Now, what about Theorem 1.2? Are there any Nash equilibria in repeated games?

In fact, it can be proven that there exist many equilibria in repeated games [30]. The next result, adapted from [49] guarantees at least one Nash equilibria.

Theorem 1.3.1. Consider a T -stage repeated game with $G = (N, (S_i)_{i \in N}, (u_i)_{i \in N})$ as the stage game, $0 < T < \infty$. Define by $\sigma^* = (\sigma_1^*, \sigma_2^*, \dots, \sigma_N^*)$, a stage Nash equilibrium of G . Then the sequence in which σ^* is continuously played is a Nash equilibrium of the T -stage repeated game.

Proof. Since σ^* is a stage Nash equilibrium, it is, in particular, a Nash equilibrium of the T th stage game. Thus, no player has any reason to deviate here. But then σ^* was also played at the $(T - 1)$ th stage also, meaning there is still no reason to deviate. Therefore, continuing via backwards induction gives the required result. \square

Hence, for the T -stage IPD, every player executing the Defector strategy yields a Nash equilibrium. However, do there exist equilibria for which selecting the action ‘cooperate’ is more beneficial?

In preparation to answer this, the next Subsection 1.3.2 discusses the case when $T \rightarrow \infty$ and results linked to *infinitely repeated games*.

1.3.2 Infinite Repeated Games

The Folk Theorem discussed in Section 1.4 considers a stronger notion of equilibria known as *subgame perfect equilibria*. In order to fully understand this notion, a new representation of games is introduced.

Extensive Form Games

According to [58], an *extensive form game* is given by the ordered vector $\Gamma = (N, V, E, x_0, (V_i)_{i \in N}, O, u)$ where N is a finite set of players; (V, E, x_0) is a *game tree*⁵; $(V_i)_{i \in N}$ is a partition of the set $V \setminus L$, where L is the set of all leaves, or terminal points, of the game tree; O is the set of outcomes for the game; and u is a function which maps each leaf in L to an outcome in O .

This leads on to the following definition, adapted from [88]:

A player’s *information set* is a subset of the nodes in a game tree where:

- Only the player concerned is deciding;

⁵The triple (V, E, x_0) is defined as a *tree* if the set of vertices, V , and the set of edges, E , create a *directed graph* (that is, each element in E is an ordered tuple) and x_0 represents the root, or starting node, of the graph.

- This player is not aware of which node has been reached except that it is definitely one of the elements found in this set.

In Figure ??, the extensive form representation of the game A Prisoner's Dilemma is provided. Here, only two players are considered and any information sets are represented by a dashed line. Note, any normal form game can be represented as an extensive form game.

According to [88], a *subgame* is a sub-graph of the game tree such that:

- The sub-graph begins at a decision node, say x_i ;
- This node, x_i , is the only element contained in its information set;
- The sub-graph contains all of the decision nodes which follow x_i .

This leads to the following definition of *subgame perfect equilibria*, adapted from [88].

Definition 1.3.1. A *subgame perfect equilibrium* is a Nash equilibrium which satisfies the condition that the strategies played define a Nash equilibrium in every subgame.

Hence the strategy defined in Theorem 1.3.1 is a subgame perfect equilibrium. A few final definitions are now highlighted before introducing the Folk Theorem.

Final Definitions Needed

Now, in order to be able to discuss the payoffs of strategies in infinite games; the notion of a *discounted payoff* is introduced. This is defined in [45] as:

$$V_i(\sigma) = \sum_{t=1}^{\infty} \delta^{t-1} U_i(\sigma),$$

where the discount factor, δ , can be thought of as the probability that the game continues. That is, the probability that another stage game will be played. Using this, the *average payoffs*, per stage game, can be defined by:

$$\frac{1}{\bar{T}} V_i(\sigma) = (1 - \delta) V_i(\sigma),$$

where $\bar{T} = \frac{1}{1-\delta}$ is the average length of a game [45].

Finally, Figure 1.3 shows those payoffs which are individually rational for a two player version of A Prisoner's Dilemma. In general, an *individually rational payoff* is an average payoff which exceeds the payoffs obtained in the stage Nash equilibria for all players [45]. Note, often the Nash equilibrium payoffs are not the optimal payoff which players could achieve.

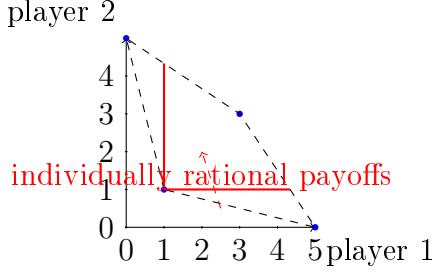


Figure 1.3: A plot highlighting the possible individually rational payoffs for the game of A Prisoner's Dilemma.

In the next section (Section 1.4), the main theorem of this project will be stated and proved.

1.4 Folk Theorem

According to [88], the class of theorems known as Folk Theorems are so-called because the result was well-known before a formal proof was provided. In general, these theorems state that players can achieve a better payoff than the Nash equilibrium (if the Nash equilibrium payoff is not optimal) when the stage game is repeated many times and the probability of the game continuing is large enough.

It is believed that Friedman, 1971 ([30]) was one of the first to publish a formal proof to the widely accepted Folk Theorem [1]. Thus, the presentation of the statement and proof given here is adapted from [30] as well as from [45].

Before stating the theorem, the list of assumptions which Friedman [30] requires the infinite repeated game to satisfy is given.

1. The mixed action sets, Σ_i are compact and convex for all $i \in N$.
2. The payoff functions, $U_i : \Sigma \rightarrow \mathbb{R}$, are continuous and bounded for all $i \in N$.
3. The $U_i(\sigma)$ s are quasi-concave⁶ functions of σ for all $i \in N$.
4. If $U'_i \leq U''_i$, for all $i \in N$ and $U'_i, U''_i \in H$, then, for all $U'_i \leq U \leq U''_i$, $U \in H$. Here, H is defined to be the set of feasible payoffs, $\{U(\sigma) : \sigma \in \Sigma\}$, where $U(\sigma) = (U_1(\sigma), U_2(\sigma), \dots, U_N(\sigma))$.

⁶According to [80], a real-valued function f , defined on a convex subset $C \subset \mathbb{R}^n$, is *quasi-concave* if for all $\alpha \in \mathbb{R}$, the set $\{x \in C : f(x) \geq \alpha\}$ is convex.

5. H^* is concave, where $H^* \subset H$ denotes the set of all Pareto optimal pay-offs.⁷
6. All stage games are identical in the infinitely repeated game.
7. The discount parameter, δ , is equal in all time periods.
8. The stage game has a unique Nash equilibrium.
9. The Nash equilibrium is not Pareto optimal.

Note, Friedman [30] later goes on to prove that assumptions six to nine can be removed with only a small effect on the result. However, since the only game being studied in this project is the IPD (which satisfies all the above assumptions), this generalisation will be omitted. Thus, only the proof of the original theorem will be provided.

Theorem 1.4.1. If the above assumptions are all satisfied for the given infinite repeated game, then for any individually rational payoff V_i (greater than the payoff yielded by the stage Nash equilibrium σ^{ne}), there exists a discount parameter δ^* such that for all δ_i , $0 < \delta^* < \delta_i < 1$ there is a subgame perfect Nash equilibria with payoffs equal to V_i .

Proof. Consider the set of all actions which yield greater payoffs than the Nash equilibrium, denoted by:

$$B = \{\sigma : \sigma \in \Sigma, U_i(\sigma) > U_i(\sigma^{ne}), i = 1, 2, \dots, N\}$$

and define the following trigger strategy:

$$\sigma_{i1} = \sigma'_i, \sigma_{it} = \begin{cases} \sigma'_i, & \text{if } \sigma_{j\tau} = \sigma'_j \text{ } j \neq i, \tau = 1, 2, \dots, t-1, t = 2, 3, \dots \\ \sigma_i^{ne}, & \text{otherwise,} \end{cases} \quad (1.4)$$

where $\sigma'_i \in B$. In words, the i th player will choose σ'_i unless any other player does not play σ'_j , in which case they continue by playing their Nash equilibrium action, σ_i^{ne} . Now, by definition, the strategy in 1.4 is an equilibrium of the repeated game if

$$\sum_{\tau=0}^{\infty} \delta_i^\tau U_i(\sigma'_i) > U_i(\sigma'_{-i}, t_i) + \sum_{\tau=1}^{\infty} \delta_i^\tau U_i(\sigma^{ne}), \quad i = 1, 2, \dots, N,$$

which can be rearranged to

$$\frac{\delta_i}{1 - \delta_i} [U_i(\sigma') - U_i(\sigma^{ne})] > U_i(\sigma'_{-i}, t_i) - U_i(\sigma'), \quad i = 1, 2, \dots, N,$$

⁷Friedman defines a *Pareto optimal payoff* as a point in the payoff space $U(\sigma^*)$ which satisfies the conditions: $\sigma^* \in \Sigma$ and $U_i(\sigma^*) > U_i(\sigma)$ for all $i \in N$

where $U_i(\sigma'_{-i}, t_i) = \max_{\sigma_i \in \Sigma_i} U_i(\sigma'_{-i}, \sigma_i)$, $t_i \in \Sigma_i$.

To check if this strategy is indeed a best response to all others players, who are executing the same strategy in 1.4, for the i th player consider their alternatives. They have two options: either the i th player executes the strategy in question, or they play the strategy in which $\sigma_{i1} = t_i$ and then $\sigma_{i\tau} = \sigma^{ne}$ will be the best response as every other player will convert to $\sigma_{j\tau} = \sigma^{ne}$, for all $\tau > 1$. Note that any other strategy is weakly dominated by one of these two, since playing t_i in any other stage $\tau \neq 1$ will yield less gains as these stages have increased discounting.

Now, if the discounted loss from playing the Nash equilibria,

$$\frac{\delta_i}{1 - \delta_i} [U_i(\sigma') - U_i(\sigma^{ne})], \quad (1.5)$$

is greater than the gain achieved by playing t_i against σ'_{-i} , then the rational strategy choice for player i , assuming all other players are executing 1.4, is to play 1.4.

Observe, as the discount parameter, δ , tends to one from below, the discounted loss in 1.4 tends to infinity. However, the gain obtained from playing t_i , that is, $U_i(\sigma'_{-i}, t_i) - U_i(\sigma')$ is finite. Thus, for all $\sigma'_i \in B$ there exists a $\delta^* \in (0, 1)$ such that for all $\delta_i > \delta^*$, the strategy 1.4 is optimal against the same strategy for all players $j \neq i$. Therefore, if the conditions are true for all players $i = 1, 2, \dots, N$, the strategy $(\bar{\sigma}_1, \bar{\sigma}_2, \dots, \bar{\sigma}_N)$, where $\bar{\sigma}_i$ denotes 1.4, yields a Nash equilibrium.

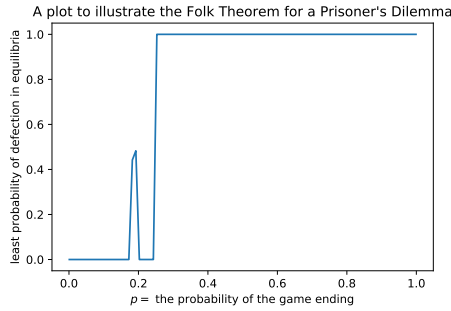
Finally, by construction, the strategy 1.4 is indeed a subgame perfect equilibrium. \square

1.5 Aims of the Project

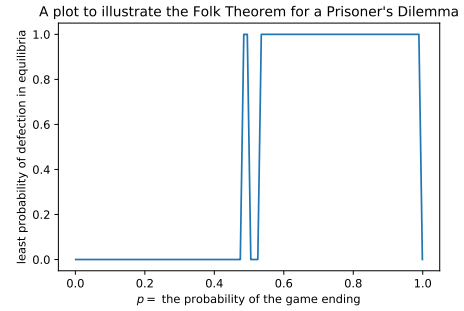
This project stemmed from an initial idea used in Game Theory Coursework written by the author. In this coursework, regarding Nash equilibria and repeated games, the two graphs, as presented in Figure 1.4, were obtained.

These two graphs were obtained using a similar method to this project and so see Chapter ?? for the details. The plots show the least probability of defection obtained in the Nash equilibria of the game. The game matrices were defined by repeating an IPD tournament 100 times for the Defector strategy, and three other opponents, with 100 distinct probabilities of the game ending.

From Figure 1.4, it can be seen that there appears to be a ‘clear probability’ of the game ending at which the least probability of defection jumps from zero



(a) A plot of the least probabilities of defection when playing against the strategies: Cooperator, TitForTat and Random. Here, the p -threshold is approximately 0.25.



(b) A plot of the least probabilities of defection when playing against the strategies: Winner21, AntiTitForTat and OmegaTFT. Here, the p -threshold is around 0.5.

Figure 1.4: Original plots obtained which inspired the creation of this project.

to one. (Throughout the rest of this project, this ‘clear probability’, p , will be referred to as the *p-threshold* of a game.) However, what was most intriguing was that the two different games obtained had a different p -threshold (approximately 0.25 for Figure 1.4a but for Figure 1.4b the threshold appears at around 0.5). This inspired the idea to investigate whether there are any specific characteristics of an IPD tournament that affect the value of this threshold. That is, does the number or the stochasticity of players, for example, cause the probability of a game ending for which the least probability of defection jumps to increase or decrease?

Therefore, the aims of this project are as follows:

1. To look thoroughly into the recent (and past) literature of research already produced in the areas of Folk Theorems and the IPD;
2. To execute a large experiment involving many tournaments of the IPD with different types / numbers of players to obtain graphs similar to those in Figure 1.4;
3. To perform analyses on where the p -thresholds seem to lie and whether it is affected by the different environments of the tournaments (that is, differing number of players, stochasticity within the tournament itself and the players, etc.); and
4. To explore other ‘Folk-like’ Theorems and perform similar experiments and analyses.

Chapter 2

Literature Review

The folk theorems are a class of results which generally state that for repeated games, any feasible and individually rational payoff vector can be achieved as a subgame perfect equilibrium if the players are patient enough [55]. The origin of these theorems is unknown however written proof and research involving these ideas first appeared in the 1970s [31, 6, 74]. Since then, many generalisations and refinements of the ideas have been explored, for differing games, including: games with private monitoring [60, 39, 70], sequential games [36, 15, 91] and games of complete information [13, 1, 14], to name a few. Due to the identification of further equilibria (as compared with the stage game), which are key in predicting future behaviour, these theorems have been commented on as ‘fundamental’ in the theory of non-cooperative games [39, 53]. On the other hand, the majority of the strategies used in the proofs assume the identification of individual deviators [59], which may not be realistic in certain situations. Hence, an area of research on the so-called ‘anti-folk theorems’ was introduced [59, 95, 70]. Folk theorems appear to still be an active area of research today [40, 69, 86] with many differing applications. Therefore, in this chapter, a review of the literature on this topic is provided with papers ranging from the ‘original’ ideas in the 1970s to applications of the theorems in 2020.

2.1 First Papers

[31] is often cited as the earliest work on Folk-type theorems [1]. In [31], infinitely repeated games with discounting are considered. In particular, focus is on a class of strategies, now known as *grim trigger*, which are used to prove that, for any feasible and individually rational payoff vector, there exists a discount parameter such that a subgame perfect equilibrium with payoffs equal to that

vector exist. This is first shown under the constraints (in addition to those made on the strategy and payoff spaces) identical stage games, constant discount parameter and unique Nash equilibrium that is not Pareto optimal. However, a more general result is then given which removes these restrictive conditions. [31] uses the application of oligopoly throughout in addition to introducing the notation of ‘temptation’. He motivates this through the explanation that ‘threat’ is no longer “credible” since players are unable to communicate in non-cooperative games. On the other hand, ‘temptation’ is said to be analogous to ‘threat’.

In contrast to this, [6] presents folk theorems for infinite games without discounting, assuming payoffs take a “limiting average form”. This choice is justified by stating that, as the discount rate approaches zero, the limit of the discounted sum behaves similarly to the limiting average payoff form. However, trigger strategies are still required in the proof. For simplicity, two player games are assumed throughout. Two folk theorems are proven in this paper with the latter one discussing subgame perfect equilibria similar to [31], whilst the former is a more generic version of the theorem. [6] discuss that, though the generic version of the theorem does exist, the subgame perfect equilibrium points allow for more ‘believable’ behaviour. They conclude by considering an example in which payoffs are discounted. A similar approach is taken in [74], with the statement of a folk theorem for infinite games with no discounting and the existence of subgame perfect equilibria. The only difference being the use of an “overtaking criterion” evaluation relation instead of a “limit of the mean criterion” evaluation relation as in [6].

2.2 Games with Complete Information

[1, 14] both study folk theorems of games assuming complete information. That is, all players have common knowledge of payoff functions, strategies etc. [1] focuses on the necessary and sufficient conditions required for a folk theorem proof. They state that feasibility and individual rationality of the payoff vector are necessary conditions and also approximately sufficient conditions for a payoff to be in the equilibrium. This is followed by a discussion on the full dimensionality constraint first introduced by [33] and often used in proving folk theorems. [1] highlight that the requirement of the convex hull, of the set of feasible payoffs, to be equal to the number of players is a sufficient condition. This is motivation for the main result of the paper; the introduction of a ‘non equivalent utilities’ condition, which is proved to be sufficient and almost necessary for a folk theorem. The condition, as indicated here, is weaker than the aforementioned full

dimensionality condition. It only requires “no pair of players have equivalent utility functions”.

Similarly, [14] proves a folk theorem for complete information games with sub-game perfect equilibria however, instead of exponential discounting (as in [1, 33]), he assumes discounting is present-biased. That is, the discount function implies a player is more willing to alter an event in the future than altering an event today. The folk theorem is proved in cases of where the player is time consistent (prefers to maximise their initial preferences) and time inconsistent (prefers to maximise current preferences) and the notion of ‘patience’ is taken to be the sum of all discount factors. Also considering time consistent and inconsistent players [55] discusses folk theorems with respect to time dependent discounting. In contrast to [14], the long-term characterisation of ‘patience’ is taken as discount factors at all stages uniformly converge to one. Motivation for the study of time dependent discounting is explained by empirical studies which seem to suggest that a player’s time is unstationary, rather than the stationarity assumed in most discounted game models.

2.3 Games with (Imperfect) Private Monitoring

In the late 1990s attention turned to games with imperfect monitoring. That is, a player’s actions can no longer be observed accurately, instead public or private signals are detected [26]. Games with (imperfect) public monitoring were the first to be considered [60]. More recent studies in this area include [44, 20]. [44] states that repeated games with public monitoring give a suitable model for studying long term relationships. They state intuition suggests that cooperation becomes easier to sustain as action observability improves. However, [44] continues to show that this intuition is false, by considering a repeated public monitored game which satisfies “the limit perfect public equilibrium payoff set can achieve full efficiency asymptotically as public information becomes less sensitive to hidden actions.” They give an example which violates the sufficient condition given in Fudenberg et al. (1994) [FIND THIS PAPER] yet a folk theorem can be obtained. This paper mainly focuses on the Prisoner’s Dilemma but, similarly to [1], they state an example which satisfies the folk theorem without the full dimensionality condition.

[20] consider the notion of a robust equilibrium to incomplete information; that is, if the equilibrium yielded is near the original equilibrium for all perturbed games consisting of a small independent and identically distributed information shock. Conversely to [31], they discuss the implication of grim trigger strategies

not being robust when the aim is to sustain cooperation. A folk theorem is proved in robust equilibria for games of public monitoring, however [20] highlights that a much stronger condition is needed in comparison to Fudenberg et al. (1994) [FIND THIS PAPER] full dimensionality requirement.

As of 2004, repeated games with private monitoring was a relatively new area of study [60]. In this paper, conditional independence of signals is assumed in order to show a folk theorem for the IPD. Applications to duopolies are then described. Another study to consider a folk theorem for the IPD is [29]. Using a similar definition of robustness to [20], [29] proves that for a discounted Prisoner's Dilemma with private monitoring technologies, a folk theorem with robust equilibrium strategies can be obtained. In particular, they consider almost-perfect private monitoring and a limit folk theorem (for sequential equilibria) follows. In a similar paper, [28] introduce the idea of "belief free" equilibrium strategies; a property which implies the belief of an opponent's history is not required when obtaining a best response. They use these strategies in proving a folk theorem for the two-player Prisoner's Dilemma however, for general games, the set of belief free payoffs is not large enough to provide a folk theorem. Moreover, [28] highlight that for a larger number of players the calculation of the payoff set becomes significantly harder.

In contrast to this, [39] discusses a folk theorem using strategies that, although are not belief free, still make beliefs "irrelevant" at the start of each T-block of stage games. Their result is much more generic than [28] since it applies to N-player finite games, under the assumption of full dimensionality with private but almost-perfect monitoring. The T-block strategies mentioned above are modified in [93, 94] such that they "can support any vector in the belief free equilibrium payoff set". This modification results in belief free equilibrium strategies. [93, 94] generalise the results from the two-player Prisoner's Dilemma in [28] to the corresponding N-player game.

2.4 Games with Communication

Academics introduced games with communication to deal with the complications faced with imperfect monitoring. For example, games with public monitoring can obtain a folk theorem under weaker assumptions than those given in Fudenberg et al. (1994) [FIND THIS PAPER] if communication is introduced [43]. In this paper, communication is a message taken from the set of possible actions which the players give simultaneously after choosing an action and observing a signal. [43] prove a folk theorem for symmetric games with four or more players without

the assumption of the number of signals relevant to the number of actions.

Regarding private monitoring, [65] claims repeated games are “very difficult to analyse without communication”. Thus examples of papers, proving folk theorems for these games with communication, include [32, 65, 54]. [32] proves a ‘Nash threats’ folk theorem, similar to [31], in the case of almost public information games without independent signals for two players. In this paper, communication is defined in the form of announcements where each set of announcements is the same for all players. They justify the decision to only consider two player games by highlighting that, although the results can be generalised, in certain cases it can be seen as advantageous to have additional players. Similarly, [65] proves a Nash threats folk theorem for private monitoring games with communication. He increases the number of environments where the folk theorem is applicable through developing further the idea of “delayed communication”, as given in Compte [FIND THIS PAPER]. This paper, in addition, uses the assumptions of correlated private signals and each player’s deviation from strategy is statistically identifiable from the other players’ signals. [54] also considers a model of private monitoring and communication within games. He has the aim of increasing the number of applicable environments for Kandori and Matsushima (1998) [FIND THIS PAPER] frequent communication folk theorem. The paper states that Kandori and Matsushima (1998) [FIND THIS PAPER] assumed only private signals were publicised in this theorem but if other information was useful and communication was free and legal then players would also want to share their actions. This motivates the reasoning of the paper. However, assumptions of both full dimensionality, and the number of actions and signals, are still required.

Another paper which considers communication is [16], regarding self referential games with codes of conduct. These codes are descriptions of how the players and opponents should play. An application to computer algorithms is provided. Two folk theorems are proved: one assuming common knowledge of the codes of conduct, and the other where only certain players observe certain codes. The latter is the main result of the paper and is motivated by the fact that, often, individuals have good knowledge of those closest to them but not the whole community. [16] obtain the sufficient condition that, with public communication, if every player is observed by another two opponents, then a folk theorem is yielded.

2.5 Finite Horizon Games

Another area of interest is the existence of folk theorem-type results for games of finite repetition. [13] explores the case of finitely-repeated games of complete information and the associated subgame perfect equilibria. They state that, despite the existence of games, when repeated finitely, “produce no non cooperative equilibrium outcomes” besides those of the single game, there may be subgame perfect equilibria of finite-repeated games when the corresponding single game has multiple equilibria. Indeed, using their “three phase punishment”, [13] prove that “any rational and feasible payoff vector can be obtained in the limit”, assuming the feasible payoff region has dimension equal to the number of players and each player has two Nash equilibrium payoff values. In a similar manner, [5] considers an alternative version of the Prisoner’s Dilemma in which an additional strategy is included. This gives a second pure-strategy equilibrium and a folk theorem result for a finitely repeated version of this game.

Although not strictly a finite game, [34] study a repeated game in which players may “strategically terminate” it. In particular, this involves the incorporation of a voting-step at the start of each repetition where a certain number of players decide whether to keep interacting. This is motivated by the increasing possibilities of ending business partnerships, due to more technology and knowledge. They prove a general folk theorem for any stage game (with the additional voting) which is satisfied “for all majority rules except the unanimous ending”. Indeed, for the unanimous ending rule, they show that the theorem may not hold but sufficient conditions are provided for when it is satisfied.

2.6 Stochastic and Sequential Games

Other game types to have associated folk theorem results include stochastic [27] and overlapping generation [15, 36].

[27] explains that, often, the standard assumption of “a completely unchanging environment”, within the theory of repeated games, is not reliable in applications. This reasoning is the motivation for studying the more generic stochastic games. These games may not have a pre-decided stage game, instead a ‘state variable’ is used to represent its environment which alters according to “initial conditions, player’s actions, and the transition law”. The paper discusses equilibrium payoffs in the case of very patient players without the need for the Markovian property. Specifically, perfect monitoring is assumed along with asymptotic state independence and either of payoff asymmetry or full dimensionality. [27] proves

two folk theorems: one with unobservable mixed strategies (in which case, full dimensionality is required) and also, similar to [1], one with the slightly weaker condition of payoff asymmetry (here, mixed strategies have to be observed).

Both [15, 36] provide insight into folk theorems associated with overlapping generation games. These are similar to the repeated normal form games except that the set of players are considered to be finite. That is, each player is involved in a certain number of stage games before they are replaced by another, identical player. [36] proves a variety of folk theorems both with and without discounting and / or observable mixed strategies. He shows that the full dimensionality assumption is not required in these games since players are assumed to not end simultaneously. On the other hand, [15], although does provide a folk theorem, his main result is an anti-folk theorem, see 2.7. He studies games with imperfect public monitoring and states that, in such overlapping generational games, cooperation becomes impossible in contrast to repeated games. [15] constructs a mixed strategy equilibrium folk theorem however he goes on to show that these strategies are unstable to perturbations, resulting in an anti-folk theorem.

A similar study by [4] looks at dynastic repeated games. These differ from the overlapping generation games in two aspects: perfect observation of the past is assumed, and the payoffs obtain no dynastic component. Under the assumptions of full dimensionality and the existence of a payoff vector which strictly Pareto dominates the stage game equilibrium, [4] proves a folk theorem for private communication games with greater than three players in sequential equilibria.

Considering now sequential games, in which players do not choose their action simultaneously, [91] introduces a concept of effective minimax values before proving a corresponding folk theorem. In his model, players pick actions in groups and thus the effective minimax value is defined to be the lowest equilibrium payoff a player will receive even if none of their opponents have equivalent utilities. [91] claims his folk theorem can be applied to other game models as it is a “uniform characterisation”.

2.7 Anti-Folk Theorems

A common theme in the proofs of folk theorems is the use of strategies which “identify and punish” deviators [59]. However, as soon as the game contains incomplete / imperfect information, deviators cannot necessarily be identified. This yields a much smaller equilibrium set and these results are termed “Anti-Folk Theorems”. This description of anti-folk theorems is adapted from [59], who

state the original term was given in Kaneko (1982) and Dubey & Kaneko (1984) [FIND PAPERS]. [59] prove an anti-folk theorem using the “long-run average criterion” in contrast to the discounted criterion.

Considering similar models to [15, 36], [95] obtain an anti-folk theorem for a limited observability overlapping generations model. They show that cooperation cannot be sustained when new players can only observe recent history. This is in contrast to the folk theorems obtained under the assumption of common knowledge of all past actions. Though their results are restrictive in certain cases, they justify the work by stating it is suitable for modelling “high turnover” rates.

Another game model where an anti-folk theorem has resulted is a repeated game with private monitoring [70]. In this paper, the following three assumptions are made: infinite and connected private monitoring (that is, infinitely many connected signals); finite past; and independent and identically distributed shocks affect the payoffs. Under these assumptions [70] shows the violation of the folk theorem. That is, the equilibria of the repeated game consist only of the equilibria of the single game.

2.8 Evolutionary Stability

Recently, there has been research into the results of the folk theorem with respect to the evolutionary game theoretic paradigm. Indeed, [53] state that the folk theorem is often used to characterise evolutionary stable strategies since exact solutions using theoretical results from evolutionary game theory are hard to obtain. However, they show the assumption that the folk theorem yields all Nash equilibria to be misleading. This is achieved by defining “type-k equilibria” which are a refinement of the Nash equilibria. [53] go on to prove that the set of type-k equilibria is contained within the set of repeated game Nash equilibria using “reactive strategies”.

In contrast to [93, 94, 28] who discuss folk theorems using belief-free strategies, [37] discusses the instability in an evolutionary sense. He shows that the belief-free equilibria are not robust to small perturbations in games with private monitoring and, in certain cases, this is extreme. Similar to [53], he states that Nash equilibria are used to predict evolutionary behaviour since they are thought of as stable. However, he goes on to show that only the choice of repeated stage game Nash equilibria satisfy evolutionary stability.

2.9 Recent Applications

In recent years, studies have been applying results of the folk theorem in various areas for creating algorithms, for example. This section briefly discusses a few of these.

[22] uses the folk theorem in the creation of a model which aims to suppress the effects of distributed denial of service attacks. They claim that all networks suffer from attacks to infrastructure and services. Thus, they use the programmability of software defined network environments to perform a game theoretic analysis. [22] create an algorithm for reward and punishment based on the Nash folk theorem. Similarly, [86] make use of the cooperative equilibrium solution from [31]s folk theorem to create an algorithm suggested to optimise a multi-period production planning based real-time scheduling method for a job shop. Moreover, [87] uses the result of the folk theorem in the IPD to study the cooperation rates of varying agent strategies in a multi-agent system.

Another application of the folk theorem is an algorithm used to obtain equilibria of a discounted repeated game [69]. [69] introduce a new algorithm, entitled “Communicate & Agree”, which can find equilibria in incomplete information, but perfect monitoring, games. Using the folk theorem in the algorithm enables the payoffs obtained to be potentially higher than those achieved by repeating the Nash equilibria of the single game. However, [69] go on to highlight that the algorithm is not guaranteed to always find equilibria. They say it is dependent on: the discount factor, sampling density and whether it is a zero-sum game or not. Finally, [40] discuss the potential of using game theoretic ideas in quantum optimal transport. In particular, they define the Quantum Prisoner’s Dilemma and explore the possibility of a quantum folk theorem in relation to the corresponding repeated game.

2.10 Conclusion

In this section, an overview into research regarding the folk theorem has been provided. The research history of the folk theorem spans from the 1970s until now, with many different models being considered. Examples include: games with complete information, games with imperfect private monitoring and finite-horizon games. However, there has also been studies into situations where the folk theorem does not hold, or the equilibrium strategies used in proving the theorems are unstable and / or not robust.

Chapter 3

Methodology and Experiment Setup

In this chapter the methods used to collect the data is provided along with justifications. The study required the execution of several IPD tournaments and thus appropriate software needed to be implemented and tested for accuracy.

3.1 Data Collection Algorithm

This section describes the overall algorithm used to collect the data and the attributes collected. Firstly, the aim of this exercise was to illustrate the Folk Theorem and analyse the p-values via a large empirical data collection. It was expected that plots similar to Figure 3.1 would appear. This clearly shows that there eventually does exist a probability of game ending where defection is not a rational decision.

Therefore, in order to observe whether any ‘environmental’ settings of the tournament does affect the p-threshold, a large amount of data is needed in order for any observations to be statistically significant. Figure 3.2 shows a pictorial representation of the collection method. Each step visible will be explained in detail throughout this chapter with references to the appropriate sections.

From Figure 3.2, it can be seen that the first step was to set up an empty database ready to input each tournament result into. The specific details of implementation into the algorithm is discussed in Section 3.2. However the choices made on the attributes to collect will be described here.

Table 3.1, shows each attribute chosen to be observed in this study. The attributes `experiment_number` and `tournament_player_set` were used to provide

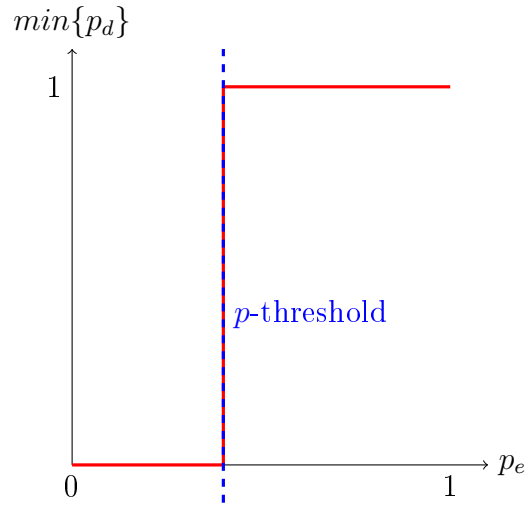


Figure 3.1: An example plot illustrating the p -threshold as indicated by the Folk Theorem

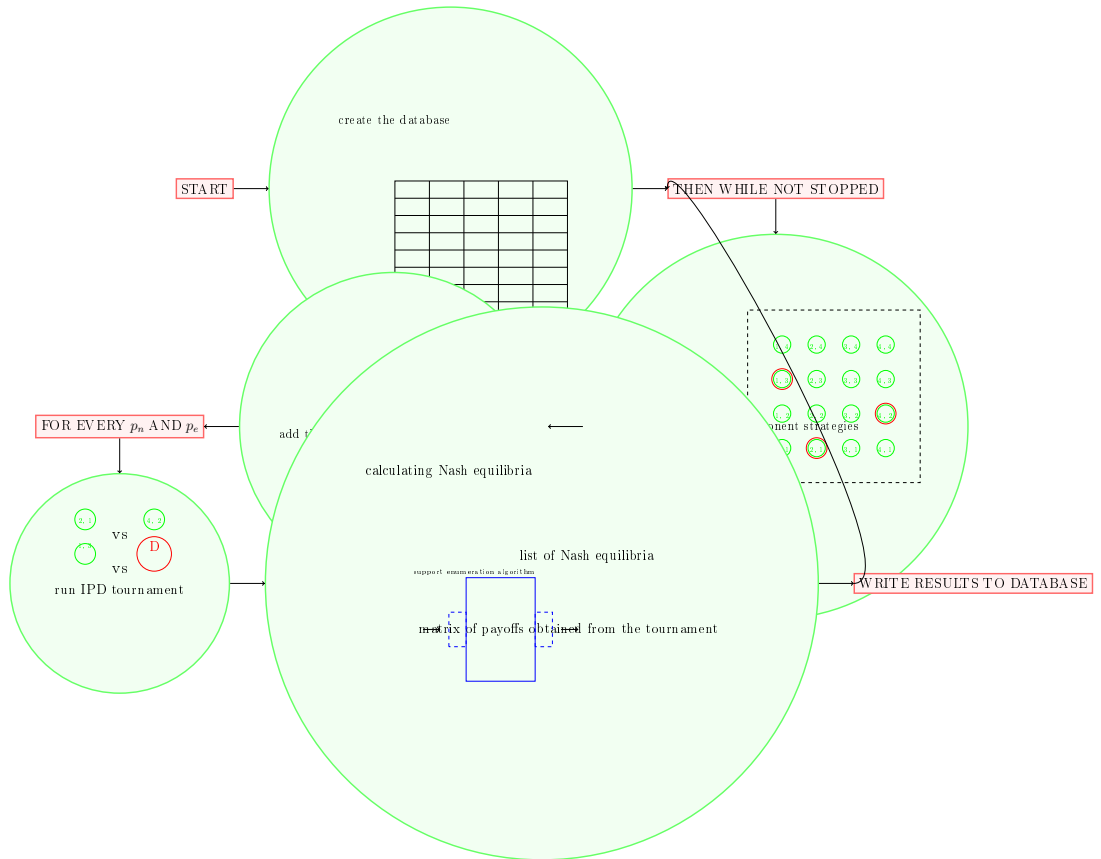


Figure 3.2: Representation of the algorithm used to collect the data.

Attribute	Description
experiment_number	A unique seed for each tournament run.
number_of_players	The number of strategies including the Defector.
tournament_player_set	A unique number for a particular set of strategies.
player_strategy_name	The strategy name as given in [24].
is_long_run_time	Characteristic from [24].
is_stochastic	Characteristic from [24].
memory_depth_of_strategy	Characteristic from [24].
prob_of_game_ending	The value of the game ending probability parameter implemented in the tournament.
payoff_matrix	A string of the matrix of mean payoff values from the tournament.
num_of_repetitions	A value indicating how many iterations of the tournament is required.
num_of_equilibria	The number of equilibria obtained as output from the algorithm chosen.
nash_equilibria	A string containing the list of yielded equilibria.
least_prob_of_defection	The lowest probability of playing the Defector obtained from the nash equilibria.
greatest_prob_of_defection	The highest probability of playing the Defector obtained from the nash equilibria.
noise	Level of additional noise added to the tournament.
warning_message	This column contained a string if the algorithm detected potential degeneracy.

Table 3.1: Table of attributes collected.

a unique identification for each tournament and set of strategies for ease of separation during analysis. The three characteristics of the strategies were collected with the aim of analysing the different ‘types’ of strategies’ effect on the threshold. The attributes `least_prob_of_defection`, `warning_message`, `number_of_players` and `noise` were the most important attributes to this study. These were the main effects with regards to analysis of the threshold and key descriptors of the game. The rest of the attributes were retained for self-evaluation of degeneracy and in order to replicate the tournaments for validity and further research.

Following this came the selection of opponents. The number of opponents selected ranged from one to eight and were randomly selected from the appropriate collection of strategies in Axelrod [24]. Out of the 235 strategies currently implemented in Axelrod, only 216 were valid for this experiment. Since this was considering the Folk Theorem for the IPD, research strategies which ‘cheated’ (that is, those which return `False` when entered into the `obey_axelrod()` function) were omitted. The Defector strategy was also removed as it would later be added to each set of players. Moreover, due to time constraints, the 18 strategies which are classified as having a long execution time were omitted.

The actual IPD tournament was run using the original Axelrod tournament setup [2] as implemented in the library Axelrod [24]. This is a round-robin type tournament where each strategy plays every other strategy once [7]. Each round robin was repeated 500 times (why?) to obtain ‘smoother’ estimates of the mean payoff values. Moreover, each strategy set was run in a tournament for 100 game ending probabilities within the range $[0.001, 0.999]$. Note, 0 was not included as this would imply the tournament would never end and 1 was omitted as the tournament would immediately end after the first turn. However, it is possible for the probabilities within the range $[0.999, 1]$ to be included in this range and this is recommended for further research. These tournaments were also repeated for additional levels of noise in the set, $\{0.1, 0.2, \dots, 1\}$. The main output of interest here is the payoff matrix, which is then inputted as the game matrix in `Nashpy`. According to [24], each entry $a_{i,j}$ gives the mean payoff of player i against player j . Consider the following example:

$$\begin{pmatrix} 2.990 & 2.996 & 0.487 \\ 2.996 & 3.000 & 0.989 \\ 3.053 & 1.042 & 1.000 \end{pmatrix} \quad (3.1)$$

The matrix in 3.1 is yielded from a three player tournament with a game ending probability of 0.001 and no additional level of noise. The strategies in this case

were Colbert; Tideman and Chieruzzi; and Defector. In this case, the entry $a_{1,2} = 2.996$ is interpreted as the mean payoff between Colbert; and Tideman and Chieruzzi.

For the calculation of the Nash equilibria, the *support enumeration algorithm* was used. This algorithm as well as the justification of its use is provided in Section 3.5.

Finally, the values obtained were written to the database file, one record is inserted for each strategy in order to retain all characteristics. Here, any entries that were not integers or floats had to be converted to strings in order to be stored. For example, the payoff matrix given in 3.1 is a list and hence could not be inserted into the database in its current format.

In summary, pseudo code for the overall algorithm is provided in Algorithm 1.

Algorithm 1: Folk Theorem Exploration

```

input : maximum number of opponents, number of strategy sets
         for each number of opponents, noise levels, game ending
         probabilities, number of repetitions of the tournament, the
         path to the database file, and whether or not support
         enumeration should be used to calculate the Nash
         equilibria.

output: a database containing the results as detailed above.

1 while True do
2   for each number of opponents do
3     for each repetition with the same number of strategies do
4       Randomly select a set of opponents and add in the
         Defector
5       for each noise level do
6         for each game ending probability do
7           Run the IPD tournament
8           Obtain the Nash equilibria and the corresponding
             probabilities of defection using the algorithm
             indicated
9           for each player in the current set do
10            Write the required information to a record in
              the database file.
11          end
12        end
13      end
14    end
15    Repeat
16  end
17 end

```

3.2 Databases

There are many different options for types of file available for storage of data. For example, csv, json, tex, txt, db etc. These are generally split into two types: plain text and binary files. In this section, the justification for using an SQLite database is provided along with how this was implemented. But firstly, the advantages and drawbacks of plain text and binary files are discussed.

Plain text or flat file is a format which stores data entries in a single table with column separated by delimiters such as commas or tabs ???. The contents are understandable by humans. Examples of these include csv and txt. The advantages of plain text formats include: a simple structure, less disk space used and portable ??. However, there are also drawbacks. Flat files are not scalable and are protected by less security [83]. Only one user can edit the file at any one time and when wanting to search through the file, it has to be fully loaded into the system [18]. Moreover, the columns must all contain the same data type [81].

On the other hand, binary file is a format in which sequences of 0s and 1s are unconstrained as compared to plain text files (where the binary codes have to represent character sets) [77]. Examples of these include databases, executables and media files [77]. The benefits of using binary file formats include: uses less storage space used, is less effort computationally and more secure (it is not understood by humans) [8] Though this this also a disadvantage as it makes a file harder to edit and understand [8].

3.2.1 Types of Database

Using the reasons provided above, it was decided that a binary file, in particular, a database file would be the most appropriate format to use. Primarily, this was due to the fact that databases are generally more robust and support out of memory operations. Indeed, a csv file could have been used however every entry would have had to be a string and if this contained commas, it would break the column structure. Research into the ideal file format for database collection resulted in the identification of two main types of databases: relational databases and noSQL (Not Only SQL) databases.

Relational databases are a file format which stores data according to the relational model as described in [23]. Examples of relational databases management systems include: SQLite, MySQL, PostgreSQL, Oracle and Db2. Briefly, this model involves the structuring of the data in a table, where each row is a record of an instance with a unique ID, or key, and each column is an attribute of the instances. This provided an ideal way to identify relationships between the varying records. This model was developed in the 1970s and was motivated by the reason that originally structures of databases varied with the application used. There are many advantages to a relational database format, including: data consistency - the data is immediately available across several instances of the database, no 'catch-up' time needed; commitment - strict rules regarding permanent changes

within the database; allows for stored procedures - blocks of code which can be repeatedly accessed; SQL (Structured Query Language) has been developed for ease of query performance using mathematics; and data locking / concurrency - allows for many users to query the database simultaneously without conflicts. A good paper on the model and benefits of relational databases is [66]. On the other hand, one major drawback to this format is its performance in handling extremely large data sets; which have become increasingly popular. Once the data goes beyond a certain size a relational database has to be distributed across many servers, also, this model does not support high scalability - that is, relational models are unable to support large volumes of workloads. Moreover, the strict structure required for this format means that if data cannot be easily transformed into this structure, the complexity of the model increases. The article [41], provides a more detailed description of these disadvantages.

Alternatively, noSQL databases were created with the motivation to be more efficient with large volumes of data. There are several different types of noSQL databases. Key-value store databases, such as RIAK, store the data as a simplistic key-value pair and are similar to hash tables. Column-oriented databases, for example Cassandra, are hybrid row-column databases and document-oriented databases, such as MongoDB, store 'records' in the form of documents with a unique key for representation. Finally, graph databases and object-oriented databases, such as Neo4j and Db4o, respectively, store the data as graphs (in the former case) and as objects, similar to those in Object Oriented Programming (in the latter case). Advantages of noSQL databases include: more flexibility with a wide range of models available; supports scalability; and are more efficient. However, these models are relatively new in comparison with relational models and there is no standard querying language. Also, some of these databases are not as effective as relational databases regarding consistency and commitment, and maintenance is challenging. For a more detailed approach to noSQL, see [63].

Using this information it was decided that a relational database would be the most appropriate. Although it is intended that a large amount of data will be collected, due to time limitations, it is thought that the database is unlikely to become too big for the system. Moreover, the structure and consistency of a relational model is ideal for comparison of the IPD experiments. The Database Management System decided upon was SQLite. This was due to the fact that there exist Python libraries, for example sqlite3 and sqlalchemy, for accessing the database and its contents. Also, it is portable, with the entire database stored in a single file meaning it could be transferred easily from the varying computers

```

database_management_sys = sa.create_engine(
    "sqlite:/// " + database_filepath + "main.db"
)
connect_dbms_to_db = database_management_sys.connect()

read_into_sql = """
    INSERT into folk_theorem_experiment
        (experiment_number, number_of_players, tournament_player_set,
         player_strategy_name, is_long_run_time, is_stochastic,
         memory_depth_of_strategy, prob_of_game_ending, payoff_matrix,
         num_of_repetitions, num_of_equilibria, nash_equilibria,
         least_prob_of_defection, greatest_prob_of_defection, noise,
         warning_message)
    VALUES
        (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)
"""

record = (
    experiment_number, number_of_players, tournament_player_set,
    str(player_strategy_name), is_long_run_time, is_stochastic,
    memory_depth_of_strategy, prob_of_game_ending, payoff_matrix_as_string,
    num_of_repetitions, num_of_equilibria, nash_equilibria_as_string,
    least_prob_of_defection, greatest_prob_of_defection, noise,
    warning_message,
)

connect_dbms_to_db.execute(read_into_sql, record)

```

Listing 1: Python code used to record the results from one strategy into the database.

being used. Other benefits include: its ease of use, with no configuration files and the fact it is self-contained [67].

3.2.2 Implementation of the Database

The ability to import results straight from Python was through the library sqlalchemy [11]. This allowed for the creation of the database through to accessing the results via Python functions and expressions. For example, consider the code in Listing 1 which was used to insert a record of results for one strategy into the database:

Also, to ensure records were being inserted into the database correctly, the graphical user interface, DB Browser [71] was utilised. It is implemented with a “familiar spreadsheet-like interface” for ease of use and is compatible with SQLite Databases making it ideal for this use [71]. See Figure 3.3, for a screenshot of the interface.

DB Browser for SQLite - C:\Users\sophi\Documents\final-project\src\database_code\data\se\main.db

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Pragma Execute SQL

Table: folk_theorem_experiment

Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	0	2	0	Inverse	0	1	Inf
2	0	2	0	Defector	0	0	0
3	1	2	0	Inverse	0	1	Inf
4	1	2	0	Defector	0	0	0
5	2	2	0	Inverse	0	1	Inf
6	2	2	0	Defector	0	0	0
7	3	2	0	Inverse	0	1	Inf
8	3	2	0	Defector	0	0	0
9	4	2	0	Inverse	0	1	Inf
10	4	2	0	Defector	0	0	0
11	5	2	0	Inverse	0	1	Inf
12	5	2	0	Defector	0	0	0
13	6	2	0	Inverse	0	1	Inf
14	6	2	0	Defector	0	0	0
15	7	2	0	Inverse	0	1	Inf
16	7	2	0	Defector	0	0	0
17	8	2	0	Inverse	0	1	Inf
18	8	2	0	Defector	0	0	0

1 - 19 of 825700

Go to: 1

Edit Database Cell

Mode: Text

Import Export Set as NULL

Type of data currently in cell: Text / Numeric

1 char(s)

Apply

UTF-8

08:46 03/03/2020

Figure 3.3: DB Browser, a database graphical user interface.

3.3 Software Implementation

There were many potential choices of language for the execution of this experiment however Python had the added advantage of pre-existing libraries, Axelrod [24] and Nashpy [82], which enabled running the IPD as well as calculation of the Nash equilibria. Thus, Python has been used for the collection and analysis of data.

Throughout the implementation of this experiment into Python, good software development principles have been followed [42, 75, 92]. Self-documenting code was ensured through the careful naming of variables as well as the use of docstrings to fully describe function parameters and usage. Python libraries Black [52] and Blackbook [49] assisted in improving the readability of the code, through formatting according to the guidelines of PEP-8 [73]. This gave consistency to all code files created during the study. Modularity through the creation of several smaller functions, focusing on one task, not only assisted with debugging but also enables future usability of the code in newer developments.

Testing is another key part of software development to ensure the durability of the code. Thus, unit tests have been implemented, using Pytest [51], to assist identification of bugs in the functions created for data collection. However, further work in this area is needed to provide a good coverage of the program. Indeed, from executing the Python library Coverage [10], a coverage of 60% was identified, yielding the report shown in Figure 3.4. This could be improved through the creation of integration tests between the database and the experiment results

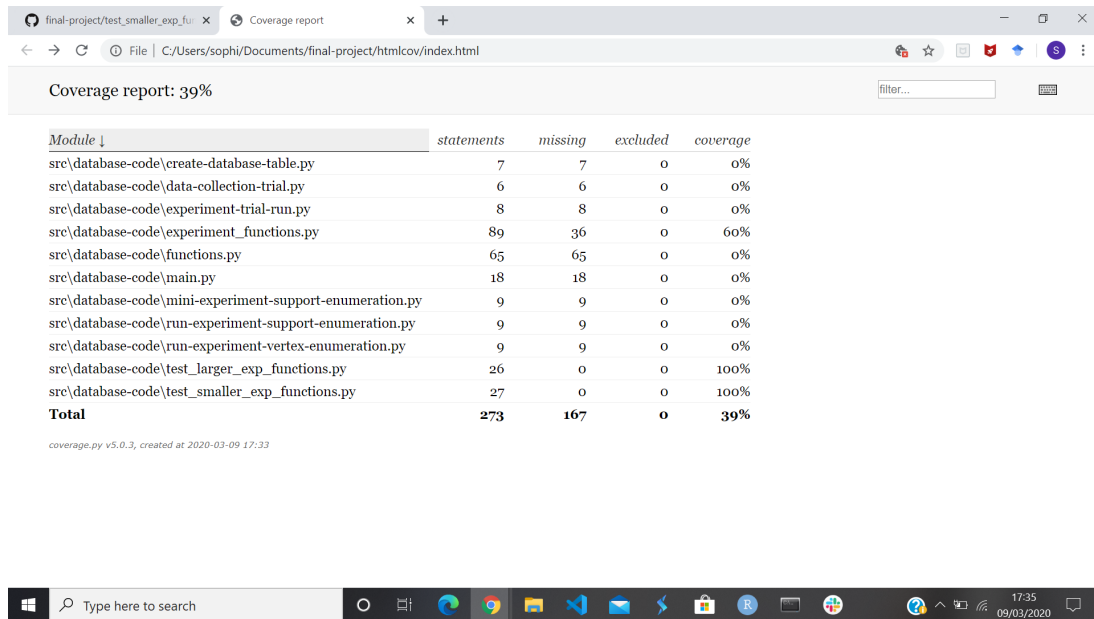


Figure 3.4: A screenshot of the html report produced when utilising Coverage.

or the implementation of functional tests to confirm the end result of the full algorithm.

The use of version control is key for keeping track of past and present changes to the system. In this study the software Git (<https://git-scm.com/>) was in use. This allowed for the adaption of code from several different function attempts and, through GitHub (<https://github.com/>), enabled collaboration between the author and supervisors as seen in Figure 3.5. Moreover, the use of GitHub ensured that a back-up copy of all project files were available, should the current system being used happen to fail and acted as an intermediate step between the author’s laptop and the remote server, used for running the experiment (see Section 3.4).

3.4 Remote Computing

This section describes the execution of the experiments via a remote computer and the reasons for doing so.

Firstly, due to the volume of data that was planned to be collected, it was decided that a running the data remotely would be ideal, in order to allow the code to run uninterrupted for several weeks. Thus, Cardiff University School of Mathematics’ computer Siren, a headless server with a large storage, was used. However, when a trial was executed, it was decided that the run time was ‘quick enough’ and hence parallel processing would not be necessary. Yet, for further runs of the

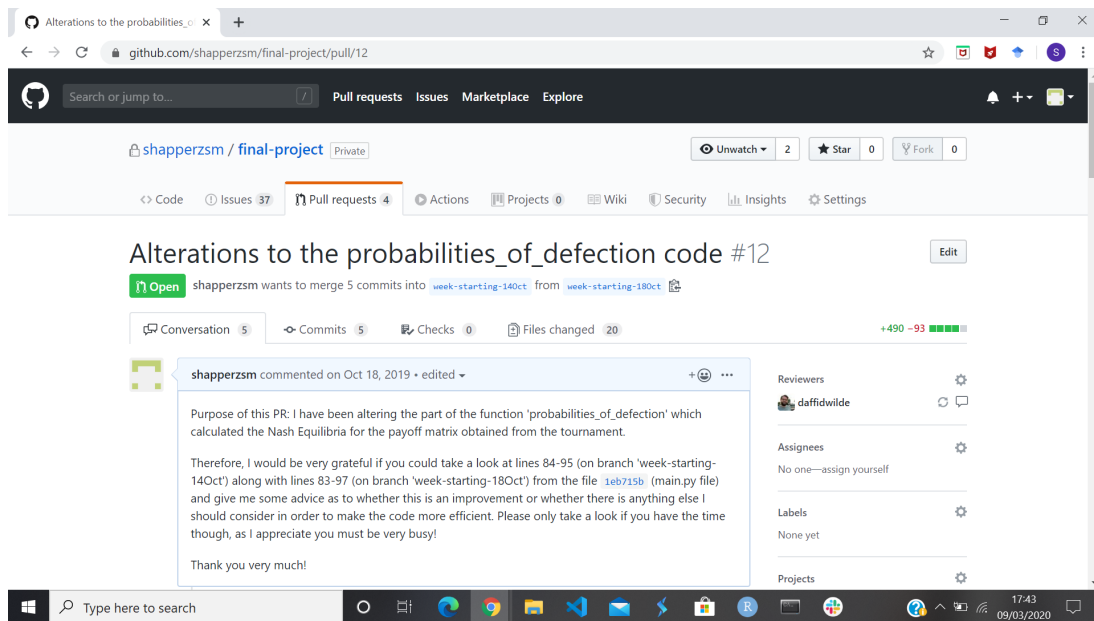


Figure 3.5: A screenshot of a GitHub pull request which allowed for collaboration between supervisors and author.

code, parallel processing is recommended, especially if the player set sizes being trialled are ‘large’. See Section 3.5.3 for an explanation.

To connect to Siren, an SSH tunnel was used. An SSH (or Secure Shell) tunnel is used for sending / receiving network data over an encrypted connection. It adds a layer of network security to those applications that do not natively support encryption and lowers the risk of interception. For a more detailed description of SSH, see [79]. Figure 3.6 shows the author’s connection to Siren via SSH.

In order for the experiment to keep running whilst disconnected from Siren, a terminal emulator was required, as Siren does not have a job scheduler. Specifically, the terminal multiplexer, TMUX [57] was used. This allowed for the creation and execution of several terminals from one screen. Moreover, a user could detach from a terminal and reattach later without execution being halted.

Figure 3.7, shows a diagram of how the remote server, ssh tunnel, and users laptop, were all in connection.

Once the experiment was running, the database had to be copied from Siren in order for the analysis to begin. This was also achieved via SSH. Note that, the code was written in a way which enabled the results to be written to the database concurrently, after every tournament. This was done to ensure that if the system were to break, data would still have been available and retained within the database. This meant the database file was almost always ‘open’ resulting

```
sophie@siren: ~/final-project/src

sophi@LAPTOP-31G4FMLS MINGW64 ~
$ ssh siren
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.4.0-173-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

 * Latest Kubernetes 1.18 beta is now available for your laptop, NUC, cloud
   instance or Raspberry Pi, with automatic updates to the final GA release.

   sudo snap install microk8s --channel=1.18/beta --classic

 * Multipass 1.1 adds proxy support for developers behind enterprise
   firewalls. Rapid prototyping for cloud operations just got easier.

   https://multipass.run/

30 packages can be updated.
0 updates are security updates.

New release '18.04.4 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

*** System restart required ***
Last login: Mon Feb 10 18:00:19 2020 from 10.163.78.29
sophie@siren:~$ htop -user sophie
Error: invalid user "ser".
sophie@siren:~$ tmux ls
0: 1 windows (created Fri Feb 7 14:12:20 2020) [172x48]
1: 1 windows (created Fri Feb 7 14:13:01 2020) [172x48]
2: 1 windows (created Mon Feb 10 18:07:45 2020) [172x48]
sophie@siren:~$ tmux attach -t 2
[detached (from session 2)]
sophie@siren:~$ tmux attach -t 0
[detached (from session 0)]
sophie@siren:~$ ls
examples.desktop  final-project
sophie@siren:~$ cd final-project/src/
sophie@siren:~/final-project/src$ ls
analysing_the_data  analysing-the-data  database_code  database-code  htmlcov  __init__.py
sophie@siren:~/final-project/src$
```

Figure 3.6: A screenshot of the authors connection to Siren via an SSH tunnel.

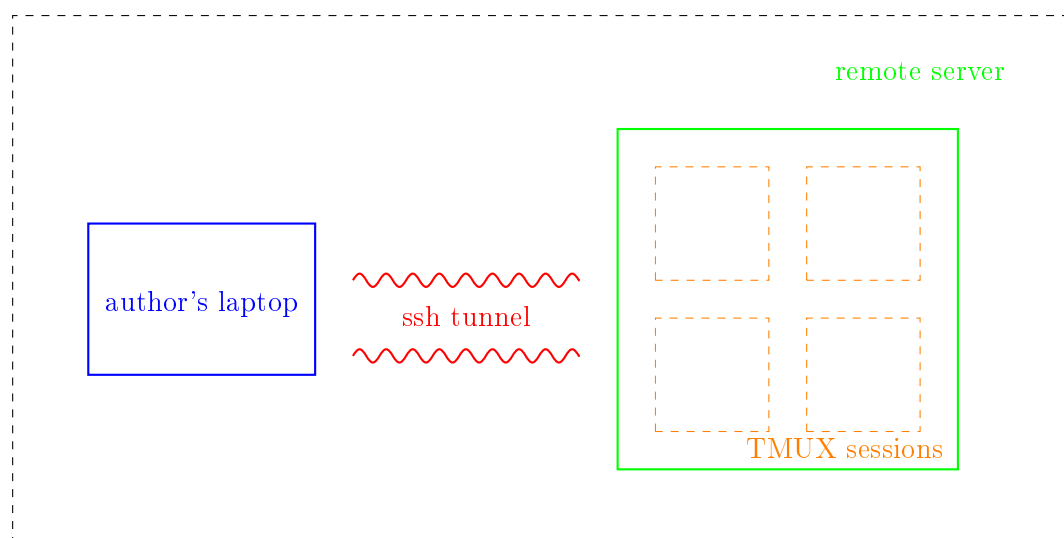
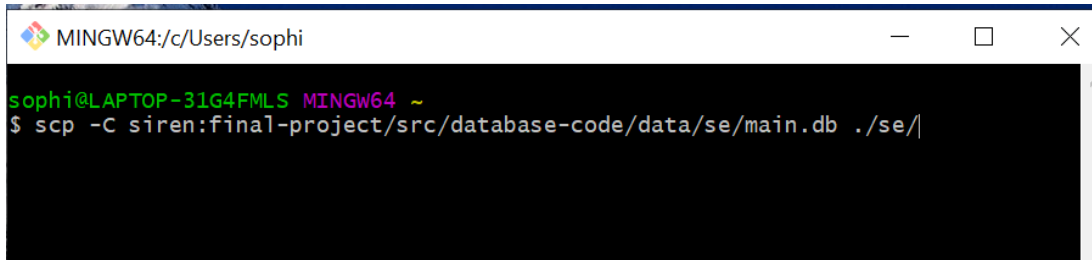


Figure 3.7: Representation of how the experiments were run remotely. Note, ‘tmux sessions’ correspond to emulators of terminals.



```

MINGW64:/c/Users/sophi
sophi@LAPTOP-31G4FMLS MINGW64 ~
$ scp -C siren:final-project/src/database-code/data/se/main.db ./se/

```

Figure 3.8: Command line code used to retrieve the database from the remote server.

in the integrity check failing once transferred over SSH and an `OperationalError` in `sqlalchemy`. Thus, in order to load the database with no failures into a Jupyter Notebook, it needed to be compressed before transferring. This was achieved using the command line code as seen in Figure 3.8.

3.5 Calculating Nash Equilibria

Recall, Nash’s Theorem, Theorem ?? explains that there exists at least one equilibrium in every finite game. However, it does not indicate how to obtain them. The proof of the theorem relies on finding the fixed point of the defined mapping however, the proof of Brouwer’s Fixed Point Theorem is an existence, and not a constructive, proof. That is, it does not give a method for obtaining the fixed point. Indeed, although not NP-complete ¹, finding Brouwer fixed points has been shown to be a hard problem [68, 38]. Thus, defining algorithms which obtain Nash equilibria to some degree of efficiency has been a large research topic for many years (include some evidence, i.e. paper references here?). Within the Python Library `Nashpy` [24], three such algorithms have been implemented: Support Enumeration, Vertex Enumeration and Lemke-Howson. However, the Lemke-Howson Algorithm will only find *one* equilibria and hence is not suitable for this study. Therefore, the definitions of the first two algorithms, in the case of a two player game, are provided. Unless specified otherwise, this section (Section 3.5) is adapted from [64].

¹Both finding Brouwer fixed points and Nash equilibria cannot be NP-complete since existence of a solution is guaranteed [64]. Most problems in the set NP-complete are situations in which a solution might not exist [64]. However, [68] has shown that these two problems belong to a alternative complexity class, PPAD or Polynomial Parity Argument (Directed case). For a discussion into this, readers are referred to [68].

3.5.1 Support Enumeration

Before stating the method for the support enumeration algorithm, a few extra theoretical ideas are needed.

Recall, a *mixed strategy*, σ , is a probability distribution over the pure strategies. Then:

Definition 3.5.1. The *support* of σ is the set of all pure strategies, $s_i \in \sigma$, such that $s_i > 0$. That is, all pure strategies which have a positive probability within the mixed strategy.

Definition 3.5.2. A game $G = (N, (S_i)_{i=1,\dots,N}, (u_i)_{i=1,\dots,N})$ is called *non-degenerate* if no mixed strategy of support size $1 \leq k \leq |S_i|$ has more than k pure best responses.

For example, consider the following 2-player normal form game:

$$A = \begin{pmatrix} 2 & 1 & 0 \\ 2 & 0 & 3 \end{pmatrix} \tag{3.2}$$

Here, if the column player was playing the strategy $\sigma_2 = (0.2, 0.8, 0)$ then the support is {strategy1, strategy2}. Also, observe that, if the column player picked strategy1 then the row player could choose either their first or second strategy and hence this game is *degenerate*.

The Algorithm

Algorithm 2: Support Enumeration	
	input : A <i>nondegenerate</i> two-player normal form game, where A, B are the row and column player's payoff matrices and σ_1, σ_2 are their strategy vectors, respectively. output : Every Nash equilibrium of the input game.
1	for all $k = 1, \dots, \min m, n$ do
2	for all support pairs I, J , with $I \in M, J \in N$ and $ I = J = k$ do
3	Solve
	$\sum_{i \in I} \sigma_{1i} B_{i,j} = v, \text{ such that } \sum_{i \in I} \sigma_{1i} = 1, \sigma_{1i} \geq \mathbf{0} \text{ for all } j \in J$ <div style="text-align: right;">(3.3)</div>
	and
	$\sum_{j \in J} A_{i,j} \sigma_{2j} = u, \text{ such that } \sum_{j \in J} \sigma_{2j} = 1, \sigma_{2j} \geq \mathbf{0} \text{ for all } i \in I$ <div style="text-align: right;">(3.4)</div>
4	Check the best response conditions
	$\sigma_{1i} > 0 \implies (A\sigma_2)_i = \max\{(A\sigma_2)_k k \in M\}$ <div style="text-align: right;">(3.5)</div>
	and
	$\sigma_{2j} > 0 \implies (\sigma_1 B)_j = \max\{(\sigma_1 B)_l l \in N\}$ <div style="text-align: right;">(3.6)</div>
5	end
6	end

Note, solutions are not guaranteed to exist for Equations 1.3 and 1.4. In this case, the support does not yield a Nash Equilibrium.

A example

Here we consider the computation of Nash equilibria via support enumeration for a payoff matrix obtained from a two player IPD tournament executed during this experiment ².

²Note, the Defector's opponent in this tournament is stochastic player, Inverse. Here, $p_e = 0.0514$, $p_n = 0$. This game was not identified as degenerate.

Consider the following matrix:

$$A = \begin{pmatrix} 3.000 & 0.829 \\ 1.686 & 1.000 \end{pmatrix} \quad B = A^T \quad (3.7)$$

Firstly, take $k = 1$, that is, looking for any pure best responses:

$$A = \begin{pmatrix} \underline{3.000} & 0.829 \\ 1.686 & \underline{1.000} \end{pmatrix} \quad B = \begin{pmatrix} \underline{3.000} & 1.686 \\ 0.829 & \underline{1.000} \end{pmatrix}$$

Thus, two pairs of pure best responses are visible, giving the following two Nash equilibria:

$$\sigma = \{(1, 0), (1, 0)\} \text{ and } \sigma = \{(0, 1), (0, 1)\}.$$

Since this is a two-player game, the only other support that needs checking is $I = J = \{1, 2\}$.

Here, Equations 1.3 and 1.4 become,

$$3\sigma_{r1} + 0.829\sigma_{r2} = 1.686\sigma_{c1} + \sigma_{c2} \text{ and } 3\sigma_{c1} + 0.829\sigma_{c2} = 1.686\sigma_{r1} + \sigma_{r2}.$$

Then, rearranging and checking the constraint $\sum \sigma_r = 1$, $\sum \sigma_c = 1$, yields

$$\sigma_{r1} = \sigma_{c1} = \frac{19}{165} \text{ and } \sigma_{r2} = \sigma_{c2} = \frac{146}{165}$$

Note, checking the best response condition for two players with the same number of pure strategies is trivial [48]. However, it is included here for completeness.

$$A\sigma_c^T = \begin{pmatrix} 3 & 0.829 \\ 1.686 & 1 \end{pmatrix} \begin{pmatrix} \frac{19}{165} \\ \frac{146}{165} \end{pmatrix} = \begin{pmatrix} 1.079 \\ 1.079 \end{pmatrix} \text{ and } \sigma_r B = \begin{pmatrix} \frac{19}{165} & \frac{146}{165} \end{pmatrix} \begin{pmatrix} 3 & 1.686 \\ 0.829 & 1 \end{pmatrix} = \begin{pmatrix} 1.079 \\ 1.079 \end{pmatrix}$$

Therefore, the best response condition holds for both players. Hence, the third and final Nash equilibrium is given by:

$$\sigma = \left\{ \left(\frac{19}{165}, \frac{146}{165} \right), \left(\frac{19}{165}, \frac{146}{165} \right) \right\}$$

Advantages and Drawbacks

Support enumeration is known to be a robust method for obtaining Nash. That is, given a non-degenerate game, it is guaranteed to return all equilibria and, even in the case of degeneracy, it will find some equilibria (see Section 3.6). However, this method essentially compares all pairs of supports and thus has an exponential complexity [72]. This implies that support enumeration is computationally expensive and the larger the game becomes the slower it will become.

3.5.2 Vertex Enumeration

Vertex enumeration is based on a geometric representation of games and hence, here, a brief introduction to this is provided. The reader is referred to [64] for a more detailed approach to this topic.

Definition 3.5.3. Let A, B be *positive* payoff matrices for the row and column player; that is, each element $a_{ij}, b_{ij} > 0$, for all $i = 1, \dots, M, j = 1, \dots, N$. Then the row, column *best response polytopes*³, denoted P, Q are given respectively by

$$P = \{x \in \mathbb{R}^M | x \geq \mathbf{0}, xB \leq \mathbf{1}\} Q = \{y \in \mathbb{R}^N | Ay^T \leq \mathbf{1}, y \geq \mathbf{0}\}. \quad (3.8)$$

Note, here it is assumed that the utility values which appear in 3 and 3 have been normalised to 1. This means that the vertices are no longer probabilities and hence scaling will be required to find the Nash equilibria. Also, the strictly positive payoffs is not a constraint since a constant can be added to each with no effect.

For example, consider the payoff matrix

$$A = \begin{pmatrix} 1 & 5 \\ 4 & 1 \end{pmatrix} \quad (3.9)$$

Then the best response polytope, P here is given by the inequalities: $x_1 \geq 0$,

$$x_2 \geq 0,$$

$$x_1 + 5x_2 \leq 1,$$

$$4x_1 + x_2 \leq 1$$

which yields the following as given in Figure 3.9.

³Note, in general, a *polytope* is defined as a bounded set $\{z \in \mathbb{R}^d | Cz^T \leq q\}$ where C is a $k \times d$ matrix and z is a $1 \times d$ vector [64].

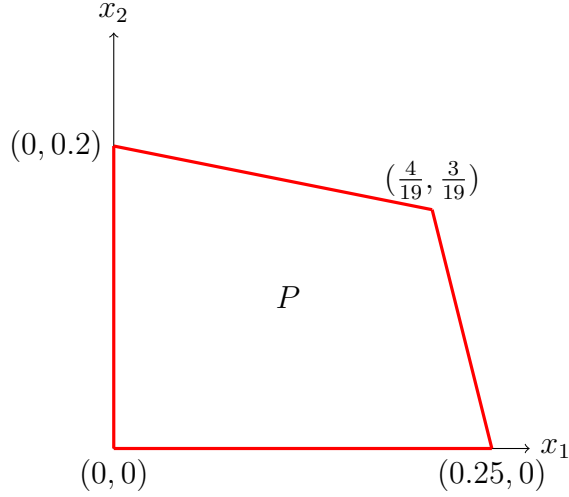


Figure 3.9: Best response polytope, P , obtained from the payoff matrix given in Equation 3.5.2.

The Algorithm

Two algorithms are stated here, the first ‘prepares’ the polytope for the second one, which returns the Nash equilibria.

Algorithm 3: Vertex Labelling

input : A polytope, $P \in \mathbb{R}^n$.
output: A labelled-vertex polytope.

- 1 enumerate each of the defining inequalities of P , c_1, \dots, c_k
- 2 **for** each vertex $v_i \in P$ **do**
- 3 find the inequalities of P which are *binding* at v_i , that is the defining equations are equalities.
- 4 the label of v_i is given by $\{c_{i1}, \dots, c_{il}\}$, where c_{ij} is in the label if and only if equation c_j is binding for v_i
- 5 **end**

A pair of labels $v_i \in P$, $u_j \in Q$ are called *fully labelled* if every inequality ‘number’ of $P \cup Q$ appears in either the label of v_i or the label of u_j . Then there is a notion which states that each fully labelled pair, when normalised, corresponds to a Nash equilibrium. Note, this does not include the vertex pair $(\mathbf{0}, \mathbf{0})$ since, although this is a fully labelled pair, it corresponds to neither opponent playing

any strategies.

Algorithm 4: Vertex Enumeration

```

input : Best response polytopes,  $P, Q$ , as defined in 3.5.2, for a
         nondegenerate game.
output: All Nash equilibria of the corresponding game.
1 for each polytope,  $P, Q$  do
2   | Execute Algorithm 3
3   | for each pair of vertices  $\{u_i, v_j\}$  in  $P, Q$  respectively, except
4   |   |  $(0, 0)$  do
5   |     | check if they are fully labelled. if they are fully labelled then
6   |       | | add to the list of equilibria
7   |       | end
8   |       | else
9   |       | | continue
10  |       | end
11 end

```

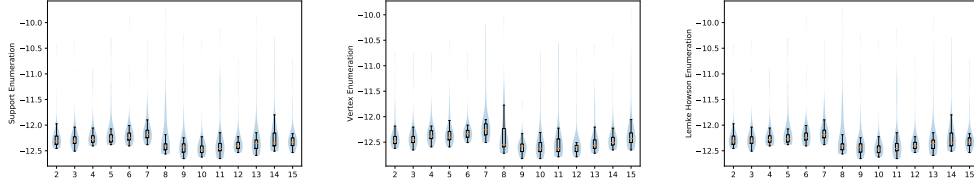
Advantages and Drawbacks

Vertex enumeration is more efficient than support enumeration, according to [64], since there are more supports in a game than there are vertices. Indeed, consider the example of $M = N$ where M, N are the number of binding inequalities in the best response polytopes, P, Q , respectively. Using the support enumeration algorithm, approximately 4^n support pairs will need to be looked at but, according to the ‘Upper Bound Theorem’ for polytopes [76, 3, 17], P and Q have less than 2.6^n vertices. Thus, given there exists an efficient method for enumerating vertices, then this implies less further computational complexity. However, if the game is degenerate, this algorithm will not return any Nash equilibria.

3.5.3 Algorithm Execution Timings

Due to the robustness of the support enumeration, it was decided that this would be the main method of calculating Nash equilibria. On the other hand, vertex enumeration was also included in a different run of the experiment, in order to compare the accuracy of the methods - though this algorithm does break if the algorithm is degenerate.

Having said this, timings of each of the three algorithms were obtained, to see if the computational complexity is significantly different for any of them. For the



(a) Log timings of the calculation of Nash equilibria using the support enumeration algorithm. (b) Log timings of the calculation of Nash equilibria using the vertex enumeration algorithm. (c) Log timings of the calculation of Nash equilibria using the Lemke-Howson enumeration algorithm.

Figure 3.10: Violinplots of the log timings obtained for the experiment and calculation of Nash equilibria using the three algorithms as implemented in Nashpy.

purposes of this trial run, the following parameters were used: 100 tournament repetitions, $p_e = 0.2$ and $p_n = 0$. The results can be seen in Figure 3.10.

From Figure 3.10, it can be seen that there is not a significant difference in the execution times of the algorithms. Hence, although support enumeration has a greater computational complexity than vertex enumeration - it is not going to have any recognisable impact on the number of experiments executed within the time frame.

3.6 Degeneracy

Recall, the definition of nondegeneracy, given in Definition 3.5.1. This implies a *degenerate* game is one in which there exists a support of size k where the number of pure best responses is greater than k . For example, consider the following payoff matrix:

$$A = \begin{pmatrix} 1 & 4 & 3 \\ 0 & 4 & 2 \end{pmatrix} \quad (3.10)$$

Then, if the column player picks their second strategy (a support of size 1), the row player can pick either of their strategies (two pure best responses). Hence this game is degenerate.

Let $G = (N, S_{ii \in N}, u_{ii \in N})$ be a degenerate game. Then, recall that support enumeration may not return all Nash equilibria. This is due to the fact that if solutions to the equations given in 1.3 and 1.4 of Algorithm 2 exist, they may not be unique. Indeed, the number of Nash equilibria in a degenerate game may be infinite [64]. Considering degeneracy in terms of vertex enumeration, means

```

with warnings.catch_warnings(record=True) as w:
    warnings.simplefilter("always")

    if support_enumeration is True:
        nash_equilibria = list(game.support_enumeration())

    else:
        highlight_numpy_warning = np.seterr(all="warn")
        nash_equilibria = list(game.vertex_enumeration())

if len(w) == 0:
    warning_message = None

else:
    warning_message = str([w[i].message for i in range(len(w))])

```

Listing 2: Python code used to ‘catch’ potential degeneracy.

that a vertex of the best response polytope $P = \{x \in \mathbb{R}^M | x \geq \mathbf{0}, xB \leq \mathbf{1}\}$ may have more than M labels leading to a ‘badly defined’ polytope. This implies the algorithm will break (need to double check).

Within the library Nashpy, the algorithms have been implemented such that if potential degeneracy is identified, for example by the reasons given above, then a warning is issued. Thus, in order to retain whether a game is possibly degenerate, the algorithm was required to ‘catch’ the warnings when given. This was achieved using the code seen in Listing 2, with the Python warnings module:

Note, warnings for potential degeneracy are only highlighted if the Nash equilibria obtained do not make sense, that is, are not a probability distribution. However, it is possible for the algorithm to obtain some correct Nash equilibria, but the game could still be degenerate. This means that, when stating statistics regarding the degeneracy of the games obtained during the experiment (see Chapter ??), these are only the ones caught by the algorithms in Axelrod and are not necessarily all of them.

In this chapter, the creation and execution of the empirical experiment was detailed in its entirety. Justifications were given as to why certain methods and software were used and / or not used. Also, how the Nash equilibria, explored in the next chapter, was explained with an example given. (Bullet point list of main conclusions?) In Chapter ??, an introductory analysis of the data obtained is given.

Chapter 4

Analyses

In this chapter, an analysis of the data collected via the methods described in the previous chapter (Chapter ??) is given. Firstly, a brief initial overview is provided, where descriptive statistics of the equilibria obtained and the overall characteristics of the strategies used is discussed. Following this, a critical analysis of the p -thresholds obtained is carried out. Here, the environmental effects, on the outcome of the game, discussed include: number of opponents and level of added noise. Note, as of writing, the database currently has 825700 entries (rows) and a total number of 159 tournament sets.

4.1 Initial Analysis

In this section, all the data (including those games which could be degenerate) are considered. Taking a brief look at the graphs produced for each set of tournaments, it can be seen that the main ‘shapes’ obtained are as seen in Figure 4.1.

In Figure 4.1a, a clear p -threshold of approximately 0.28 is apparent, clearly visualising the Folk Theorem. In this tournament set the opponent was *Inverse*, a stochastic player, indicating that perhaps the stochasticity of a player does not affect the accuracy of the threshold as first thought. In Figure 4.1b, the precise value of the p -threshold is less clear, lying approximately in the range $[0.1, 0.3]$. This could be due to the potential degeneracy identified or just the element of randomness that appears within each tournament, possibly magnified by the number of stochastic players. The opponents within this set were: *Feld: 1.0, 0.5, 200*; *Cooperator*; *EvolvedLookerUp2_2_2*; *Tullock: 11*; and *ZD-GEN-2: 0.125, 0.5, 3*. Figure 4.1c shows a potential problem with the visualisation of the Folk Theorem when degenerate games are involved. It becomes unclear

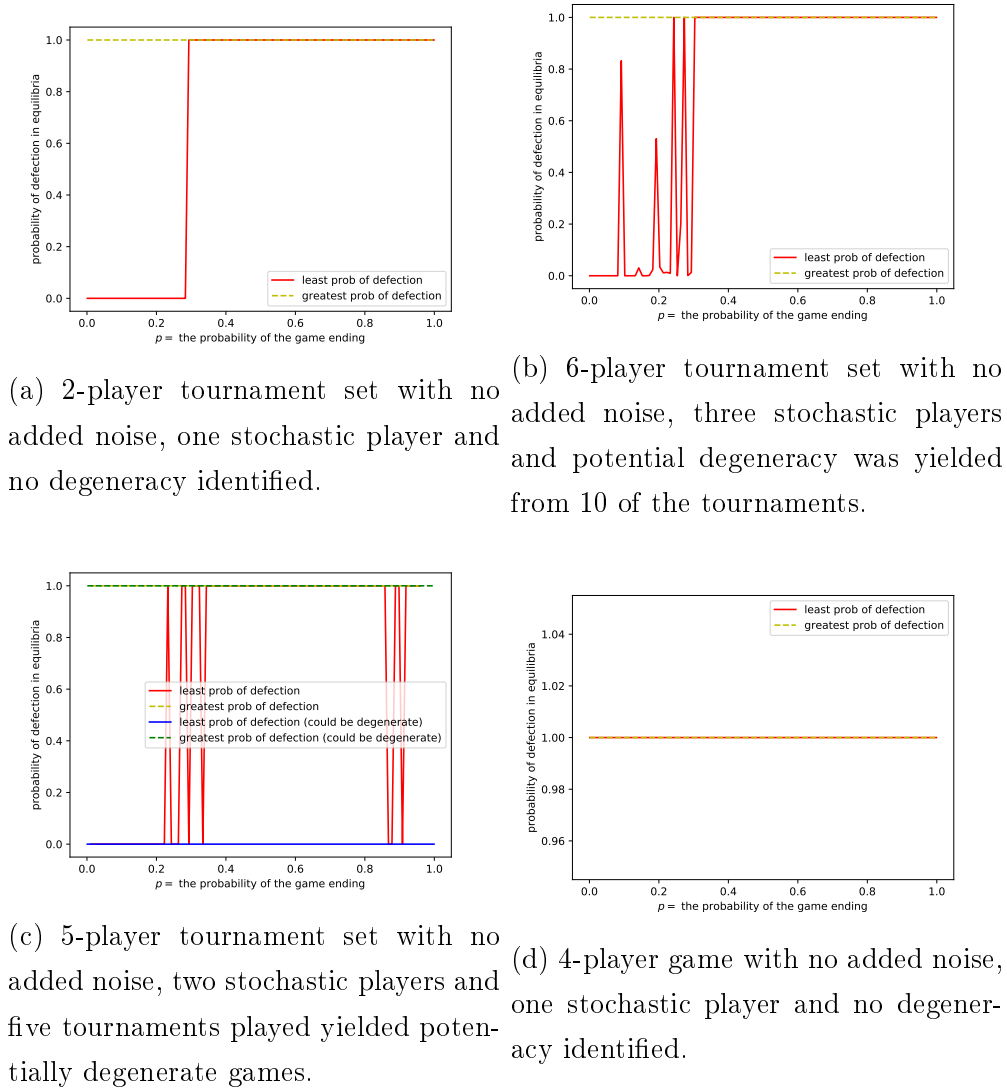


Figure 4.1: Example graphs obtained from the experiment.

as to what is happening in the graph, especially where the p -threshold lies. The opponents, in this case, were: *Random*: 0.5; *Grumpy*: Nice, 10, -10; *Fortress3*; and *Negation*. Finally, Figure 4.1d gives an example of a tournament set for which there was always a non-zero probability of defection, regardless of the game ending probability. In this case, the precision of game-ending probabilities chosen was not accurate enough to identify the p -threshold. It is implied that the tournament has to have an ending probability of almost 0 (within the interval $(0, 0.001)$) in order for a zero probability of defection to be rational. Here, the opponents of the *Defector* were: *AntiCycler*; *\$e\$*; and *Stalker*: (D ,). Similarly, it is observed that some graphs obtained were constant at 0, again indicating that the precision of game-ending probabilities were not fine enough to highlight the p -threshold. This indicates that the ending probability of the tournament has to lie within the interval $(0.999, 1)$. That is, almost immediately, the decision to defect is no longer rational.

The summary statistics gained from running the *describe* method of a pandas database is given in Table ?? . From this, it can be seen that the number of opponents the *Defector* played against ranged from one to seven, with an average of four opponents. Also, as expected, the mean probability of the game ending encountered was 0.5. Observe that, overall, there were 175,399 distinct tournaments played with a total of 159 distinct sets of strategies. Looking now at the statistics for Nash equilibria, it can be seen that a total of 823,823 equilibrium points were calculated in this experiment, with an average of $1.914 \approx 2$ equilibria per game. However, observe, at least one game obtained 39 equilibria which will be explored into later on in this section. Considering the probabilities of defection within these equilibria, notice that both the greatest and the least probabilities of defection ranged from zero to one inclusive with a 50th percentile of zero. But, looking at the average values, the least probability has a mean of 0.342 and only just above this, the greatest probability has a mean of 0.460.

Next, further descriptive statistics are calculated for the strategies. This is to obtain a more in-depth view on the types of strategies randomly chosen to play and their characteristics. Executing *value_counts* method on the column of strategy names, it is observed that the player which appeared the most times (9 times) is *ZD-GEN-2*: 0.125, 0.5, 3; followed closely by *Tideman* and *Chieruzzi* with 7 sets of tournaments. On the other hand 38 out of the 200 strategies playing in this experiment appeared only once. Running the *value_counts* method again, but this time on the memory depths of the strategies found the majority of strategies to have an infinite memory depth. On the other hand, strategies having no memory or a depth equal to one were also significant. Considering the stochas-

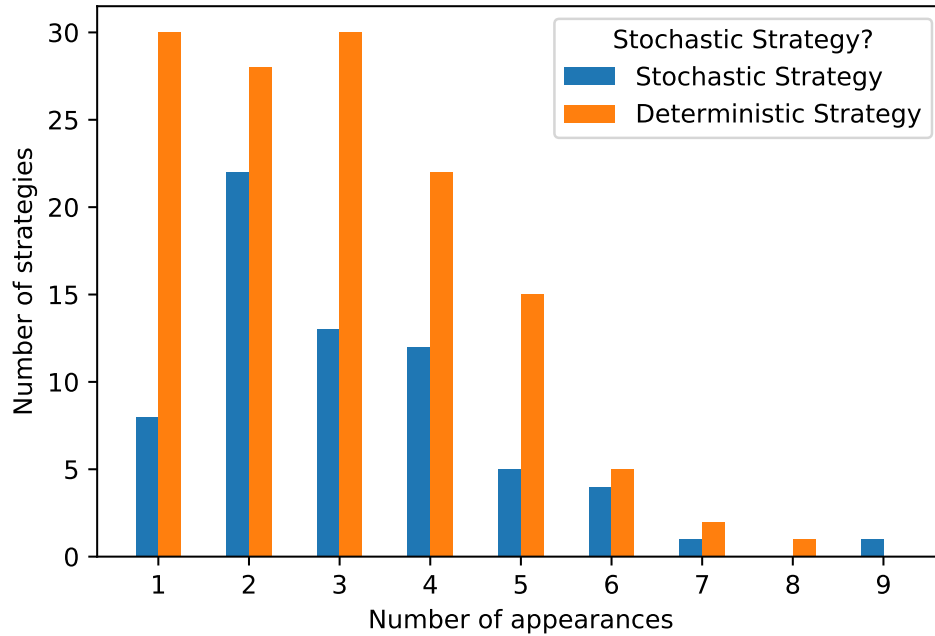


Figure 4.2: A plot to show the ratio of stochastic to deterministic strategies randomly chosen throughout the experiment.

ticity of players alongside how many appearances each strategy made yielded the following chart in Figure 4.2.

It is clear that there is a clear bias towards deterministic strategies in this experiment. However, this is to be expected as running the following code:

```
len(axl.filtered_strategies(filterset={"stochastic": True})),
len(axl.filtered_strategies(filterset={"stochastic": False}))

(86, 156)
```

it can be seen that over half the strategies coded into the Axelrod library are classed as deterministic. Looking at Figure 4.2 again, observe, the majority of deterministic strategies were executed either once or three times. On the other hand, a large proportion of the stochastic strategies were played twice.

Further, the number of Nash equilibria obtained for each game was analysed and their distributions with respect to the number of opponents of the *Defector* plotted. Executing the *value_counts* method on the 'num_of_equilibria' column gave the conclusion that the majority of games (131773) yielded one Nash equilibria with 28793 games obtaining 3 equilibria. Also, the maximum number of equilibria yielded, 39, was for one game, which, doing a search on the

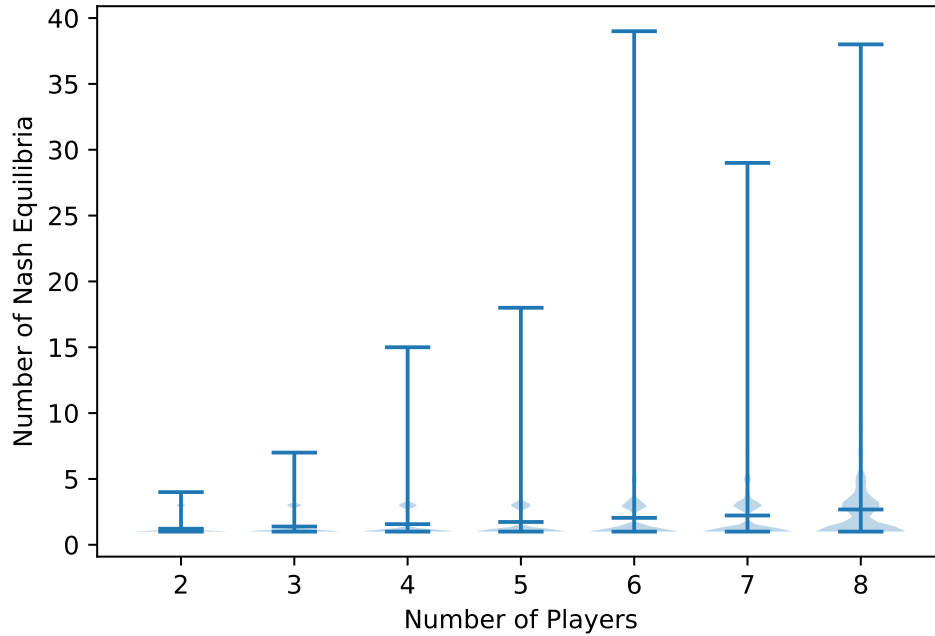
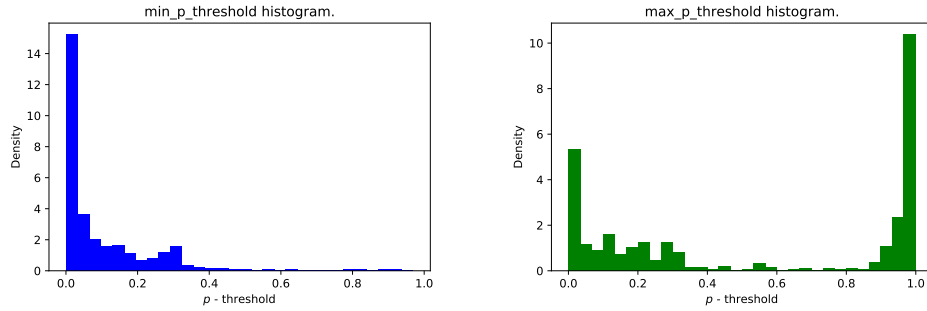


Figure 4.3: A violinplot showing the distribution of the number of equilibria obtained for varying number of players.

database, was found to be a six player game with noise=0.1. The opponents of the *Defector* were: *Inverse Punisher*; *Prober*; *PSO Gambler 2_2_2 Noise 05*; *Handshake*; and *More Tideman and Chieruzzi*. This game was expected to be degenerate however, when taking a closer look at this entry in the database, no degeneracy was identified. This could be worth looking at in further investigation of the work.

Figure 4.3 shows the distributions of the number of Nash equilibria as per the number of players. This visualisation turns out to not be extremely revealing — possibly as an effect of degeneracy — however some insights can be found here. Observe, all of the distributions observed on this plot have a clear modal value of one, that is, irrespective of the number of players, the majority of games yielded only one equilibria. Moreover, there also seems to be an increase in density around 3 equilibria which becomes more prominent as the number of players increases. As can be seen from the plot, the variance in the number of equilibria increases with the number of players, apart from when there were 6 players (5 opponents of the defector), where the spread is maximum. This could be due to the 39 equilibria gained for one game as discussed in the paragraph above. Looking now at the mean of the distributions, observe that these are also slightly increasing as the number of players increase.



(a) A plot to show the minimum p -thresholds. (b) A plot to show the maximum p -thresholds.

Figure 4.4: Plots to show the p -thresholds for all 1001 sets of tournaments.

4.2 Analysis of the p -Threshold

Firstly, for clarity, here is a restatement of the definition of a p -threshold: The probability of the tournament ending for which the least probability of defection in Nash equilibria is non-zero.

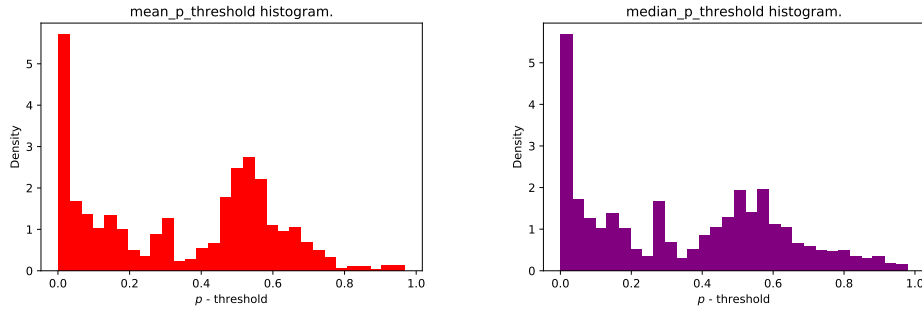
In order to analyse the p -thresholds of the tournaments, a csv file was created¹ containing the minimum, mean, median and maximum probabilities for each set of tournaments. This was in order to gain as much information as possible from tournaments which gave graphs such as Figure 4.1b as described above. Within this file, other than the varying thresholds, the information about the number of players, tournament strategy and level of noise was retained. Moreover, it contained a column which identified whether any of the strategy sets lead to possible degenerate games.

Now, an exploration into the overall p -thresholds is given.

Observe, in Figure 4.4a, the majority of minimum thresholds were less than 0.4, with a clear modal value of 0. That is, in a significant proportion of the tournaments, there was no probability of the game ending for which the probability of defection was zero. Now considering Figure 4.4b, it can be seen that the modal value for the maximum threshold is 1. Also, in comparison with the minimum threshold, there is a larger spread in the density. Yet, there is still a peak around zero as with the minimum threshold data.

Looking at the mean and median threshold data, Figure 4.5, observe that the distributions obtained are similar with, again, a modal value of zero, indicating there is always a chance of defection irrespective of the game ending probability.

¹Please see Appendix ?? for the code used to obtain these files.



(a) A plot to show the mean p -thresholds. (b) A plot to show the median p -thresholds.

Figure 4.5: Plots to show the p -thresholds for all 1001 sets of tournaments.

However, in these histograms there is also a clear peak around a threshold of 0.5. This suggests that some of the tournaments may have several p -thresholds, perhaps due to stochasticity, degeneracy or just inevitable randomness that appears in the tournaments. Indeed, obtaining the minimum and maximum p -thresholds for those tournaments where the mean threshold was within the range $[0.5, 0.6]$, it can be seen that, from Figure 4.6 for a significant proportion, their minimum threshold was around zero and their maximum around one. Moreover, looking closer at three of the tournaments which satisfied the above, there is a clear randomness within the corresponding graphs, Figure ??.

The plots contained in Figure 4.7 were sampled randomly using the random library in Python. What is interesting here is that they all contain varying amounts of additional noise and further analysis into this would be beneficial in seeing if this is one of the main causes for the inaccuracy of the thresholds. For example, looking at how many of the tournaments with this property had an added noise level.

Before continuing onto the general overview of the thresholds for those tournaments which led to definite non-degenerate games, a brief point is made regarding those tournaments which, for all probabilities of the game ending, had no positive probability of defection. Indeed, out of the 1754 total tournaments 753 of them had the above property of no apparent threshold (that is, lie within the unobserved interval of $(0.999, 1)$). That is, the plots obtained for these tournaments were (approximately). Further investigation into these tournaments, as well as those in which the probability of defection was always positive, is highly recommended.

Regarding degeneracy, out of all 1754 tournaments, 372 were highlighted as potentially leading to degenerate games. These are omitted from the following plots

the min / max p -thresholds for tournaments whose mean p -threshold was

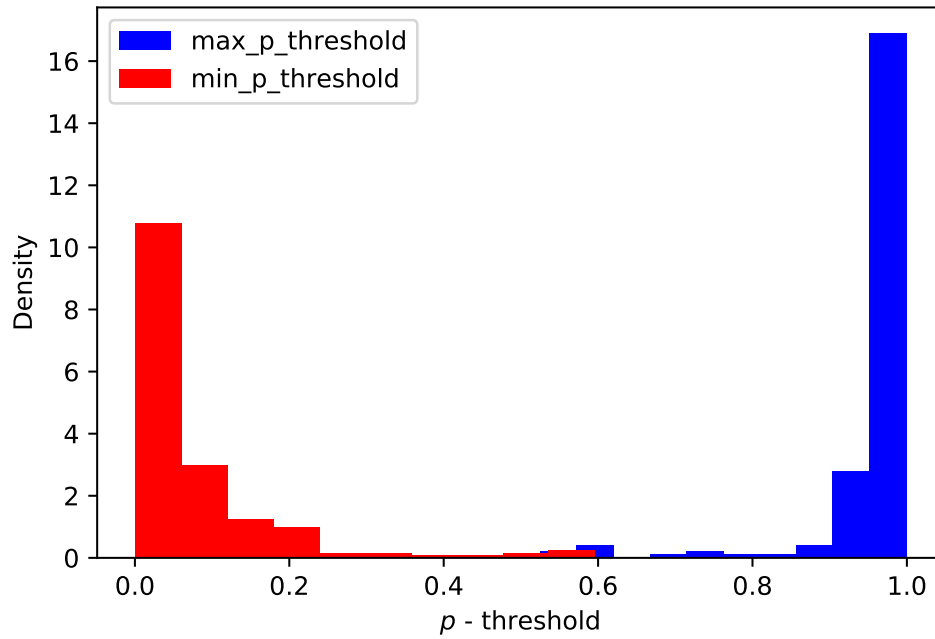


Figure 4.6: A plot to show the minimum and maximum p -thresholds for those tournaments which had an mean threshold within the range $[0.5, 0.6]$.

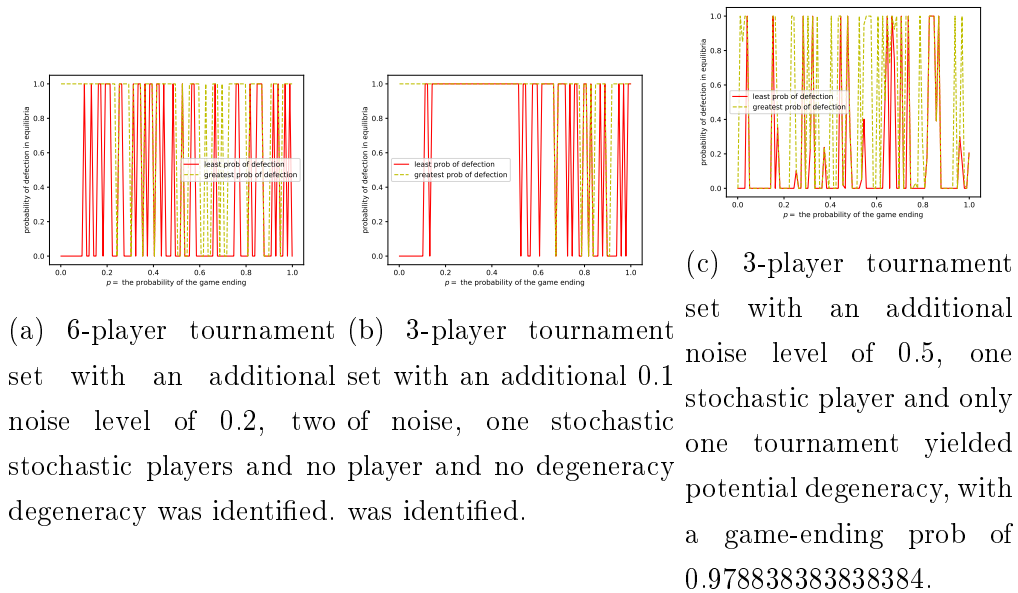
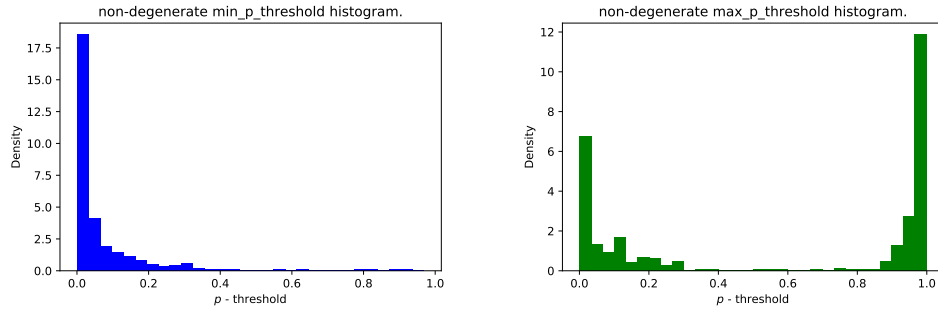
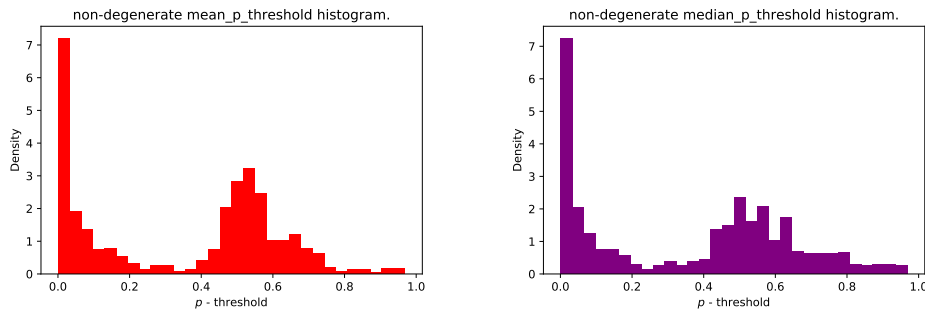


Figure 4.7: Example plots of the tournaments where the mean p -threshold was within the range $[0.5, 0.6]$.



(a) A plot to show the minimum p -thresholds. (b) A plot to show the maximum p -thresholds.

Figure 4.8: Plots to show the p -thresholds for all tournaments which led to non-degenerate games.



(a) A plot to show the mean p -thresholds. (b) A plot to show the median p -thresholds.

Figure 4.9: Plots to show the p -thresholds for all tournaments which led to non-degenerate games.

in order to focus solely on non-degenerate games.

Comparing Figures 4.8a, 4.8b, 4.9a and 4.9b with Figures 4.4a, 4.4b, 4.5a and 4.5b, respectively, it can be seen that, in general, there is no significant change in the distributions of the thresholds. But, there is a more prominent peak in Figure 4.5a around 0.3 than in the corresponding non-degenerate plot of Figure 4.9a. Further work regarding the effects of degeneracy is advised as the above discussion seems to indicate that degeneracy does not seem to have as much of an affect on the thresholds as initially expected.

4.2.1 Effects of the Number of Players

In this section, the p -thresholds will be analysed with respect to the number of opponents the *Defector* played against. Note, in this section, only non-degenerate tournaments will be considered.

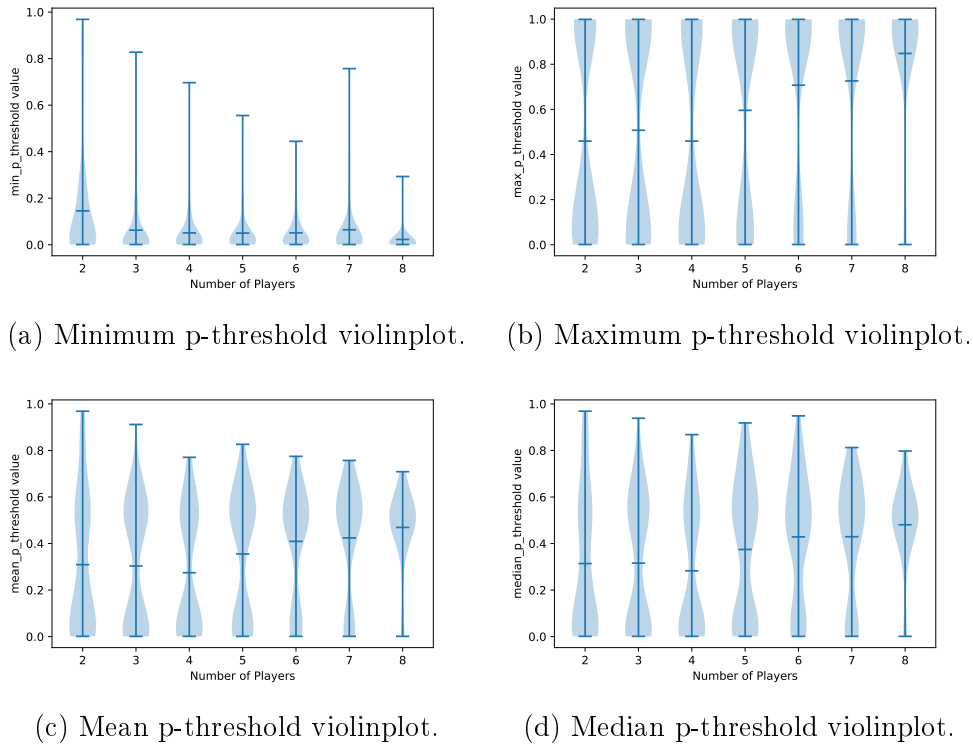


Figure 4.10: Violinplots of the thresholds for each number of opponents.

From Figure 4.10c, it can be seen that the distributions of the minimum p -thresholds with respect to the number of players all have a modal value around 0. However, apart from the 7-player tournaments, the spread of the distributions decrease, along with the mean values. Considering the maximum thresholds, Figure ??, the distributions become bimodal with mode values of around 0 and 1. But, as the number of players increases the modal value at zero becomes less prominent with the 8-player tournament distribution not having a mode around 0. The variance of the distributions are similar and, apart from 4-player tournaments, the means increase with the number of players. Looking now to the mean and median violinplots, Figures 4.10c and 4.10d respectively, there is no significant difference between the two plots other than the spread is a little larger in the median thresholds. Within the mean plot, Figure 4.10c, it can be seen that the distributions also start off bimodal, at 0 and approximately 0.5. But again, the modal value at zero becomes less distinct with the 8-player tournament distribution being unimodal. The modal value at around 0.5 is a consequence of the reason stated in the previous section. Moreover, observe that, apart from 4-player tournaments, the variance of the distributions seem to decrease with the size of the player set whilst the means increase from a value of approximately 0.3 for 2-player tournaments to around 0.5 for 8-players. Looking at Figure 4.10 as a whole, it is implied that the number of players has no significant effect on the

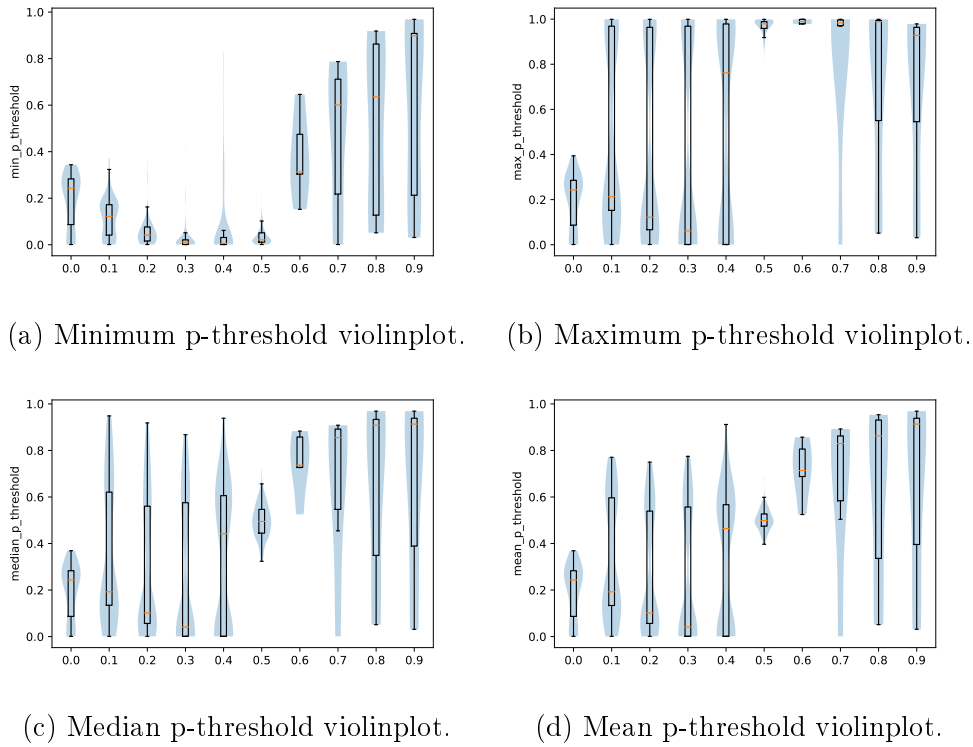


Figure 4.11: Violinplots of the thresholds for each level of noise.

value of the p-threshold.

4.2.2 Effects of Noise

Here, an analysis on the effects of noise on the p -threshold is provided. The addition of noise to a tournament indicates that, with a certain probability, the action of a particular strategy is altered [35]. That is, an action of *coop* changes to *defect* and vice versa.

Figure 4.11a, shows the distribution of the minimum p-thresholds for each level of additional noise added to the tournaments. Here, it can be noted that the distributions of noise levels at least 0.6 have a large variance, indicating that adding a too large noise probability is highly random and thus no conclusions can be drawn. On the other hand, considering the noise levels less than 0.6, observe that the mean p-thresholds decrease from around 0.25 to approximately 0 as the noise increases. These distributions are also clearly unimodal. Looking now at Figure 4.11b, it can be seen that the majority of distributions have a much larger spread here varying over the full spectrum of game-ending probabilities and similar observations can be made from Figures 4.11c and 4.11d.

Figure 4.12 shows an example of the same tournament set through a variety

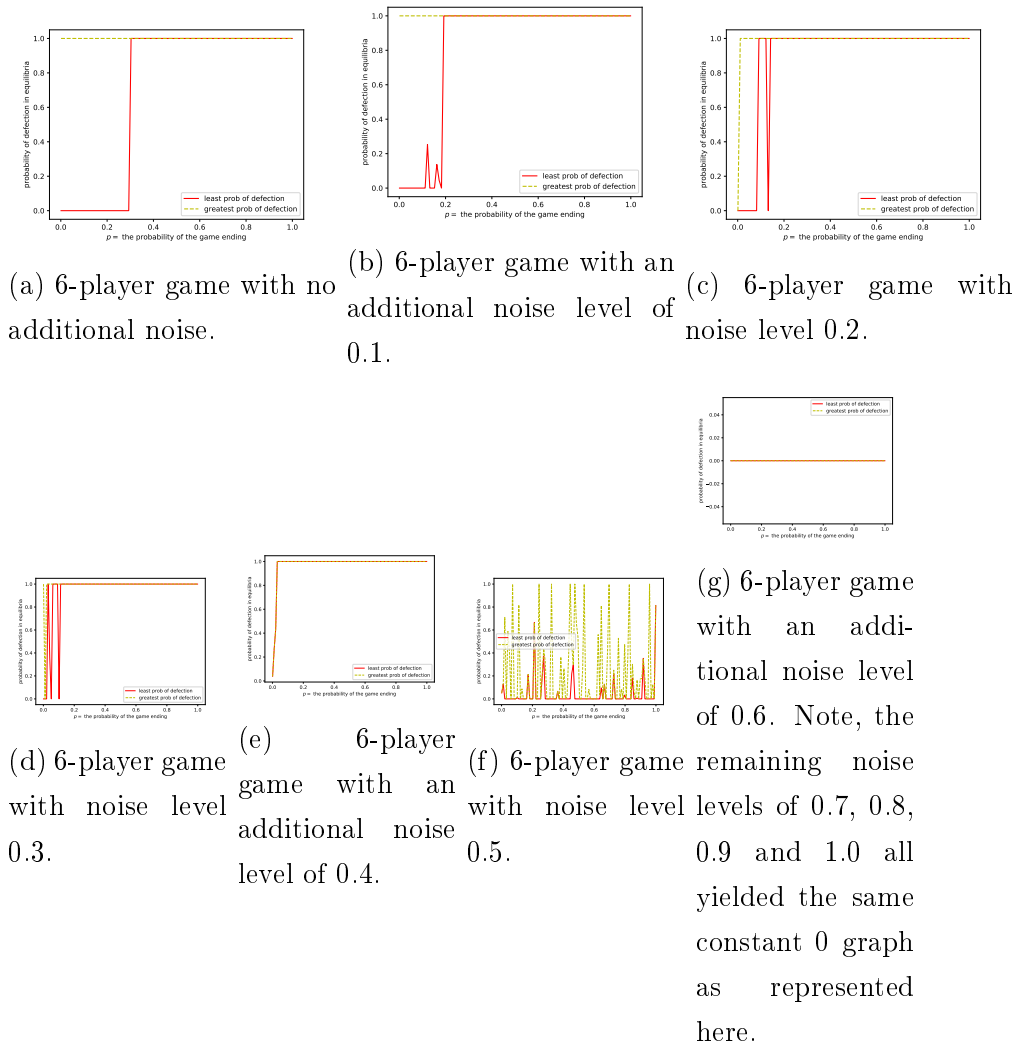


Figure 4.12: Observation of one 6-player tournament set through the varying levels of additional noise. There was one stochastic player and 13 out of the 1100 tournaments played yielded potential degenerate games. The opponents here were: *Getzler*; *Punisher*; *Forgiver*; *GrudgerAlternator*; and *GraaskampKatzen*.

of differing noise levels. Indeed, here, it can clearly be seen that the amount of noise does effect the p -threshold. However, this is to be expected since, by definition, as the noise level increases, the *Defector* will be observed as similar to the *Cooperator* when there is no additional noise.

Therefore, on the whole, there are not many significant conclusions that can be made here. This implies that the addition of noise to an already random tournament obscures any possible visions of the threshold, especially as the magnitude increases.

4.3 Conclusions and Further Work

In this section, the beginnings of an analysis into the p -thresholds was discussed. The effects of the number of players and level of additional noise on these thresholds were the main focus with a brief discussion regarding the degeneracy of games also given. This turned out to be a non-trivial task due to the proportion of tournament sets which yielded potential degeneracy at certain game ending probabilities and the inevitability of randomness within the tournaments. However a few points of interest were highlighted and these are summarised here.

Firstly, the potential degeneracy of games yielded by the tournaments, which was highlighted as a potential factor, at first glance appears to not create as much of an issue as originally thought. The histograms of the p -thresholds, when the tournaments including potential degeneracy were omitted, did not have any significant changes when compared to the original histograms of all tournaments. However, further exploration is advised here. Regarding the number of players in a tournament, it was initially hypothesised that this would be a key factor in the variability of the p -threshold. However, on obtaining the distributions of the thresholds for tournament sets of size one to eight, it was implied that the number of players does not have a significant impact on the value of the threshold. Finally, the effect of additional noise on the p -threshold was analysed. Here, it was observed that, as expected, the level of additional noise did affect the p -threshold however there was no significant trends appearing out of the randomness.

As stated above, this is only the very start of an analysis into the p -thresholds described by the ‘original’ Folk Theorem and the effects of the varying environmental factors which appear in tournaments of the IPD. Thus many questions regarding this are still to be researched and a few recommendations regarding further work are now given. Firstly, it was observed that there were a significant proportion of tournaments for which the graph remained constant at zero or one. That is, the p -threshold was not identified using the precision of game-ending probabilities chosen and therefore must lie within the intervals $(0, 0.001)$ or $(0.999, 1)$, respectively. Hence, these tournament sets could be rerun with a much finer precision within the appropriate intervals to highlight exactly what is happening here. Also, an analysis into the characteristics of the strategies involved and the additional noise levels included in the tournaments could provide a clearer insight into potential reasons for this. Moreover, with regards to analysing the characteristics of players, it is suggested that those tournament sets in which stochastic players were included could be removed and the tournaments

rerun. This could help in revealing whether the stochasticity of the player has any effect on the threshold, as the author has hypothesised.

To check the reliability of the data collected, a second experiment is recommended using a different algorithm for calculating the Nash equilibria, for example vertex enumeration. This could be used in comparison with the data already collected to identify whether the algorithms are producing the same Nash equilibria or, more importantly, whether they identified the same games as being degenerate. Furthermore, it is suggested that this experiment be executed with a larger number of tournaments repeats (greater than 500) to observe whether this ‘smooths’ the payoff matrices with greater success to enable for a clearer visualisation of the p-thresholds. Finally, some multivariate data analysis of the results, for example regression, could provide some more insights into this topic.

Overall, this chapter has been successful in visualising the Folk Theorem using the data collection setup as explained in Chapter ?? and using the plots obtained as in Figure 4.1a.

Chapter 5

Conclusions and Recommendations

This project looked into the Folk Theorem for the IPD and, in particular, the threshold at which the probability of defection is less than that given in the stage game or the finitely repeated game. The three aims of the study were stated in Chapter 1. These are provided here for convenience:

1. To look thoroughly into the recent (and past) literature of research already produced in the area of Folk Theorems;
2. To execute a large experiment involving many tournaments of the IPD with different types and numbers of players to obtain graphs similar to those in Figure ??; and
3. To perform analyses on where the p -threshold seem to lie and whether it is affected by the different environments of the tournaments (that is, differing number of players, stochasticity within the tournament itself and the players etc.).

Therefore, in this chapter, a brief section discussing the results of each aim is provided. Then, a few shortcomings of the project are mentioned, before highlighting areas in which the research could be expanded further.

5.1 Conclusions

In this section, the results of each aim is explored with regards to what was found and how well they were addressed. Overall, the first two aims were successfully achieved with a well rounded knowledge in the recent research of folk theorems

and a successful empirical experiment carried out. However, maximal results were not obtained for the final aim due to time limitations and the unforeseen non-triviality of the proportion of degeneracy and randomness.

5.1.1 Aim 1

The first aim was addressed in Chapter ?? . Here, a literature search on “folk theorem” yielded a vast range of results which spanned the last fifty years. It was discussed that the true origin of the ‘folk theorem’ is unknown however its statement and proofs appear in written work since the 1970s. These papers focused on infinitely repeated games with subgame perfect equilibria. Following this, many generalisations and refinements of the folk theorem were explored. Varying types of games, for which folk theorems have been considered, were discussed. These included: games with complete information, games with imperfect private monitoring and games with communication, among others. Within these papers, it was also discovered that other equilibria besides the more common subgame perfect equilibria have been considered. Some examples are: sequential equilibria, belief-free equilibria and type-k equilibria. Throughout, there appeared to be two main assumptions required for a folk theorem to exist. These regarded full dimensionality and the number of signals in proportion to the number of actions. Since these assumptions are quite restricting, a few papers have attempted to find situations in which they can be weakened, whilst still obtaining a folk theorem. On the other hand, an area of research has developed which show the instability of equilibria and / or the violation of folk theorems. The chapter is concluded by a discussion on recent applications of the folk theorem to computing, multi-agent systems and quantum transportation. Finally, to the best of the author’s knowledge, no folk theorem experiments of the same size, as executed in this study, have been computed before.

5.1.2 Aim 2

Chapter ?? discusses the set-up and execution of the IPD experiment referred to in Aim 2. This included justifications for the use of certain software and methods as well as a detailed view of the data collection algorithm. Within this, reasons were given as to why the particular attributes of the tournaments were collected. Following this an exploration into the varying file types available for data collection was provided. After considering the positives and drawbacks it was decided that a binary file, in particular a relational database file, would be the most appropriate format. The chapter then continues with an explanation

of the software implementation and execution of the aforementioned algorithm. It provides details as to how good software development principles were followed and the programs utilised (for a more detailed overview which is not as involved as the chapter, see ??). The actual experiment was executed remotely over a few weeks and how this was achieved was explained, before the issues, how to compute the Nash equilibria and degeneracy, were addressed. Out of the three algorithms: Support Enumeration, Vertex Enumeration and Lemke-Howson; it was decided that Support Enumeration would be the main method used in calculating Nash equilibria, due to its robustness. The execution timings of each algorithm were also explored but did not yield any significant differences. To conclude, the difficulties which could be faced due to degeneracy were briefly acknowledged and possible solutions discussed.

5.1.3 Aim 3

The final aim regarding the analysis of p -threshold was addressed in Chapter 4. Firstly, an initial analysis was carried out which identified the main ‘shapes’ of the graphs obtained and detailed the summary statistics. This included looking at characteristics of the strategies as well as the number of equilibria obtained from each tournament. The majority of strategies chosen were deterministic - this is because of the ratio of deterministic to stochastic strategies implemented in the Axelrod library. When analysing the p -threshold, the minimum, mean, median and maximum potential values of the threshold were taken, due to the varying sources of noise obscuring a clear threshold. Here, the focus was on the effects of the number of players and the effects of the tournament noise level. However, this was concluded as a non-trivial task due to the proportion of tournaments which were potentially degenerate and the many sources of randomness within them. As a result of this, significant conclusions were unable to be drawn with the obtained plots not revealing many (if any) trends. Thus, suggestions are provided on how further work regarding this study may reveal more significant information. These are restated in Section 5.3. In conclusion, however, the experiment was successful in providing clear visualisations for the folk theorem and gives insight into a wide range of directions for future research.

5.2 Limitations

In this section a few of the shortcomings of the study are highlighted along with justifications.

Firstly, this project included a very large empirical study on the folk theorem which required the development and implementation of an algorithm into code. This inevitably took a significant amount of time; to ensure the functions were accurate, clear and gave the required information. Therefore, one of the main drawbacks was the time constraint of completing the project. Indeed, after the successful implementation of the data collection algorithm into Python, there was a certain time frame before analysis could start, to ensure enough data had been collected. Then the analysis took another significant portion of the time, especially due to the data's non-triviality.

As a result of this, the amount of data collected was much less than what was required for a large-scale statistical analysis on extremely random data. Also, parameters inputted into the algorithm were restricted by this. Only 500 repetitions of each tournament were performed but this was only able to 'stabilise' some of the payoff matrices, whilst the rest resulted in less clear threshold graphs. Thus, higher repetitions were probably needed but unrealistic for the time frame. Moreover, the eighteen strategies which are classified with a long run-time had to be omitted but, in an ideal situation would have been included to obtain more information.

Furthermore, the analysis which was carried out only touched briefly on some of the potential effects of the environment on the p -threshold. The 'randomness' of the data obtained meant that trends were harder to find and key conclusions could not be drawn. Ideally, many more possible environmental changes, in addition to number of players and level of tournament noise, would have been discussed. Another factor which may have improved the situation, but had to be omitted due to time, was the packaging of the code. Creating a fully working Python package, with all the required functions in, may have made the remote computing easier.

Finally, degeneracy was a major limitation due to its uncertainty. Nashpy only identifies potential degeneracy if it is 'strange'. That is, if the algorithm detects division by zero or 'peculiar' equilibria, for example. Indeed, some games may still be degenerate even if some correct equilibria are identified. This implies that all the discussions regarding degeneracy in Chapter 4 are based on the assumption that support enumeration detects all degeneracy, which is potentially unrealistic.

5.3 Recommendations

Throughout this study, many interesting questions have been raised as potential directions for further work. Hence, in this section, recommendations on how this research could be extended, to provide more insightful results, are given.

Firstly, during analysis (Chapter 4), a significant proportion of the tournament graphs showed no threshold, appearing constant at zero or one. This indicates that the threshold must lie in the intervals $(0, 0.01)$ or $(0.999, 1)$ and hence the precision level was not fine enough. Therefore, it is suggested that these tournament sets are rerun within these intervals with a fine precision in order to obtain more information. Moreover, an analysis of the strategies involved and level of tournament noise in these sets might provide a clearer insight into reasons for this. In addition, rerunning the whole experiment, with a finer precision in the game ending probabilities, may give more accurate thresholds.

In order to explore the effects of stochastic players on the threshold, it is recommended that those tournament sets which contain one (many) stochastic player(s) are rerun without them. If the threshold is different in both of the runs, this could indicate that stochasticity does indeed have an effect.

Recall, vertex enumeration was also used in a separate run of the experiment however, unfortunately, it was omitted from the analysis due to the time restrictions mentioned previously. Having said this, it is highly recommended that these results are looked into (or rerun) as a way of checking the reliability of the collected data. In particular, it could be used in comparison with the support enumeration data to identify whether the same Nash equilibria are yielded or, more importantly, whether the same games are identified as potentially degenerate. However, care is needed when analysing vertex enumeration as it is less robust, as discussed in Chapter ???. Furthermore, increasing the number of tournament repeats is suggested to observe whether this ‘smooths’ the payoff matrices with more success, perhaps, resulting in clearer p -thresholds.

Regarding multivariate data analysis of this large empirical study, it may be insightful to perform a regression or clustering algorithm. This is motivated by the possibility of being able to predict approximately the threshold of a tournament by its characteristics: strategies, level of tournament noise etc. Finally, the experiment could potentially be extended to consider the different ‘types’ of folk theorem discussed in Chapter ??.

References

- [1] Dilip Abreu, Prajit K. Dutta, and Lones Smith. “The Folk Theorem for Repeated Games: A NEU Condition”. In: *Econometrica* 62.4 (1994), pp. 939–948. URL: <https://www.jstor.org/stable/2951739>.
- [2] IB Adeoye et al. “Application of game theory to horticultural crops in south-west Nigeria.” In: *Journal of Agricultural and Biological Science* 7.5 (2012), pp. 372–375.
- [3] N. Alon and G. Kalai. “A Simple Proof of the Upper Bound Theorem”. In: *European Journal of Combinatorics* 6.3 (Sept. 1985), pp. 211–214. DOI: 10.1016/s0195-6698(85)80029-9.
- [4] L. Anderlini, D. Gerardi, and R. Lagunoff. “A "super" folk theorem for dynastic repeated games”. In: *Economic Theory* 37.3 (2008). Cited By :6, pp. 357–394. URL: www.scopus.com.
- [5] VERA ANGELOVA et al. “CAN SUBGAME PERFECT EQUILIBRIUM THREATS FOSTER COOPERATION? AN EXPERIMENTAL TEST OF FINITE-HORIZON FOLK THEOREMS”. In: *Economic Inquiry* 51.2 (Sept. 2011). cited By 2, pp. 1345–1356. DOI: 10.1111/j.1465-7295.2011.00421.x. URL: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-84874353398&doi=10.1111%2Fj.1465-7295.2011.00421.x&partnerID=40&md5=83e392d6d27e1febb1639f619a306357>.
- [6] RJ Aumann and L Shapley. “Long term competition”. In: *A Game Theoretic Analysis* (1976).
- [7] Robert Axelrod. “Effective choice in the prisoner’s dilemma”. In: *Journal of conflict resolution* 24.1 (1980), pp. 3–25.
- [8] Kalid Azad. *A little diddy about binary file formats*. [Accessed on:04/03/2020]. Better Explained. Apr. 16, 2005. URL: <https://betterexplained.com/articles/a-little-diddy-about-binary-file-formats/>.
- [9] Margherita Barile. *Open Cover*. [Accessed on; 06/01/2020]. Wolfram MathWorld. URL: <http://mathworld.wolfram.com/OpenCover.html>.

- [10] Ned Batchelder. *Coverage.py 5.0.3*. Version 5.0.3. Jan. 12, 2020. URL: <https://github.com/nedbat/coveragepy>.
- [11] Michael Bayer. “SQLAlchemy”. In: *The Architecture of Open Source Applications Volume II: Structure, Scale, and a Few More Fearless Hacks*. Ed. by Amy Brown and Greg Wilson. aosabook.org, 2012. URL: <http://aosabook.org/en/sqlalchemy.html>.
- [12] Michael Bayer. “SQLAlchemy”. In: *The Architecture of Open Source Applications Volume II: Structure, Scale, and a Few More Fearless Hacks*. Ed. by Amy Brown and Greg Wilson. aosabook.org, 2012. URL: <http://aosabook.org/en/sqlalchemy.html>.
- [13] Jean-Pierre Benoit and Vijay Krishna. “Finitely Repeated Games”. In: *Econometrica* 53.4 (July 1985), p. 905. DOI: 10.2307/1912660.
- [14] Axel Bernergård. “Self-control problems and the folk theorem”. In: *Journal of Economic Behavior & Organization* 163 (July 2019), pp. 332–347. DOI: 10.1016/j.jebo.2019.05.004.
- [15] V. Bhaskar. “Informational Constraints and the Overlapping Generations Model: Folk and Anti-Folk Theorems”. In: *Review of Economic Studies* 65.1 (1998). Cited By :42, pp. 135–149. URL: www.scopus.com.
- [16] Juan I. Block and David K. Levine. “A folk theorem with codes of conduct and communication”. In: *Economic Theory Bulletin* 5.1 (Sept. 2016), pp. 9–19. DOI: 10.1007/s40505-016-0107-y.
- [17] A. Brøndsted. *An Introduction to Convex Polytopes*. Graduate Texts in Mathematics. Springer New York, 2012. ISBN: 9781461211488. URL: <https://books.google.co.uk/books?id=7PXxBwAAQBAJ>.
- [18] Alex Burke. *The Advantages of a Relational Database Over a Flat File*. [Accessed on: 04/03/2020]. bizfluent. Sept. 26, 2017. URL: <https://bizfluent.com/list-7269497-advantages-database-over-flat-file.html>.
- [19] Stephan C. Carlson. *Brouwer’s fixed point theorem*. [Accessed on: 06/01/2020]. Encyclopaedia Britannica. Sept. 13, 2016. URL: <https://www.britannica.com/science/Brouwers-fixed-point-theorem>.
- [20] Sylvain Chassang and Satoru Takahashi. “Robustness to incomplete information in repeated games”. In: *Theoretical Economics* 6.1 (Jan. 2011). cited By 12, pp. 49–93. DOI: 10.3982/te795. URL: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-78651454708&doi=10.3982%2fTE795&partnerID=40&md5=53dbe8255f303468a1e5d95095c08326>.

- [21] Bor-Sen Chen, Chia-Hung Chang, and Hsiao-Ching Lee. “Robust synthetic biology design: stochastic game theory approach”. In: *Bioinformatics* 25.14 (2009), pp. 1822–1830.
- [22] Ankur Chowdhary et al. “Dynamic Game based Security framework in SDN-enabled Cloud Networking Environments”. In: *Proceedings of the ACM International Workshop on Security in Software Defined Networks & Network Function Virtualization - SDN-NFVSec '17*. ACM Press, 2017. DOI: 10.1145/3040992.3040998.
- [23] E. F. Codd. “A Relational Model of Data for Large Shared Data Banks”. In: *Software Pioneers*. Springer Berlin Heidelberg, 2002, pp. 263–294. DOI: 10.1007/978-3-642-59412-0_16.
- [24] The Axelrod project developers. *Axelrod: v4.7.0*. Apr. 2016. DOI: 10.5281/zenodo.3517155. URL: <http://dx.doi.org/10.5281/zenodo.3517155>.
- [25] Oxford English Dictionary. *game theory, n*. Online. Accessed: 08.10.2019. Mar. 2013. URL: <https://www.oed.com/view/Entry/319028?redirectedFrom=game+theory&>.
- [26] Steven N. Durlauf and Lawrence E. Blume. “Repeated Games: Imperfect Monitoring”. In: *The New Palgrave Dictionary of Economics*. Ed. by Steven N. Durlauf and Lawrence E. Blume. Springer, 2016, p. 102. ISBN: 9781349588022.
- [27] Prajit K. Dutta. “A Folk Theorem for Stochastic Games”. In: *Journal of Economic Theory* 66 (1995), pp. 1–32.
- [28] Jeffrey C. Ely, Johannes Horner, and Wojciech Olszewski. “Belief-Free Equilibria in Repeated Games”. In: *Econometrica* 73.2 (Mar. 2005). cited By 70, pp. 377–415. DOI: 10.1111/j.1468-0262.2005.00583.x. URL: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-27744460948&doi=10.1111%2fj.1468-0262.2005.00583.x&partnerID=40&md5=7db49053936b7d19cd405059a246462e>.
- [29] Jeffrey C. Ely and Juuso Välimäki. “A Robust Folk Theorem for the Prisoner’s Dilemma”. In: *Journal of Economic Theory* 102.1 (Jan. 2002). cited By 95, pp. 84–105. DOI: 10.1006/jeth.2000.2774. URL: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-0036167111&doi=10.1006%2fjeth.2000.2774&partnerID=40&md5=b8cc560b170e8796d5239cbd7ef72ef9>.
- [30] James W Friedman. “A non-cooperative equilibrium for supergames”. In: *The Review of Economic Studies* 38.1 (1971), pp. 1–12.

- [31] James W. Friedman. “A Non-cooperative Equilibrium for Supergames”. In: *The Review of Economic Studies* 38.1 (Jan. 1971), p. 1. DOI: 10.2307/2296617.
- [32] Drew Fudenberg and David K. Levine. “The Nash-threats folk theorem with communication and approximate common knowledge in two player games”. In: *A Long-Run Collaboration on Long-Run Games*. Vol. 132. 1. WORLD SCIENTIFIC, Dec. 2008, pp. 331–343. DOI: <http://dx.doi.org/10.1016/j.jet.2005.08.006>.
- [33] Drew Fudenberg and Eric Maskin. “The Folk Theorem in Repeated Games with Discounting or with Incomplete Information”. In: *Econometrica* 54.3 (May 1986), p. 533. DOI: 10.2307/1911307.
- [34] Takako Fujiwara-Greve and Yosuke Yasuda. “The Folk Theorem in Repeated Games with Endogenous Termination”. In: *SSRN Electronic Journal* (2018). DOI: 10.2139/ssrn.3267427.
- [35] Nikoleta E. Glynatsi and Vincent A. Knight. *A meta analysis of tournaments and an evaluation of performance in the Iterated Prisoner’s Dilemma*. 2020. arXiv: 2001.05911 [cs.GT].
- [36] Olivier Gossner. “Overlapping Generation Games with Mixed Strategies”. In: *Mathematics of Operations Research* 21.2 (May 1996), pp. 477–486. DOI: 10.1287/moor.21.2.477.
- [37] Yuval Heller. “Instability of belief-free equilibria”. In: *Journal of Economic Theory* 168 (Mar. 2017). cited By 1, pp. 261–286. DOI: 10.1016/j.jet.2017.01.001. URL: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85008894373&doi=10.1016%2fj.jet.2017.01.001&partnerID=40&md5=e6021d15598968bf7d198f9c0619e9b7>.
- [38] Michael D Hirsch, Christos H Papadimitriou, and Stephen A Vavasis. “Exponential lower bounds for finding Brouwer fix points”. In: *Journal of Complexity* 5.4 (Dec. 1989), pp. 379–416. DOI: 10.1016/0885-064x(89)90017-4.
- [39] Johannes Hörner and Wojciech Olszewski. “The Folk Theorem for Games with Private Almost-Perfect Monitoring”. In: *Econometrica* 74.6 (2006), pp. 1499–1544. URL: <https://www.jstor.org/stable/4123082>.
- [40] K. Ikeda. “Foundation of quantum optimal transport and applications”. In: *Quantum Information Processing* 19.1 (2020). cited By 0. DOI: 10.1007/s11128-019-2519-8. URL: <https://www.scopus.com>.

- [41] Nishtha Jatana et al. “A Survey and Comparison of Relational and Non-Relational Database”. In: *International Journal of Engineering Research and Technology* 01 (06 Aug. 2020). ISSN: 2278-0181.
- [42] Rafael C. Jiménez et al. “Four simple recommendations to encourage best practices in research software”. In: *F1000Research* 6 (June 2017), p. 876. DOI: 10.12688/f1000research.11407.1.
- [43] M. Kandori. “Randomization, communication, and efficiency in repeated games with imperfect public monitoring”. English. In: *Econometrica* 71.1 (2003). Cited By :13, pp. 345–353. URL: www.scopus.com.
- [44] Michihiro Kandori and Ichiro Obara. “Less is more: an observability paradox in repeated games”. In: *International Journal of Game Theory* 34.4 (Sept. 2006). cited By 5, pp. 475–493. DOI: 10.1007/s00182-006-0032-7. URL: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-33750923312&doi=10.1007%2fs00182-006-0032-7&partnerID=40&md5=d8b2b09a452a7522691b58f57a8abf89>.
- [45] Vincent Knight. *Chapter 10 - Infinitely Repeated Games*. [Accessed on: 15/01/2020]. 2017. URL: https://vknight.org/Year_3_game_theory_course/Content/Chapter_10_Infinetely_Repeated_Games/.
- [46] Vincent Knight. *Chapter 2 - Normal Form Games*. Online. [Accessed on: 13/10/2019]. 2017. URL: https://vknight.org/Year_3_game_theory_course/Content/Chapter_02-Normal_Form_Games/.
- [47] Vincent Knight. *Chapter 3 - Rationalisation*. Online. Accessed on: 23/10/2019. 2019. URL: <https://vknight.org/gt/chapters/03/>.
- [48] Vincent Knight. *Chapter 5 - Support Enumeration*. Online. [Accessed on: 16/03/2020]. 2019. URL: <https://vknight.org/gt/chapters/05/>.
- [49] Vincent Knight. *Chapter 8 - Repeated Games*. [Accessed on: 08/01/2020]. 2019. URL: <https://vknight.org/gt/chapters/08/>.
- [50] Vincent Knight. *Chapter 9 - Finitely Repeated Games*. Online. Accessed on 06/01/2020. 2017. URL: https://vknight.org/Year_3_game_theory_course/Content/Chapter_09_Finitely_Repeated_Games/.
- [51] Holger Krekel et al. *pytest x.y*. 2004. URL: <https://github.com/pytest-dev/pytest>.
- [52] Lukasz Langa. *Black: v19.10b0*. 2019.

- [53] Jiawei Li and Graham Kendall. “On Nash Equilibrium and Evolutionarily Stable States That Are Not Characterised by the Folk Theorem”. In: *PLOS ONE* 10.8 (Aug. 2015). Ed. by Pablo Brañas-Garza, e0136032. DOI: 10.1371/journal.pone.0136032.
- [54] Rui Li. “Sufficient communication in repeated games with imperfect private monitoring”. In: *Economics Letters* 108.3 (Sept. 2010), pp. 322–326. DOI: 10.1016/j.econlet.2010.06.005.
- [55] Xiaoxi Li. “The Folk Theorem for Repeated Games With Time-Dependent Discounting”. In: *SSRN Electronic Journal* (2019). DOI: 10.2139/ssrn.3407197.
- [56] Xiannuan Liang and Yang Xiao. “Game theory for network security”. In: *IEEE Communications Surveys & Tutorials* 15.1 (2012), pp. 472–486.
- [57] Nicholas Marriott. *tmux*. URL: <https://github.com/tmux/tmux>.
- [58] Michael Maschler, Eilon Solan, and Shmuel Zamir. *Game Theory*. Cambridge University Press, 2013. DOI: 10.1017/CB09780511794216.
- [59] J. Masso and R. W. Rosenthal. “More on the "anti-folk theorem"”. In: *Journal of Mathematical Economics* 18.3 (1989). Cited By :7, pp. 281–290. URL: www.scopus.com.
- [60] Hitoshi Matsushima. “Repeated Games with Private Monitoring: Two Players”. In: *Econometrica* 72.3 (May 2004). cited By 40, pp. 823–852. DOI: 10.1111/j.1468-0262.2004.00513.x. URL: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-2642524464&doi=10.1111%2fj.1468-0262.2004.00513.x&partnerID=40&md5=a972047919e3b64a6e8a254e603c11e7>.
- [61] John Nash. “Non-cooperative games”. In: *Annals of mathematics* (1951), pp. 286–295.
- [62] John F Nash et al. “Equilibrium points in n-person games”. In: *Proceedings of the national academy of sciences* 36.1 (1950), pp. 48–49.
- [63] Ameya Nayak, Anil Poriya, and Dikshay Poojary. “Type of NOSQL Databases and its Comparison with Relational Databases”. In: *International Journal of Applied Information Systems* 5.4 (Mar. 2013). ISSN: 2249-0868.
- [64] Noam Nisan et al. *Algorithmic Game Theory*. With a forew. by Christos H. Papadimitriou. Cambridge University Press, 2007. ISBN: 978-0-521-87282-9.
- [65] Ichiro Obara. “Folk theorem with communication”. In: *Journal of Economic Theory* 144.1 (Jan. 2009), pp. 120–134. DOI: 10.1016/j.jet.2007.08.005.

- [66] Oracle. *What is a Relational Database*. [Accessed on 02/03/2020]. Oracle. 2020. URL: <https://www.oracle.com/uk/database/what-is-a-relational-database/>.
- [67] ostezer and Mark Drake. *SQLite vs MySQL vs PostgreSQL: A Comparison Of Relational Database Management Systems*. [Accessed on: 02/03/2020]. Digital Ocean. Mar. 2019. URL: <https://www.digitalocean.com/community/tutorials/sqlite-vs-mysql-vs-postgresql-a-comparison-of-relational-database-management-systems>.
- [68] Christos H Papadimitriou. “On the complexity of the parity argument and other inefficient proofs of existence”. In: *Journal of Computer and system Sciences* 48.3 (1994), pp. 498–532.
- [69] Juan Parras and Santiago Zazo. “A distributed algorithm to obtain repeated games equilibria with discounting”. In: *Applied Mathematics and Computation* 367 (Feb. 2020), p. 124785. DOI: 10.1016/j.amc.2019.124785.
- [70] Marcin Peřski. “An anti-folk theorem for finite past equilibria in repeated games with private monitoring”. In: *Theoretical Economics* 7.1 (Jan. 2012), pp. 25–55. DOI: 10.3982/te760.
- [71] Mauricio Piacentini et al. *DB browser for SQLite*. 2015.
- [72] Safraz Rampersaud, Lena Mashayekhy, and Daniel Grosu. “Computing Nash Equilibria in Bimatrix Games: GPU-Based Parallel Support Enumeration”. In: *IEEE Transactions on Parallel and Distributed Systems* 25.12 (Dec. 2014), pp. 3111–3123. DOI: 10.1109/tpds.2014.2307887.
- [73] Guido van Rossum, Barry Warsaw, and Nick Coghlan. *PEP 8 – Style Guide for Python Code*. Python. July 5, 2001. URL: <https://www.python.org/dev/peps/pep-0008/>.
- [74] Ariel Rubinstein. “Equilibrium in supergames with the overtaking criterion”. In: *Journal of Economic Theory* 21.1 (Aug. 1979), pp. 1–9. DOI: 10.1016/0022-0531(79)90002-4.
- [75] Geir Kjetil Sandve et al. “Ten Simple Rules for Reproducible Computational Research”. In: *PLoS Computational Biology* 9.10 (Oct. 2013). Ed. by Philip E. Bourne, e1003285. DOI: 10.1371/journal.pcbi.1003285.
- [76] Raimund Seidel. “The upper bound theorem for polytopes: an easy proof of its asymptotic version”. In: *Computational Geometry* 5.2 (Sept. 1995), pp. 115–116. DOI: 10.1016/0925-7721(95)00013-y.
- [77] John Spacey. *9 Types of Binary File*. [Accessed on: 04/03/2020]. Simplicable. May 14, 2017. URL: <https://simplicable.com/new/binary-file>.

- [78] V. Srivastava et al. “Using game theory to analyze wireless ad hoc networks”. In: *IEEE Communications Surveys Tutorials* 7.4 (Apr. 2005), pp. 46–56. DOI: 10.1109/COMST.2005.1593279.
- [79] SSH.COM. *SSH tunnel*. [Accessed on: 05/03/2020]. SSH Communications Security, Inc. Oct. 11, 2016. URL: <https://www.ssh.com/ssh/tunneling>.
- [80] Christopher Stover. *Quasi-Concave Function*. [Accessed on: 15/01/2020]. Wolfram Mathworld. URL: <http://mathworld.wolfram.com/Quasi-ConcaveFunction.html>.
- [81] Techopedia. *Flat File*. [Accessed 04/03/2020]. Techopedia. Oct. 31, 2011. URL: <https://www.techopedia.com/definition/25956/flat-file>.
- [82] The Nashpy project developers. *Nashpy: v0.0.19*. June 2019. DOI: 10.5281/zenodo.3256475. URL: <http://dx.doi.org/10.5281/zenodo.3256475>.
- [83] Steve Thomas. *What are the advantages and disadvantages of using the flat file data model?* [Accessed on: 04/03/2020]. Quora. Dec. 2018. URL: <https://www.quora.com/What-are-the-advantages-and-disadvantages-of-using-the-flat-file-data-model>.
- [84] John Von Neumann, Oskar Morgenstern, and Harold William Kuhn. *Theory of games and economic behavior (commemorative edition)*. Princeton university press, 2007.
- [85] Stéfan van der Walt, S Chris Colbert, and Gaël Varoquaux. “The NumPy Array: A Structure for Efficient Numerical Computation”. In: *Computing in Science & Engineering* 13.2 (Mar. 2011), pp. 22–30. DOI: 10.1109/mcse.2011.37.
- [86] Jin Wang et al. “Infinitely repeated game based real-time scheduling for low-carbon flexible job shop considering multi-time periods”. In: *Journal of Cleaner Production* 247 (Feb. 2020), p. 119093. DOI: 10.1016/j.jclepro.2019.119093.
- [87] Shuyin Wang and Limei Jiang. “Study of Agent Cooperation Incentive Strategy Based on Game Theory in Multi-Agent System”. In: *Lecture Notes in Electrical Engineering*. Vol. 463. cited By 0. Springer Singapore, June 2018, pp. 1871–1878. DOI: 10.1007/978-981-10-6571-2_227. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85048687335&doi=10.1007%2f978-981-10-6571-2_227&partnerID=40&md5=6988d25bb46e794e764ed6761aa9a99e.

- [88] James N. Webb. *Game Theory: Decisions, Interaction and Evolution*. Springer Undergraduate Mathematics Series. Springer, 2007. ISBN: 978-1-84628-423-6.
- [89] Eric W. Weisstein. *Compact Space*. [Accessed on: 06/01/2020]. Wolfram Mathworld. URL: <http://mathworld.wolfram.com/CompactSpace.html>.
- [90] Eric W. Weisstein. *Convex*. [Accessed on: 06/01/2020]. Wolfram MathWorld. URL: <http://mathworld.wolfram.com/Convex.html>.
- [91] Q. Wen. “A folk theorem for repeated sequential games”. In: *Review of Economic Studies* 69.2 (2002). Cited By :29, pp. 493–512. URL: www.scopus.com.
- [92] Greg Wilson et al. “Best Practices for Scientific Computing”. In: *PLoS Biology* 12.1 (Jan. 2014). Ed. by Jonathan A. Eisen, e1001745. DOI: 10.1371/journal.pbio.1001745.
- [93] Yuichi Yamamoto. “A limit characterization of belief-free equilibrium payoffs in repeated games”. In: *Journal of Economic Theory* 144.2 (Mar. 2009). cited By 15, pp. 802–824. DOI: 10.1016/j.jet.2008.07.005. URL: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-59749087963&doi=10.1016%2fj.jet.2008.07.005&partnerID=40&md5=7d88a156fde28ffdf65ea6b55>
- [94] Yuichi Yamamoto. “Characterizing belief-free review-strategy equilibrium payoffs under conditional independence”. In: *Journal of Economic Theory* 147.5 (Sept. 2012). cited By 3, pp. 1998–2027. DOI: 10.1016/j.jet.2012.05.016. URL: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-84864804226&doi=10.1016%2fj.jet.2012.05.016&partnerID=40&md5=51959f6f0a44c992e657870ad82e701b>.
- [95] Kiho Yoon. “An Anti-folk Theorem in Overlapping Generations Games with Limited Observability”. In: *Review of Economic Dynamics* 4.3 (July 2001), pp. 736–745. DOI: 10.1006/redy.2000.0127. URL: www.scopus.com.

Appendix A

Computer Report

This short report summarises the computer work involved in the study. A key aim of this project was the development and execution of a large experiment regarding the Iterated Prisoner's Dilemma and the Folk Theorem. As a result of this, much focus was given to ensuring good software development principles [42, 75, 92] were followed and hence only a brief overview will be provided here. The interested reader is encouraged to read Chapter ?? of the main report, where full details are given.

A.1 The Overall Program

Firstly, the pseudo-code of the program written is provided to give a general idea of the purpose. This can be found in Algorithm 5.

The main functions created were implemented in Python. This was due to its versatility and, especially, due to the two game theoretic libraries, Axelrod [24] and Nashpy [82], which provided key functions already implemented. Aside from these, other dependencies of the program include: NumPy [85], Random, Warnings, os and SQLAlchemy [12], plus additional libraries for the analysis (in Chapter ??). This program was implemented through the creation of eight functions to ensure the code was easier to read, debug and test for any current, or future, user. Furthermore, variables and functions were given descriptive names in order to produce self-documenting code.

Observe, Algorithm 5 states the use of a database to retain the information collected. This file type was chosen because it supports out of memory operations and is known to be robust. In particular, the relational database SQLite was utilised due to the existence of Python libraries for accessing the file, its structure

Algorithm 5: Folk Theorem Exploration

input : maximum number of opponents, number of strategy sets
 for each number of opponents, noise levels, game ending
 probabilities, number of repetitions of the tournament, the
 path to the database file, and whether or not support
 enumeration should be used to calculate the Nash
 equilibria.

output: a database containing the results as detailed above.

```

1 while True do
2   for each number of opponents do
3     for each repetition with the same number of strategies do
4       Randomly select a set of opponents and add in the
         Defector
5       for each noise level do
6         for each game ending probability do
7           Run the IPD tournament
8           Obtain the Nash equilibria and the corresponding
             probabilities of defection using the algorithm
             indicated
9           for each player in the current set do
10            Write the required information to a record in
              the database file.
11          end
12        end
13      end
14    end
15    Repeat
16  end
17 end

```

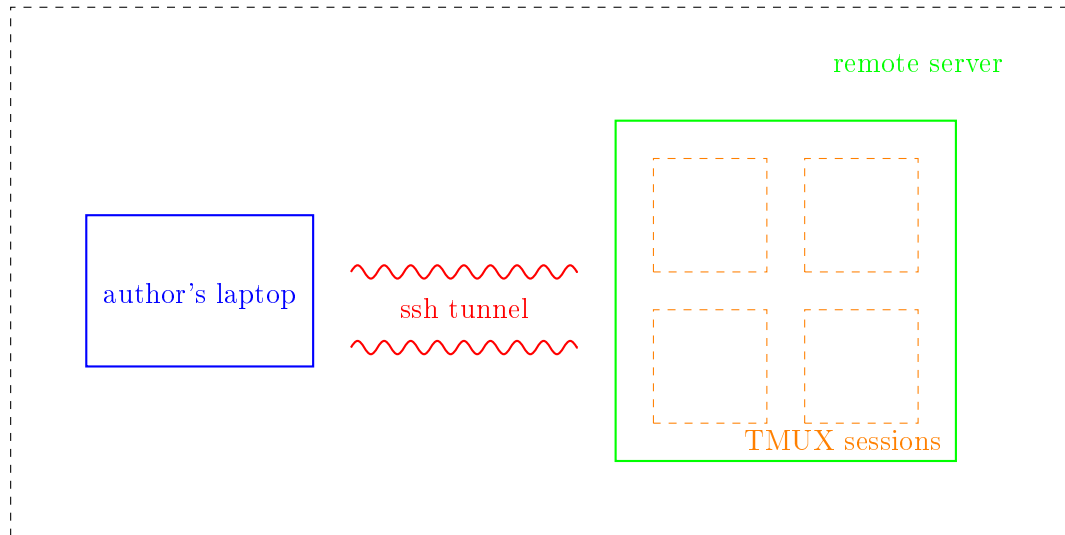


Figure A.1: Representation of how the experiments were run remotely. Note, ‘tmux sessions’ correspond to emulators of terminals.

and its consistency [23, 67].

A.2 Software Development and Remote Computing

Version control software Git (<https://git-scm.com/>) and associated platform GitHub (<https://github.com>) allowed the author to keep track of how the program was evolving. It also ensured a back-up copy was always available, should the system fail and meant that past versions of functions could be easily accessed. Moreover, GitHub acted as an intermediate step between the author’s laptop and remote server, used for the main data collection exercise.

Due to the volume of data which was to be collected, it was decided that the program would be executed remotely. This allowed the code to run uninterrupted for several weeks. However, parallel processing was not required, as the computational time was deemed ‘quick enough’. Figure A.1 provides a visualisation of how the author’s laptop was connected to the School of Mathematics’ headless server, Siren, via a secure shell (SSH) tunnel [79] and terminal multiplexer TMUX [57]. The latter ensured the code kept running whilst the author was disconnected from Siren. Finally, in order to access the database for analysis, the file was compressed and securely transferred from Siren.

A.3 Further Details

In conclusion, there were several components to the computing elements of this study. The reader is referred to the GitHub repository of this project (<https://github.com/shapp> project/), where all the relevant code can be accessed. Also, the data collected during the study is archived at <URL> for future reference.