Lies, Fraud and Game Theory
Using the Peer to Peer Trading Game to Investigate Strategies on Online Marketplaces

Christopher Taylor

Master of Computing in Computer Science with Honours
The University of Bath
May 2015

Lies, Fraud and Game Theory: Using the Peer to Peer Trading Game to Investigate Strategies on Online Marketplaces

Submitted by: Christopher Taylor

Declaration

This dissertation is submitted to the University of Bath in accordance with the requirements of the degree of Bachelor of Science in the Department of Computer Science. No portion of the work in this dissertation has been submitted in support of an application for any other degree or qualification of this or any other university or institution of learning. Except where specifically acknowledged, it is the work of the author.

Signed:

❑ **Retain for Sample Mark**

UNIVERSITY OF BATH **Department of Computer Science** **INDIVIDUAL COURSEWORK** Submission Cover Sheet **Please fill in both columns in BLOCK CAPITALS and post into the appropriate Coursework Submission Box in the Reception area of the Department Office.**	*for office use* **Date and time received** **This section will be retained by the Department Office as confirmation of hand-in**
How to present your work 1. Bind all pages of your assignment (including this submission sheet) so that all pages can be read by the marker without having to loosen or undo the binding. Ensure that the binding you use is secure. Missing pages cannot be marked. 2. If you are required to submit part of the work on a disk, place the disk in a sealed envelope and bind the envelope into the submission. You must keep a copy of your assignment and disk. The original is retained by the Department for scrutiny by External Examiners.	**Declaration** *I certify that I have read and understood the entry in the Department of Computer Science Student Handbook on Cheating and Plagiarism and that all material in this assignment is my own work, except where I have indicated with appropriate references. I agree that, in line with Regulation 15.3(e), if requested I will submit an electronic copy of this work for submission to a Plagiarism Detection Service for quality assurance purposes.*
FAMILY NAME	FAMILY NAME
GIVEN NAME	GIVEN NAME
UNIT CODE	UNIT CODE
UNIT TITLE	UNIT TITLE
DEADLINE TIME & DATE	DEADLINE TIME & DATE
COURSEWORK PART (if applicable)	COURSEWORK PART (if applicable)
SIGNATURE	SIGNATURE

Abstract

Online marketplace systems, such as eBay, are attractive and useful to a wide variety of users. However, the success of these systems attract malicious users: fraudsters, scammers and thieves. We draw from several fields to obtain a more comprehensive view of how a user might interact with these systems. Using this knowledge, we create a model - the Peer to Peer Trading Game (P2PTG) - and synthesise practical experimentation on this model with literature from the social sciences, economics and mathematics in order to better understand and to suggest possible improvements to these systems.

Contents

List of Figures

List of Tables

Acknowledgements

I would like to thank Marina De Vos for her assistance and guidance throughout this project, and to the various people who helped by proofreading and giving advice along the way.

Chapter 1

Introduction

1.1 Problem Background

There are many websites and services that facilitate trade between users, one of the best known of these systems is eBay. These services can be of great value to users for a wide variety of different purposes including selling spare items, picking up items they need cheaply, and even creating an entire business. However it can come at a cost: there are malicious users who attempt to defraud and scam others.

These platforms have several institutional methods of preventing fraud and reimbursing victims, but the feature which most factors into users' decision making is the feedback system. The feedback system gives buyers an opportunity to leave a comment and either positive, negative or neutral feedback on a transaction. All of this information is made available to users when they are deciding whether or not to carry out a transaction - including a feedback score and the seller's percentage of positive feedback. From this, the potential buyer can decide whether or not they trust the user not to defraud them and to deliver the goods in a satisfactory state and time.[1]

We intend to create an abstract model of an online marketplace system in order to explore some of the strategies users could employ on these services to avoid being defrauded. A simple model will be used to begin, with more factors then to be introduced (section 2.8) into agents' decision making processes, exploring factors in the model and in their wider academic context. Rather than running each simulation just once with a randomly generated set of agents, genetic algorithms will be employed in order to refine the strategies and parameters over many generations.

We will draw from a number of different fields to create and analyse our model: we will use the hawk-dove game as a basis for our game theoretic model, and supplement this with knowledge from the social sciences as to how humans behave, trust, and make decisions. Our main focus will be to use the multi-disciplinary literature review in synthesis with experimentation to formulate possible strategies for real users of online marketplaces to avoid being defrauded, and to anticipate how changes in a platform - or in the behaviour of its userbase - could alter the properties of the system. A significant portion of research into online marketplaces are descriptive, while we hope to provide a more comprehensive look into behaviour on these systems.

Parameterising human personality and behaviour meaningfully is difficult, but we believe that our approach will allow us to both explore the properties of online marketplaces and construct methods for future experimentation and research into modelling in other contexts. Online marketplaces as a class of sociotechnical systems have properties that lend themselves to this kind of investigation: the most significant property is that users interact with the system and other users in a limited, set number of ways. Despite these limitations, however, they are still sophisticated and intricate social systems: there are rules and systems in place to enforce them, security systems, and relationships - however ephemeral they may be.

1.2 Project Structure

We begin with a literature review (*Chapter 2*), discussing and collating research from a number of fields, including game theory, the social science of decision making, reputation and trust systems, and genetic algorithms. We will draw knowledge from all of these fields, discussing how they interact with and contradict one another. We will design the Peer-to-Peer Trading Game (P2PTG) as a model for users on an online marketplace system, and use the information gathered from the various fields to suggest possible extensions and factors with which we could experiment.

Using the design as formulated in the literature review, we will implement the P2PTG and conduct a series of experiments (*Chapter 3*) on it. For each factor, we will conduct a series of experiments: establishing the system properties with that factor on the individual level, on the population level, and then how strategies evolve over time using a genetic algorithm.

Finally, *Chapter 4* contains a detailed discussions of our results and conclusions in context, alongside suggestions for future experimentation, extensions, and additional applications of the system. We will discuss the system's place within existing literature.

Chapter 2

Literature Review

2.1 Introduction

This project is inter-disciplinary and thus this literature review will cover a range of topics related to the online marketplace problem. First, we will discuss the Peer-to-Peer Trading Game (P2PTG) - our model - the form it takes and the reasoning behind the design decisions. We will then discuss the game theory behind our model, including the similar games from which we drew inspiration and analysis. To contrast the purely mathematical analysis that is game theory, we will then discuss the sociological and psychological basis for human decision making, including ideas of fairness, altruism and social pressure.

Following this we will discuss mechanisms already in place on online marketplaces for feedback and reputation, as well as common attacks and concerns regarding reputation systems. We will use genetic algorithms to simulate the evolution of strategies in the system, and discuss what they are and why they are useful. To conclude, we will consolidate and discuss the factors we will explore during the experimentation portion of the project, taking all of the different fields into account in order to build an effective model of the underlying problem.

In particular, we will focus on the interplay and contradictions between the various fields we will discuss, and how elements from each can be taken to enhance our model. Of particular importance is the combination of mathematical concepts and human decision making, using the former to simulate the latter, including the limitations of doing so. We will repeatedly discuss the idea that game theoretical and economic mathematics are in no way a full explanation of human rationality, and how the social sciences can be used as a supplement to better model and understand behaviour.

2.2 The Peer-to-Peer Trading Game (P2PTG)

2.2.1 Modelling an Online Trading Platform

Simplification and abstraction are inherent parts of modelling, necessary in order to simulate a real world sociotechnical system with an uncountable number of factors. As such, there are a number of simplifications the model will have relative to the real platforms. In the most basic version that we will use in this paper, they are as follows:

Auction - A considerable number of online marketplaces use auctions as a method of distributing resources. This is further discussed in section 2.2.2.

Transaction content - In the real world problem there are physical goods selected, purchased, packaged and sent, with different goods costing different amounts. The model will simply involve both agents playing "cooperate", "defect" or "decline" and the (static) payoffs being rewarded appropriately.

Symmetry - In the real problem, a transaction involves a buyer and a seller - the model will not have this, it will be simplified to be a symmetric game (section 2.3).

Types of defection - In the real world problem, there are many different ways to defect, such as: not sending goods, sending counterfeit or broken goods, misleading or false item information, and so on. There is also the possibility that a transaction may fall through or a buyer or seller may be unsatisfied due to reasons other than malicious fraud[2]. In the model, there is only one "defect" option to represent all of the malicious actions and there is also no representation of the ways a transaction can fall through when neither parties have malicious intent.

Randomness - In reality, people do not choose to trade with a person at random; they look for items they wish to purchase. However, the model will pick two agents from the list at random to trade with each other.

Temporality - In this model, there is no sense of time other than the distinct unit of one game round. Consequently, there is no such thing as a long-term but casual user, or a newly signed up user who went on a shopping spree, nor other such distinguishing patterns of behaviour which could impact other individual's decision-making.

These simplifications are in place for both for implementational simplicity and for bringing the model in line with other games studied in game theory so that it can be related to and analysed alongside other games. Despite this, the model has a number of key features in common with the original problem:

Trading - At its heart, the model still involves two agents making a conscious decision to cooperate (or not) in a business transaction while having imperfect knowledge of their trading partner.

Leaving feedback - After a transaction, both users have the opportunity to leave feedback on the other user.

Using feedback - Each user has the opportunity to view their trading partner's transaction history and/or feedback score when deciding whether or not to trade with them.

2.2.2 Justification of the Model

As previously discussed, the P2PTG model is a simplification of the underlying problem; this abstraction still leaves the heart of the concept of an online trading platform intact, while omitting auction related features. While removing the auction part from the model does help with simplification, it also omits out one important thing: there are fraudulent attacks on the auction system itself - some sellers may cooperate to artificially inflate prices, for instance. A similar investigation could be carried out on similar lines, investigating which strategies are best for avoiding shill bidding[3] and other auction scams, but in the interest of clarity it will not be further discussed here.

Similarly, other types of fraud that could be carried out are not modelled; in the underlying scenario, there are many ways to attempt to defraud someone - some more complicated than others: non-payment for goods/non-delivery of goods, misrepresentation of items, or shipping damaged/broken goods[2]. All these methods are consolidated into one "defect" option in this model - for simplification, and for the purpose of generalising: it can be assumed that the agent uses a variety of methods. The expected return is as is defined by the payoff matrix as a single figure, and it is assumed that there is a probability function that defines each possible outcome's probability and its corresponding payoff (figure 2.1). It is easier for an agent to make a decision based on an expected value rather than having to model how competent an agent is at fraud (and how competent the other agent thinks they will be).

Nor are the actual goods themselves modelled. Many of the issues reported are with issues with, for instance, the postal service (or at the very least, the seller *claims* that it is - another possible avenue for fraud): late or non-arriving items, items damaged in the post or for which the shipping is over-charged[2]. There is also no distinguishing between goods of different values or modelling of the fact that people tend not to trade randomly - they want a particular item for a particular reason. Again, this is explainable as being an expected value of a probability function reduced to a single figure for simplicity's sake.

The payoff matrix in the formal definition of the model is ultimately arbitrary and subject to change as an experimental variable. In the case of a successful transaction, both agents are assumed to have benefits from gains from trade[4]: an economic concept referring to the difference in how they value the item they traded and the value they traded for. An agent successfully defrauding another is assumed to have made profit greater than that they would have made had they had actually had to trade something for the payoff they received, minus any costs they may have incurred (and the risk of being shut down by the platform, or have money frozen by the payment processor, etc.) Two agents that attempt to defraud each other do not lose very much other than a small amount of time and resources. The payoff matrix is reasonable, although still arbitrary - and rooted in rough estimations of real life value.

With modelling and simplification comes limitations, but we believe that this model has merit. The model and the problem share some fundamental properties (section 2.2.1); the model removes specific circumstances (such as the particulars of an act of fraud) in favour of general utility and cost values, which - at the cost of some depth - makes mathematical and other analyses considerably easier. The model is easier to gather data from than a real system, and also allows the system parameters to be altered in order to test a particular hypothesis.

In our experiments we will be looking at simple strategies and methods of parameterising said strategies and behaviours. We will discuss possible ways a set of numbers can represent an agent's cunning or cautiousness, and how changing the parameters of the system can - over time - influence the strategies and behaviours being exhibited. The P2PTG's extensibility and generalisability will allow it to be used to experiment with a variety of different factors and strategies while still keeping some of the game theoretical simplicity to assist our analysis.

Probability	Outcome	Payoff
0.3	Perfect Crime	5
0.4	Mostly Successful	2
0.3	Failure	-1
-	Expected Value	2

Figure 2.1: Probability function for a fraudulent transaction

2.2.3 P2PTG Definition

Each round of the *Peer-to-Peer Trading Game* (henceforth referred to as *P2PT Game or P2PTG*) will happen as follows:

- N agents (representing users) are selected or generated using one of several possible methods.
- The number of transactions to run - T - is decided.
- T transactions run, each one being carried out as follows:
  - Two random agents from the list are selected.

  – Each choose whether to Cooperate, Defect or Decline. An agent that chooses to cooperate goes into the transaction with good faith, whilst an agent that chooses to defect attempts to scam or cheat the other. Each agent also has the choice to decline a transaction. The payoff for these actions is defined in figure 3.1: it is a symmetric game (the payoff is the same for both players).
  – Each user then has the opportunity to leave feedback about the user they traded with, assuming the transaction went ahead; that is, if neither declined it.

- After all the transactions, each agent's score is computed.

After each round of the game is complete, the results can be analysed and/or a new generation can be bred for the next round.

A1	A2 Cooperate	A2 Defect	A2 Decline
A1 Cooperate	2	-2	0
A1 Defect	4	-1	0
A1 Decline	0	0	0

Figure 2.2: The payoff matrix for a transaction

2.3 Game Theory Analysis

We can discuss the P2PTG, as defined in section 2.2, in terms of its properties and how it relates to other games.

2.3.1 P2PTG: Population Games and Evolutionary Game Theory

The P2PTG is a population game, in which pairs of random individuals are matched against each other to play a single round of the game. After a number of these rounds, the simulation ends and, by some method - which will award greater scores (evolutionary fitness) with a greater share of "genetic" material in the next generation (section 2.7) - a new or modified population is generated. This is archetypal of an evolutionary game[5] as these games allow us to model how players alter their strategies over time, in order to maximise their own interest. This is important in the P2PTG, and in the online marketplace problem that underlies that model, to appropriately simulate the ways that humans respond to changing circumstances.

2.3.2 P2PTG: Symmetry

A game is said to be symmetric - if and only if - all players have the same payoff matrix, or that the payoff is determined only by which strategies are played and not who they are played by. The formulation in terms of a payoff matrix is as in figure 2.3, where E and F are the strategies that can be employed, and a-d are the payoff values.

The P2PTG is symmetrical, which is a simplification of the underlying problem (as discussed in section 2.2), as are all of the other games discussed.

2.3.3 P2PTG: Perfect and Complete Information

There is a distinction between games in which all players have perfect information, and those in which they have complete information. Overlap between these two categories are common, but not necessary:

Figure 2.3: A generic two-player two-option symmetric game.

A1,A2	A2 E	A2 F
A1 E	a,a	b,c
A1 F	c,b	d,d

Complete Information - Refers to a game where all players know all of the strategies and payoffs available to all players[6, p62].

Perfect Information - Refers to a game where all players know all of the previous moves made by all players[7, p45]. This is relevant in iterated games such as the Iterated Prisoner's Dilemma (section 2.4.2) and in the P2PTG.

In the underlying problem, as well as essentially all real world situations/models of the real world, neither of these assumptions are perfectly applicable, but we deal with the two issues differently in the P2PTG.

Our agents in the P2PTG have complete information of the problem - the payoff matrix is fixed throughout the entire game and available to all agents at all times. In the underlying problem this is not entirely the case - while a rough estimate of the payoffs of the opponent can be calculated (how much are their costs likely to be? How much will I be paying them?), the actual figures cannot be so easily divined. This is, however, a fairly trivial assumption to make - while knowing completely the payoff matrices may impact the minutae of an agent's utility function it most probably will not impact large decisions.

On the other hand, the lack of perfect information in both the underlying problem and in the model is definitely an issue: instead of having a full record of all transactions a possible trading partner has participated in, a user has to rely on their feedback/reputation instead (section 2.6). It is not only vulnerable to certain attacks on reputational systems (as discussed in section 2.6.3), there is no guarantee that any other users are not just simply lying in their feedback for any reason. An agent may or may not want to take this into account in their evaluations of other agents - and, indeed, into the decision whether or not to leave truthful feedback.

2.3.4 P2PTG: Nash Equilibrium

A two-player game can be said to be in equilibrium if, after both players have chosen a strategy, neither player can improve their utility by changing their strategy[7, p22]. Or, to restate, neither player has a reason to deviate from their already chosen strategy. Assume that Amy and Becka are playing a single-shot instance of the prisoner's dilemma (section 2.4.1):

- Both Amy and Becka have decided to cooperate.
- Amy realises that if Becka cooperates, it is best for her to defect to improve her utility.
- Becka realises that if Amy defects, it is best for her to defect to improve her utility.
- Neither Amy nor Becka can improve their utility by changing from cooperate to defect, so both finally choose to defect.

Thus, in the prisoner's dilemma, the (one and only) Nash equilibrium is (Defect, Defect). There are a number of different mathematical and diagrammatic tools for establishing Nash equilibria (such as in [7, p23]), but the informal written method is sufficient for the purposes of this paper.

The Nash equilibrium (section 2.3.4) for a single round of this game is (Decline, Decline). While playing cooperate, an individual can get a better score by changing to defect. They will reason that the other player will make a similar judgement and thus the payout for both would be -1. Both players will then realise that they can increase their payout to 0 by declining the transaction.

As we will discuss in section 2.5, there are other factors other than simple material utility that come into play in a real-life example of the game: Amy and Becka might have a feeling of loyalty towards one another, or they might be a member of a criminal organisation with rules and enforcement against snitching, or any number of other human factors that influence decision making.

2.4 Similar Games

We can discuss the P2PTG in terms of games that share properties and similarities with it, relating their similar mathematical analyses and strategies to our model. Most of these other games have been developed to represent real-world scenarios like ours has, and so we can take inspiration from the way some of these games work in our model.

2.4.1 Prisoner's Dilemma

Perhaps the most well known and most studied two-player competitive game is the Prisoner's Dilemma. It is definable both as a payoff matrix (figure 2.4) and in informal terms.

Two people suspected of committing a crime have been arrested and are being interrogated by the police. The police officer in question does not have sufficient evidence to arrest them for the crime they have committed, but can arrest them on a lesser charge. The officer places them in separate rooms and offers them a deal - they can either stay quiet (cooperate), or confess and incriminate their partner (defect).

- If both suspects stay quiet, they will both be charged with the lesser offence and given a short prison sentence.
- If one of the suspects talks, that suspect walks free while the other suffers a longer prison sentence.
- However, if both of them talk, neither can reap the benefits from being a witness and both will suffer a long sentence.

Figure 2.4: The payoff matrix for a prisoner's dilemma

A1,A2	A2 Cooperate	A2 Defect
A1 Cooperate	-1,-1	-7,0
A1 Defect	0,-7	-3,-3

The single-shot (two players only play a single instance of the game) Prisoner's Dilemma has been analysed through a number of methods, and it has been established that the only stable, rational (in the game theoretical sense) strategy is for both players to defect, all of the time. Nash equilibrium is one of these methods, as has been explained in section 2.3.4.

2.4.2 Finite Iterated Prisoner's Dilemma

An expansion on the standard one-shot Prisoner's Dilemma is the finite iterated prisoner's dilemma. Two players play N rounds of the prisoner's dilemma, with full knowledge of the previous rounds and the number of rounds to be played. This complicates the previously simple "always defect" answer from the single-shot prisoner's dilemma. While, for a fixed and known n, the game has the same Nash equilibrium (to always defect) - the repeated nature of the game gives an opportunity for counterplay and changing decisions.

The most simple strategies are a naive *always cooperate* or *always defect* strategy, but there are other more complicated strategies, which may or may not produce better results on average. In [8], Alexrod releases the results of the first set of computerised IPD competitions, often referred to as Axelrod's tournament. The paper discusses the tournament results, including the fact that one of the most simple strategies - Tit for Tat (see the list below) - was consistently winning. The paper then uses the results of the tournaments alongside theoretical underpinnings to discuss the role of reciprocity in cooperation. It is a quintessential example of computer simulation fueling game theoretical research, as we will use it in our project.

Some of the strategies used in the IPD, and exhibited in Alexrod's tournament, are:

Random - Chooses an option at random. Sometimes $P(defect) = P(cooperate) = 0.5$, but the probabilities can theoretically be anything.

Tit for Tat - Cooperate in the first round, then repeat the opponent's move every round after that. (I.E. defect against an opponent who defected last round.)

Tit for Two Tat - Cooperate unless the opponent defects twice in a row, then defect

Grim Trigger - Cooperate until opponent defects, then defect forever.

Probing - A cooperative strategy (such as Tit for Tat), except it occasionally randomly defects, to test the opponent's response.

Strategies can also take elements and pieces from other strategies, in order to create even more complicated strategies. The strategies that will be used in the main problem will draw elements from the deep analysis of the IPD, but will have to take into account the number of ways that the problem differs from the IPD.

Decline - Unlike in the standard IPD, agents can choose to decline a transaction with an agent for no cost/payoff if they believe it is in their best interest. There have been IPD experiments including this ability to decline trading partners (known as the IPD with Choice and Refusal[9]), but the standard problem does not involve this choice.

Lots of agents - The standard IPD is played with two agents over a large number of games. In this problem, a large number of agents are playing, and agents will trade with a large number of other agents. Strategies differ in large, evolving populations than from two-player games[10] .

Unreliable history - Instead of having an accurate knowledge of past games played with an agent, each agent instead have to rely on the feedback and reputation scores left by other agents, which may be unreliable.

2.4.3 Chicken

Chicken[11] is a game that originated in a street competition: two drivers race towards each other in a straight line at high speeds, having the option to either swerve or not. While the players crashing into each other is a catastrophically negative outcome for both players, neither player wants to be the one to swerve unilaterally and risk being called a coward and/or suffer some social/reputational loss. This game has been used to model a number of real-world situations, such as brinkmanship in nuclear warfare. The payoff matrix is described in figure 2.5, where $T = tie$, $W = win$, $L = loss$, $D = Disaster$. $W > T > L > D$

A1	A2 Swerve	A2 Do not Swerve
A1 Swerve	T	L
A1 Do not Swerve	W	D

Figure 2.5: Payoff in a game of chicken.

2.4.4 The Hawk-Dove Game

The Hawk-Dove game - originally discussed in [12] and expanded further in [13] - is a simple Chicken-like game played on a population scale, as follows:

Two individuals randomly stumble across a food source.

The cooperative dove strategy involves attempting to share the food with the other individual, and fleeing from any aggressors.

The defecting hawk strategy involves fighting the other individual to try and obtain all of the food.

The generic payoff matrix is defined in figure 2.6, where F = The payoff from getting food, C = The cost from injuries/getting into a fight. Two doves share the food between each other, a dove confronted by a hawk will flee, getting nothing, while the hawk gets all the food; and two hawks will fight: this is sometimes formulated as one hawk beating the other (injuring it) and taking all the food, or - as in this formulation - both individuals taking half of the injury and half of the food.[1]

In a population of biological individuals, individuals who obtain more food are better placed to have a larger share of the offspring in the next generation and, assuming that the trait of being a dove or hawk is inheritable, the population's makeup will shift as to reflect the relative efficacy of the two strategies. In the Hawk-Dove game, the efficacy of the two strategies is linked to the values of F and C: if the possible injuries from getting into a fight outweighs the benefit of getting all the food, then fleeing can be a very good idea. Decreasing the lethality of fights and/or increasing the value of food, will change the overall outcome.

The first experiment will be very similar to the Hawk-Dove game (section 2.8) - its similarities to the underlying problem and its simplicity makes the game a good basis to build from.

Figure 2.6: The payoff matrix for a single round of the Hawk-Dove game.

A1	A2 Dove	A2 Hawk
A1 Dove	$F/2$	0
A1 Hawk	F	$(F-C)/2$

2.4.5 Alarm Calls

Alarm calls[14] are an important part of survival for herd and flock animals; when one individual in the group sees a predator, it can signal to the rest of the group so that they can make their escape. This often poses a risk to the animal that is sounding the alarm, thus the alarm is for the benefit of only the group, and not the individual - resulting in this behaviour being cited as an example of an altruistic behaviour (section 2.5.2).

The prevalence of individuals who will perform alarm calls in the population has a great impact on its effectiveness and on the risk to the individual: if an individual is the only one who will sound the alarm,

[1]This is the expected payoff in general, if $P(win) = P(loss) = 0.5$.

they put themselves at risk in scenarios when the are the first to spot a predator, with no benefit in any other scenarios - on the other hand, if every individual would sound the alarm, the entire flock benefits every time a predator is sighted.

Thus, we have a scenario where each individual would ideally prefer everyone else to be alarm-sounders while not having to be themselves. The selfish, purely self-interested approach would result in no one sounding the alarm at all. Despite this a variety of animal groups who do participate in alarm calling behaviours has been observed. Much discussion has taken place into how these behaviours could evolve in the first place[15] given the loss of utility for the alarm-calling individuals.

2.4.6 Summary

We drew knowledge and design principles from a number of these games for our own model, referencing both their mathematical properties and their relation to the type of real-world problems they are trying to represent. In these games we find strategic dynamics that we can observe throughout the literature and throughout our experimentation: the idea that two agents can do better both cooperating than both defecting but ending up defecting anyway (Prisoner's Dilemma), the idea that iterated and repeated games give greater room for cooperation and strategic diversity (Iterated Prisoner's Dilemma), and the concept of individual versus group utility (alarm calls).

These games are by no means the only games that would be applicable to our scenario, but they represent the most common and most researched. One commonality of these games is that they have not only been studied mathematically, but they have been studied when played by humans (or, in the case of alarm calls, other animals). The social element is a large part of our model and choosing these games as a foundation allows us to relate and combine the mathematical and social scientific analyses.

2.5 Sociological and Psychological Perspective on Decision Making

Humans do not always follow the game theory defined optimal solution all the time and take into account a number of other factors other than their utility gain when making decisions[16]: people instead consider possible social goals, reputational matters (as discussed in section 2.6) and are beholden to emotional/psychological concerns such as fairness, altruism and vindictiveness.

2.5.1 Social Pressures and Goals

A purely rational (in the game theory sense), utility-maximising model of human decision making is incredibly limiting and does not reflect the way humans think. In [17], Schneier writes:

> Prisoner's Dilemmas involve a risk trade-off between group interest and self-interest, but it's generally only a dilemma if you look at it very narrowly. For most people, most of the time, there's no actual dilemma. We don't stand at the checkout line at a store thinking: "Either the merchant is selling me a big screen TV, or this box is secretly filled with rocks. If it's rocks, I'm better off giving him counterfeit money. And if it has a TV, I'm still better off giving him counterfeit money." Generally, we just pay for the TV, put it in our car and drive home. And if we're professional check forgers, we don't think through the dilemma, either. We pay for the TV with a bad check, put it in our car... and drive back to our lair.

From a purely game theoretical perspective, in a single instance of the prisoner's dilemma (as the example from the quote) the dominant strategy is for both players to defect - the real world is a lot more complicated, and so should our models of the situation. There are many reasons why someone would not pay for a TV with fraudulent money: the possibility of facing jail time, the possibility of being ostracised for being a thief, they have probably been told that stealing is wrong and would experience guilt from acts they see to be immoral, and so on.

Schneier defines a taxonomy of four kinds of social pressures:

Moral - Broadly speaking, a person's internal feelings and thought processes as to what is "right" and "wrong" - with corresponding emotions that come with them. Things that a person thinks are "morally right" tend to make them feel good when doing such actions, and vice versa. Guilt can be a powerful motivator. This factor is discussed in greater detail in section 2.5.2.

Reputational - A person or organisation may gain a reputation for being honest or dishonest, which will impact whether or not people will engage with them in transactions in the future. Maintaining a reputation for future business can be worth more than an instant, short-term gain from defecting. This factor is discussed in greater detail in section 2.6.

Institutional - Rules and laws imposed by governments and other organisations with the power to enforce them help ensure cooperation by increasing the cost of defection. The threat of state enforcement of laws ensures stealing results in jail time, and so changes the payoff matrix for the person. In terms of the underlying problem, a platform could penalise defectors by banning their accounts or confiscating their funds.

Security - Systems which aim to prevent damage or harm by defectors, by making it more difficult to defect successfully or reducing the damage a successful defection does. These take a wide variety of different forms, such as: guards, key/keycards, walls, locks, anti-counterfeiting measures in currency, and passwords and two-factor authentication for logging into websites. In terms of the underlying problem, the platform could require proof of goods being sent, or withhold the money from the seller until the buyer acknowledges receipt of the items.

2.5.2 Altruism, Generosity and Morality

Humans are altruistic: they exhibit selfless behaviour, and they do things to benefit other people even when it costs themselves some amount of utility, beyond the levels we see in other animals[18]. In [19], the author distinguishes between "reciprocal altruism" and "altruism" - in the latter case, the individual is unconditionally kind, while in the former case, the individual expects some sort of long-term payoff from their altruism. This payoff can come from expecting future assistance from the individual they are helping (direct reciprocity), or by establishing their reputation as a altruistic individual so third-party individuals may help them in the future (indirect reciprocity)[20].

Cooperation is a much more pronounced characteristic in humans than in other animals, who usually only exhibit cooperative behaviour in kinship groups[18]. It has been argued that our ability to cooperate with non-kin individuals is one of the most important factors in human civilisation as we know it.

2.5.3 Vindictiveness and Altruistic Punishment

As well as positive altruism, humans also exhibit behaviour that is known as altruistic punishment - they will go out of their way to punish deceitful and/or defecting behaviour, even when it costs that individual utility[19]. We can view this as an individual using their resources to enforce a sense of fairness, even if it is not in their direct interest[21].

Similarly, we can also observe that humans can be vengeful - exhibiting irrational (in the game theoretical sense) behaviour in games with individuals who have cheated them in the past[16]. We can consider a conscious, reasoned act to punish a defecting individual out of a sense of fairness and an emotional response to a defecting individual as similar and related phenomena.

2.5.4 Social Science of Decision Making and the P2PTG model

The purpose of our model is to simulate real people using a real online marketplace platform, not just produce mathematical constructs interacting in a pre-defined way. Furthermore, taking into account actual human thought processes and methods will allow us to better represent human behaviour on these platforms. From the Hawk-Dove game and the Schneier quote relating to the Prisoner's Dilemma in real-life scenarios (section 2.5.1), we decided that splitting the population into a majority of genuine agents and a minority of malicious agents best reflects real-life use cases.

We also allowed for our agents to have a range of strategies, even if some of them are irrational from a game theory perspective, humans can act irrationally, be incompetent or naive, and can act in such a way that takes into account concepts such as fairness and morality. Not only that, allowing strategies that are not traditionally rational allows for dynamics such as the one seen in the Alarm Calls scenario (section 2.4.5) to arise.

2.6 Reputation and Trust Systems

2.6.1 What is reputation? What is trust?

Reputation and trust are a pair of concepts referring to the perceived probability that an agent will exhibit cooperative behaviour in a situation. The degree to which an individual believes that another will cooperate can be based on one or some combination of personal experience (trust) and indirect knowledge of that agent's behaviour (reputation)[22].

If there exists some method for reputational feedback to be distributed throughout members of a population, individuals in that system will be able to make transaction decisions based on information about a user's previous transactions with other members of that population. In terms of the underlying problem, online auction platforms tend to offer platforms for feedback on transactions with a score, either positive, negative, or neutral, and possibly written comments: this has been shown to be an effective mechanism for reducing fraud[1].

2.6.2 Factoring reputation into cost-benefit analysis

As well as reducing the information asymmetry between agents about their trading practices and informing decisions, the maintenance of a reputation is also motivation to be honest during transactions[17]. In a somewhat reductive manner, we can see possible reputational gains or losses from trade as a modification of the payoff matrix for the individual's decision. We can see this in figure 2.7: P is defined as the value of a single transaction's worth of positive feedback, while N is defined as the cost of a single transaction's worth of negative feedback. The values of P and N will vary depending on the individual, how much personal/emotional value they place on their reputation, and how important a good reputation is to their strategy.

We also know that the ability to damage a possibly deceitful individual's reputation inclines people to be more likely to trust, so long as they believe that the individual cares about their reputation[23]. Facilitating this trust is vital for society to survive - as discussed in section 2.5.1 - and is vital for the continued business of a platform such as those discussed in the underlying problem. Thus, for the health of the system, it is important to seek to maximise the values of P and N, in order to make positive reputation desirable and negative reputation harmful.

A1	A2 Cooperate	A2 Defect	A2 Decline
A1 Cooperate	2+P	-2+P	0
A1 Defect	4-N	-1-N	0
A1 Decline	0	0	0

Figure 2.7: The payoff matrix for a transaction (including reputation)

2.6.3 Attacks on Computerised Reputational Systems

When they are electronic, centralised and pseudonymous, reputation systems are vulnerable - to varying degrees - to attacks. In [24], Hoffman lays out a taxonomy of these attacks:

Self-promoting - A malicious user falsely increasing their own reputation.

Whitewashing - Escaping the consequences of having a bad reputation by abusing system loopholes to repair it. The user can then continue carry on the behaviour without other users of the system being aware of the user's malicious intent.

Slandering - Deliberately and falsely attacking the reputation of other users.

Orchestrated - Attackers plan and implement strategies involving several of the above methods.

Denial of Service - Preventing the reputation system from functioning in the first place.

The relevance of each of these to the P2PTG:

Self-promoting - A strategy that is definitely implementable in the P2PTG - a user could choose to spend some amount of utility in order to increase their reputation score.

Whitewashing - Similarly, this is an option; in exchange for some amount of their score, a user could be given the option to reset their reputation back to zero.

Slandering - Whilst it is certainly possible for users to give false feedback, though it is questionable how beneficial it would be to that individual. Perhaps if slandering became common in the ecosystem, it would be difficult for genuine users to distinguish between other genuine users with bad feedback due to slander and hawks.

Orchestrating - There is no communication between agents in the same way there could be on a real platform, but that does not mean there cannot be some form of "cooperation" in the strategy the malicious users employ - as suggested in the previous bullet point. Certain combinations of individuals with particular genomes could influence the entire ecosystem in their favour.

Denial of Service - Not particularly applicable in this case; difficult to model and not particularly useful for any particular malicious user to carry out.

Certain systems with particular properties are more vulnerable to certain types of attacks than other - in the P2PTG we could possibly simulate this by varying the different costs for particular actions.

How each attack will be modelled is discussed in further detail alongside potential defences in section 2.8.

2.7 Genetic Algorithms and Evolution

2.7.1 Why use Genetic Algorithms in the P2PTG?

Using a genetic algorithm in the P2PTG allows us to model how strategies will evolve over time. Humans do not blindly follow one strategy all of the time when it is ineffective, and neither should our agents. The strategies employed on a particular platform will improve and alter over time in a constant arms race between attackers and defenders (as discussed in section 2.8). The similarities between the Hawk-Dove (section 2.4.4) game and this model have been previously noted, and this game draws heavily from evolutionary ideas of populations.

The different strategy parameters as they are developed through the experiments will have some method of representing them - whether as an integer, floating point number or binary value. The ways each gene is represented and the way each gene mutates can also be the topic of experimentation and tweaking throughout the process.

2.7.2 What are Genetic Algorithms?

Genetic algorithms are a tool for finding solutions to problems by a process that mimics biological evolution[25]. The general process of a genetic algorithm is:

Generation - A set of individuals are selected or generated in order to form a population. Each individual has a genetic code that represents its properties or strategies.

Simulation/testing - The individuals in the generation are subjected to a test or simulation in order to calculate each individual's fitness.

Termination test - If some criteria, such as the number of generations being above a certain value, highest fitness of an individual meeting a set criteria, or there being no improvement in several generations, has been reached, then the algorithm terminates. Otherwise, it proceeds to the next step and the algorithm continues.

Breeding - In a similar manner to that of real organisms, the individuals breed and their genetic code is mutated and combined in such a way to generate a new generation of individuals. The individuals with higher fitnesses obtain a higher share of the offspring, and thus their genetics are more impactful on the next generation. The algorithm then continues again with this new generation.

It is important to emphasise the importance of mutations in finding the best solutions: unlike a binary search or a simple hill climbing algorithm, which have no way to get away from local maxima[26, p3] (that is, results that are peaks in a certain part of the search space but are not the best result in the entire search space), genetic algorithms include random mutations in new offspring. As in sexual reproduction in biological organisms, the offspring is different from both of the parents in a way that may or may not be better in the evolutionary sense of being fitter: more suitable/adaptable to its environment.

Biological genes can mutate and interact in different ways during the fusion of gametes, and so too can the genes in genetic algorithms: binary (on/off) genes can be flipped in value, genes with a numeric value can have a bit flipped or a random digit added/subtracted to them, various binary functions can be performed on the genes of the two parents, such as transposition, AND, OR and XOR.

2.8 Agent Strategies and Factors in the P2PTG

Throughout this paper we will introduce a number of different factors aimed at exploring different facets of the online marketplace scenario. The different factors are taken from our research into game theory, reputation systems, and the social science of human decision making. The research will be referenced at every stage of the process alongside experimental results and considerations of the online marketplace scenario in order to evaluate each factor's usefulness in the model and the underlying problem.

2.8.1 Agent Types: Hawks and Doves

In [2], the rate of accusations of fraud on eBay is established as 0.2%, with official estimates placing it at 0.01%. The true value is difficult to ascertain due to the issues with establishing the validity of a large number of claims, but it is obvious that the vast majority of transactions are *not* fraudulent. This is obviously a positive thing for the platform, as too high a chance of being defrauded is likely to drive people away. We can reasonably say that the eBay system is healthy due to its low rate of fraud.

In terms of our agents, it means that the majority of the agents at the start of the scenarios will be genuine users with only a few malicious agents; that, despite the fact that the "rational strategy" (in game theoretical terms) may be to defect some or all of the time, they will not. These honest users have invested in ideas of fairness and altruism (see section 2.5), in their reputation (section 2.6), and possibly even in the fear of sanctions from the platform itself. As referenced in section 2.5.1, most people are honest and the exceptions, such as the professional cheque forger in the example, have premeditated their defections. The two types of agents will be described using terms of the Hawk-Dove game (section 2.4.4), a prisoner's dilemma-like game on a population level. The agent types are:

Dove - Genuine user, wishes to cooperate and engage in mutually beneficial transactions - maximising its utility while still remaining honest. The dove will not defect, but will decline transactions if it sees fit.

Hawk - Malicious user, wishes to maximise their own utility at the expense of others. The hawk can defect, cooperate or decline as it believes is best.

Within the dove and hawk categories, there will be differences in the strategies employed by each individual agent - but the spirit of the agent's goal will stay the same.

2.8.2 The Hawk-Dove Game

The first experiment will be almost identical to the Hawk-Dove game. The dove strategy will be to always cooperate, the hawk strategy to always defect. This experiment is simple and will be an opportunity to test the simulation and modify the parameters. It is quite clear from a cursory inspection of the game parameters that hawk will be the dominant strategy, preying on the doves, and will very quickly continue to take over the entire population, which will, in turn, lead to low scores for all the agents - if every agent is a hawk, they all get negative scores.

2.8.3 Basic Reputation Inspection and Trickery

This experiment will start to include the reputation/feedback system into the agents' utility function. Doves will be able to take into account the feedback score when making their decisions either to cooperate if they think their opponent is likely to cooperate or to decline if they think their opponent is likely to defect. hawks will be able to only defect a certain amount of the time in order to maintain their reputations at a certain level to try and fool doves. Both the doves and the hawks will therefore be given one gene each: the doves get a gene describing how much weight they place on their opponent's reputation score, and the hawks what percentage positive feedback they will try and maintain.

2.8.4 False Feedback and Direct Experience

In this experiment, feedback will no longer be automatically assigned on the basis of what that user did (positive for cooperate, negative for defect) - it will be up to the individual agent what feedback to leave. Doves, of course, will be honest all the time - but the hawks need not be so scrupulous. To compensate, all agents will be given the ability to take into account personal experience with a user (*trust*, rather than reputation - see section 2.6) rather than just that user's feedback score. If a user has a 90% positive rating but they have defrauded you twice, it is unlikely you would trade with them. Genes will be added to the genome for hawks to deal with the circumstances in which they may leave false feedback and one or two more genes added for all agents that deal with their weighting of direct experience.

2.8.5 Altruistic Punishment

Dove agents will be able to, for a cost, punish an agent with who they have just transacted with more than one point of negative feedback. This is a common occurrence in human behaviour - as discussed in section 2.5.3 - and will also lead to a situation similar to alarm calls in animals; something which has been studied in great detail and discussed in detail in section 2.4.5. This is a very similar scenario: all of the doves in the population will benefit from one dove deciding to punish using this system, yet it will cost the individual dove utility and - as noted in the initial definitions of the doves - they are still out to maximise their own utility, even if not by deception.

While this is not a common feature in online trading platforms, it is an interesting experiment for its use of observed human nature to enhance a sociotechnical system's health. It is very possible that an altruistic punishment feature could work on one of these platforms if it were tried, and the P2PTG experimentation could very well lay the groundwork for this.

A gene will be added to the dove genome to represent how inclined that agent is to altruistic punishment.

2.8.6 Whitewashing and Reputation Length

Hawks will, for a utility cost, be able to whitewash (section 2.6.3) their accounts - clear their reputations entirely - wiping away all the sins of the past and allowing them to trade again with a fresh "account". To compensate, doves will be able to take into account the *length* of a user's reputation rather than just their overall positive/negative percentage (e.g. an agent will probably value a 450 positive/50 negative score over a 9 positive/1 negative score). Genes will be added that decide when a hawk will use the whitewashing feature, and also to modify the reputation weighting algorithms for doves.

2.8.7 Summary

Each one of these factors will be implemented into the P2PTG system, which will then be experimented upon to determine how they perform and how each factor interacts with others. These interactions between factors is possibly the most important part of the P2PTG system, as in the real online marketplace system, all of the information is presented at once and not separately.

Most of these factors are fully or partially representing what could be considered as "intuition" on the part of a human user: it is unlikely that they will calculate a probability of a successful transaction with a user given a set of parameters, but instead get a feeling as to how the user will act based on what they value and their pre-conceived notions about what a reliable user is like. Experimentation using this system could allow us to establish how a user may wish to weight different parts of the user's history - to take one thing more into account than another - but providing a mathematical formula is unlikely to help any users at all.

Chapter 3

Experiments

3.1 Introduction to Experiments

We will conduct a series of experiments with the intent of investigating the relative effectiveness of a variety of strategies. We will carry out four experiments, each adding extra factors to the model and each containing a number of sub-hypotheses to be tested. The four factors are discussed in section 2.8 and are, in order: the basic Hawk-Dove game, reputation management and consideration, false feedback and personal experience, and altruistic punishment. Note that whitewashing and reputation length is not included here, due to time constraints - it is discussed as future work in section 4.2.2.

For each of these four experiments, we will design hypotheses; methods to test these hypotheses; state the results for each sub-experiment; analyse the results for each hypotheses; then finally conclude and link all of the experiments together. The experiments are designed to confirm and further build on the mathematical analyses of the model, which we will discuss alongside the experimental results.

3.1.1 P2PTG Basic Form

Each generation of the *Peer-to-Peer Trading Game* (henceforth referred to as *P2PT Game or P2PTG*) will happen as follows:

- N agents are selected or generated using one of several possible methods.
- The number of transactions to run - T - is decided.
- T transactions run, each one being carried out as follows:
  - Two random agents from the list are selected.
  - Each choose whether to Cooperate, Defect or Decline. Dove agents are only allowed to cooperate or decline, while hawk agents can do any of the three actions as they see fit. The payoff for these actions is defined in figure 3.1: it is a symmetric game (the payoff is the same for both players).
  - Each user then has the opportunity to leave feedback about the user they traded with, assuming the transaction went ahead; that is, if neither declined it. In experiments 2 and 3, hawk agents will have the opportunity to leave false feedback at this step. In experiment 3, dove agents will be able to utilise the altruistic punishment option at this step.
- After all the transactions, each agent's score is computed.

- If there is another generation to run, a new generation is bred from the old generation, as discussed in section 3.1.2 and section 3.1.3.

A1	A2 Cooperate	A2 Defect	A2 Decline
A1 Cooperate	2	-2	0
A1 Defect	4	-1	0
A1 Decline	0	0	0

Figure 3.1: The payoff matrix for a transaction

The game is discussed and analysed in detail in section 2.2.

3.1.2 Basic Parameters

There are certain basic properties of the system that must decide before running any simulation - rather than hard-coding these parameters, making them alterable allows us to change them to better fit the scenario at hand. The following list contains these parameters and what they mean:

N_AGENTS - The number of agents that will be used in each round.

TRANSACTIONS_PER_AGENT - This parameter is multiplied by the number of agents in order to give the total number of transactions to be performed in a simulation. [1]

N_GENERATIONS - The number of generations the simulation will run for.

TOP_X - A method of breeding the new generation of agents. Simply put; the top x (where x is the number set in this parameter) agents are taken and put into the breeding pool. For each agent that has to be bred, two agents are chosen randomly from this pool. The default value of TOP_X is 30%.

The first three parameters give us control over the scale of the experimentation as the computational power required to run a simulation scales linearly with the values of N_AGENTS, TRANSACTIONS_PER_AGENT and N_GENERATIONS.

The breeding method is fully customisable; whilst in future extensions it could be altered (section 4.2.2), it remains constant throughout these experiments. TOP_X is set to 30% after preliminary testing of the system revealed that is provided reasonable results: favouring the better scoring strategies whilst allowing for strategic diversity (section 2.7.2).

3.1.3 Gene Types

There are two types of genes that are used during these experiments: the binary gene and the integer value gene. The binary gene is a simple gene that can hold one of two values. An integer value gene holds an integer value I, where $1 \leq I \leq 1023$. These values of I are representable as a 10-bit integer, and this representation is used during the mutation stages of breeding - flipping bits is one of the several possibly mutations a gene can undergo. Often during this paper, these integer value genes will be expressed as a percentage for presentational reasons, e.g. $100 \approx 10\%$.

Table 3.1 is a table containing details about all the genes used in these experiments.

[1] The number of transactions actually undertaken by each agent is $2 \times TRANSACTIONS\_PER\_AGENT$, because each transaction necessarily requires two agents to be participating.

[2] E is removed in experiment 3. See section 3.5.1 for details.

Table 3.1: Breakdown of genes

Gene Name	Symbol	Experiment(s)	Notes
strategy	S	0, 1, 2, 3	"H" for Hawk, "D" for Dove.
prob_defect	P	1, 2, 3	Integer gene.
rep_weighting	R	1, 2, 3	Integer gene.
false_feedback	F	2, 3	Integer gene.
direct_experience	E	2 [2]	Integer gene
altruistic_punishment	E	3	Integer gene.

3.1.4 Statistics and Variables

Throughout these experiments we will use a variety of statistical techniques in order to establish the veracity of our hypotheses. A number of our experiments are multiple repeats while observing the mean, standard deviation and/or interquartile range of a particular value: This is often the average score, other times this value will be related to the strategic breakdown of the population.

We will use single-factor ANOVA to establish difference between a number of populations, and single-tailed t-tests to establish difference between pairs of populations. The standard p value for these statistical tests will be $p = 0.05$. In order to compensate for multiple t-test values causing false positives, we will use Bonferroni's method - dividing the p value by the number of tests to be performed.

In addition to this we will use linear regression to establish correlations when the independent variable we are testing is continuous as opposed to discrete. The linear regression will provide us with a line (in the form $y = mx + c$) and a Product Moment Correlation Coefficient (PMCC). We will consider a correlation to be weak if the absolute value of the PMCC is less than 0.5, moderate if it is between 0.5 and 0.75, and strong if it above 0.75.

A number of the independent variables in our experiments will be probability distributions: integer value genes for an agent consist of a single number, but the population consists of a number of individuals - the value for each agent's gene will be drawn from that probability distribution. There are four probability distributions that will be used throughout these experiments, referred to as I_c, I_1, I_2 and I_3 respectively, and they are listed in table 3.2.

Table 3.2: Probability distributions to be used throughout the experiments.

Identifier	Value	Comment
I_c	A uniform distribution, $1 \leq I \leq 1023$	The control
I_1	A normal distribution, mean = 300, SD = 150	Significantly lower than 50%
I_2	A normal distribution, mean = 500, SD = 150	Around 50%
I_3	A normal distribution, mean = 700, SD = 150	Significantly higher than 50%

3.2 Experiment 0: The Hawk-Dove Game

3.2.1 Introduction

In this very basic experiment, the game will resemble the Hawk-Dove game. The strategies for the two agents are as follows:

Dove - Always play "cooperate".

Hawk - Always play "defect".

These two strategies make up the single gene that will be considered in this experiment - the "strategy" gene, which is a binary gene (as defined in section 3.1.3) with values "hawk" (H) and "dove" (D).

3.2.2 Parameters

As well as the default parameters, there are a number of parameters that are specific to this experiment:

INITIAL_HAWK - The initial proportion of "hawk" agents in the population.

GENETICS_STRATEGY_HAWK_DOM - When an agent with the "hawk" gene and an agent with the "dove" gene breed, this parameter determines with what probability it is a hawk, and the probability of it being a dove is the inverse.

GENETICS_STRATEGY_MUTATE - After a child agent's strategy gene has been selected, this parameter determines with what probability it is "flipped".

The first parameter refers to the expected number of hawks in the initial population - each randomly generated agent in the first generation has an INITIAL_HAWK% chance of being a hawk. The expected number of hawks in the first generation is equal to $INITIAL\_HAWK \times N\_AGENTS$. This parameter being changeable allows us to experiment with the impact of including more or less malicious users at the very start of the experiment.

The next two parameters deal with how two agents breed. The breeding method works as follows:

- Whether the child is a hawk or a dove is decided as follows:
  - If the parents are both doves, the child is a dove.
  - If the parents are both hawks, the child is a hawk.
  - If one parent is a dove, and the other a hawk, the child is a hawk GENETICS_STRATEGY_HAWK_DOM% of the time, or otherwise a dove.
- After the agent's strategy has been decided, it has a GENETICS_STRATEGY_MUTATE% chance of it being "flipped" - changed to the other strategy.

This method allows the new generation to roughly reflect the make-up of the best agents of the previous generation, while ensuring the gene pool does not stagnate by injecting a small amount of randomness at two separate points.

3.2.3 Hypotheses

This experiment is serving as almost a "test run" for the main set of experiments, and this is reflected in the hypotheses that will be tested. They are as follows:

H0.1 - In one generation with any number of hawks and doves, all hawks will score higher than all doves.

H0.2 - In any population where there are enough hawks initially, or there is a high enough probability of hawks entering the population through mutation, hawks will eventually dominate the population.

H0.3 - Changing N_AGENTS or TRANSACTIONS_PER_AGENT will not significantly impact the result of the experiment, so long as both parameters remain above a certain threshold.

H0.1 is looking at the fact that, in this limited scenario, "cooperate" is (in the game theory sense) strictly dominated by "defect" - thus, there should be no incidents of a dove agent beating a hawk agent in score.

H0.2 is directly related to the underlying auction problem: in a situation where legitimate users have no recourse, protection or action beyond being prey for malicious ones, we will be left with a situation with no legitimate users. In real terms, this probably means that everyone but the fraudsters would leave.

H0.3 is about establishing parameters for later experiments. While N_AGENTS, TRANSACTIONS_PER_AGENT and N_GENERATIONS are all completely tweakable parameters, it is important to note that any increase in any of these parameters will come with a corresponding increase in computational time required to run the simulation. It is important to establish the extent that smaller simulations impact the results in order to make informed decisions about computational time in future simulations.

3.2.4 Method

Experiment 0.1

H0.1 is very simple, and thus this experiment will follow a very simple method:

- Run single generation simulations for values of INITIAL_HAWK between 0% and 100%, incrementing by 1% each time.
- After the simulation has finished, check whether any dove scored above any hawk.

Experiment 0.2

Building on experiment 0.1, we are trying to establish that - because of the dominance of the hawk strategy over the dove strategy - all or most populations will eventually trend to be hawk-dominated. To do this, we will run a simulation using a wide variety of different variables to establish how these populations end up.

The relevant parameters here are TOP_X (which determines how many agents are taken into the breeding pool, and thus the proportion of hawks that will make up the breeding pool), INITIAL_HAWK (determines the number of hawks initially in the population), GENETICS_STRATEGY_HAWK_DOM (determines the chance of a hawk-dove couple resulting in a hawk) and GENETICS_STRATEGY_MUTATE (which determines the probability of the strategy gene being flipped during breeding.)

We will use five combinations of values of GENETICS_STRATEGY_HAWK_DOM (D) and GENETICS_STRATEGY_MUTATE (M) in this experiment, as found in table 3.3. These variables have been chosen to be a sample of the whole range of possible values whilst requiring a manageable number of experiments.

Table 3.3: The values of D and M used in experiment 0.2.

D	M
0.1	0.05
0.3	0.05
0.7	0.05
0.1	0.4
0.7	0.4

The method will be as follows:

- Run 10 five-generation simulations for each pair of values as described in table 3.3.

- For each pair of values, observe the mean percentage of hawks in the population between all the tests.

The other parameters used in this experiment are laid out in table 3.4.

We should observe in all cases that the population is mostly hawk (trending to $1 - M$).

Table 3.4: Experiment 0.2 parameters

Parameter	Value
N_AGENTS	250
TRANSACTIONS_PER_AGENT	250
TOP_X	30%
INITIAL_HAWK	5%
Repeats	10

Experiment 0.3

In experiment 0.3, we are trying to determine the extent that we can reduce the computational requirements of these experiments while minimising the extent that the small N causes deviations from the expected values. We will be using five values of N_AGENTS (10, 25, 50, 100 and 250) and five values of TRANSACTIONS_PER_AGENT (10, 25, 50, 100 and 250).

The method will be as follows:

- For each pair of values, run 20 two-generation simulations.
- At the end, observe the interquartile range and standard deviation of:
  - The average score of dove agents (divided by TRANSACTIONS_PER_AGENT).
  - The average score of the hawk agents (divided by TRANSACTIONS_PER_AGENT).
  - The percentage of hawks in the population.
- Plot the graph of $log(N \times T)$ against each of these observations, and look for a negative linear correlation.

Standard Deviation (SD) and interquartile range (IQR) are both measures of spread in a set of data - we wish to minimise the difference that randomness makes on experiments, and a higher spread indicates a higher random impact.

For larger values of $N \times T$, we should see that we get a corresponding smaller value of the variance - hence the expectation of a negative correlation. We will take the logarithm of $N \times T$ for presentational reasons (the smallest value of $N \times T$ is 100, the largest is 62500 - which is a large range to display linearly).

The other parameters used in this experiment are laid out in table 3.5.

Table 3.5: Experiment 0.3 parameters

Parameter	Value
TOP_X	30%
INITIAL_HAWK	10%
Repeats	20

3.2.5 Results

Experiment 0.1

In no test did any dove score better than any hawk.

Experiment 0.2

Table 3.6 is an abridged table of the values of D and M (the pair of which are referred to as I here) against the proportion of hawks in the population after the fifth generation (H). The full table is available in the appendix (section A.1.1).

Table 3.6: Experiment 0.2: abridged table of I against H

-	H_{mean}
$D = 0.1, M = 0.05$	0.956
$D = 0.3, M = 0.05$	0.944
$D = 0.7, M = 0.05$	0.953
$D = 0.1, M = 0.4$	0.612
$D = 0.7, M = 0.4$	0.595

Experiment 0.3

This experiment contains a substantial amount of data, the majority of which is in the appendix (section A.1.2).

This results section contains results for one out of six of the dependant variables in greater detail - the standard deviation of the percentage of hawks in the population. The table containing this data is at table 3.7 and the corresponding graph at figure 3.2. The value of the PMCC in the case provided is -0.6724, which is a moderate correlation as defined by our method. These results are typical of the results for all six dependent variables.

Table 3.7: Experiment 0.3 results: standard deviation of the percentage of hawks in the population.

T N	10	25	50	100	250
10	32.592	20.881	14.904	9.559	6.421
25	34.293	17.302	13.298	12.123	7.102
50	33.512	21.106	13.697	10.26	5.741
100	30.43	20.15	16.363	10.865	7.017
250	31.509	19.518	15.884	11.696	6.847

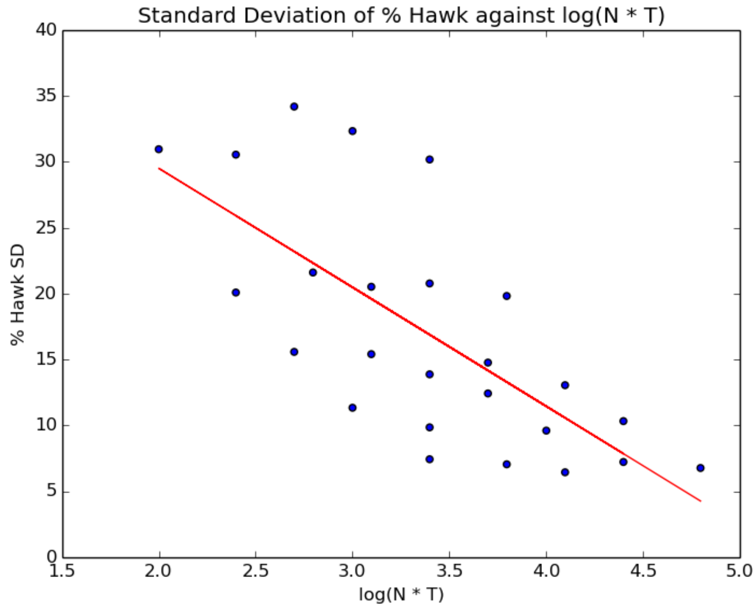
Observe in table 3.7 that there is a moderate correlation between (the logarithm of) N_AGENTS and TRANSACTIONS_PER_AGENT, and the standard deviation of the percentage of hawks in the population.

3.2.6 Analysis

Experiment 0.1

This is a fairly trivial issue, in the game as described in section 3.1.1: we can clearly see that the "defect" strategy (weakly) dominates the "cooperate" strategy: which is to say, playing "defect" will always obtain

Figure 3.2: Experiment 0.3 results: standard deviation of the percentage of hawks in the population.



a greater than or equal payoff to playing "cooperate".

Experiment 0.2

If we consider a breeding pool where H is the proportion of hawks, D is the value of GENETICS_STRATEGY_HAWK_DOM and M is the value of GENETICS_STRATEGY_MUTATE, we can establish that:

If two hawks are breeding (with probability $= H^2$): $P(hawk) = 1 - M$

If two doves are breeding (with probability $= (1-H)^2$): $P(hawk) = M$

If one hawk and one dove are breeding (with probability $= 2H(1-H)$): $P(hawk) = D(1-M)+(1-D)M = D + M - 2DM$

(These probability values are based on the assumption that the number in the pool is large enough that the fact that two *different* agents must be chosen makes little impact on the probability)

For various values of D and M, we can produce a table (table 3.8) of the expected proportion of the new generation who will be hawks (P).

If we further assume that the TOP_X parameter is a reasonable value less than the entire population, and that H0.1 holds (that is, hawks always come at the top of the scoreboard) - we can be reasonably sure that most populations will trend as to be predominately hawk (they will trend towards $1-M$ for small M[3]) after some number of generations.

We can clearly see from the experimental results that, for these set of values, the population easily trends towards being proportioned $1 - M$ hawk. These set of values are in no way the full set of potential values, but they represent a wide spread of the possibilities.

[3]For larger values of M, the population will fluctuate wildly between generations. $M = 1$ roughly speaking means that the generation will flip between being hawk-dominated and dove-dominated on a two-generation cycle.

Table 3.8: Theoretical outcomes for the values of D and M used in experiment 0.2.

D = 0.1, M = 0.05		D = 0.3, M = 0.05		D = 0.7, M = 0.05		D = 0.1, M = 0.4		D = 0.7, M = 0.4	
H	P	H	P	H	P	H	P	H	P
0.0	0.05	0.0	0.05	0.0	0.05	0.0	0.4	0.0	0.4
0.1	0.075	0.1	0.108	0.1	0.172	0.1	0.406	0.1	0.427
0.2	0.115	0.2	0.172	0.2	0.288	0.2	0.414	0.2	0.453
0.3	0.169	0.3	0.244	0.3	0.396	0.3	0.426	0.3	0.477
...	...	...	...	...	...	...	...	...	...
0.9	0.795	0.9	0.828	0.9	0.892	0.9	0.566	0.9	0.587
1.0	0.95	1.0	0.95	1.0	0.95	1.0	0.6	1.0	0.6

The fact that the population will always trend to be mostly hawk is a simple extension of H0.1 - doves, in this scenario, are helpless prey and haws will always score better than them in any population. Hawks will then become a large proportion of the breeding pool, and thus a large proportion of the next generation. In the next generation, the problem exacerbates further, and hawks quickly overwhelm the doves.

Experiment 0.3

We can see that, for all parameters and both measures of spread, an increase in the values of N_AGENTS (N) and N_TRANSACTIONS_PER_AGENT (T) implies a reduction in the spread of values. Increasing the values of N_AGENTS and N_TRANSACTIONS_PER_AGENT will give us more reliable (less impacted by randomness in individual simulations) results - the correlation observed is “moderate”, as defined by our method.

With a larger data set beyond that which is computationally feasible for these experiments, we might well find that this correlation is non-linear: the impact of increasing N and T on reducing the standard deviation and IQR of the results decreases over time. We can observe, as N and T increases, the measure of spread decreases. However, we can also note that the spread is unlikely ever to be 0, despite trending toward it.

Together, increasing the values of N and T, as well as running simulations a number of times to obtain, will give us data that is more indicative of the properties of the system.

3.2.7 Conclusion

Through these three experiments we have learned that some of our key assumptions as to how the system works are true. We have also demonstrated a number of methods by which we can tie mathematical analysis to system properties. We have discovered that, without a defence, honest agents will be quickly overrun by malicious users preying on them, regardless of the situation. Furthermore, we laid the groundwork for future experimentation by establishing the impact of reducing/increasing the computational requirements of an experiment on the overall results.

From this foundation, we can begin to add layers of complexity to the strategies employed by the agents.

3.3 Experiment 1: Basic Reputation and Trickery

3.3.1 Introduction

This stage of the experiment involves simple reputation management/inspection and adds two genes to the genome, prob_defect (P) and rep_weighting (R). Both of these genes are integer value genes (section 3.1.3). The strategies played by the agents are as follows:

Hawk - The hawk strategy is to defect $P\%$ of the time and cooperate the rest of the time.

Dove - The dove strategy is to cooperate with any agent with positive feedback above $R\%$ and decline otherwise.

3.3.2 Parameters

There are two new parameters for this experiment - the probability functions for P and R. P and R are, in the first generation, randomly assigned on a per-agent basis and in future generations, derived from an agent's two parent agents.

The default value for the P and R parameters is a uniform distribution, in which all values $1 \leq P, R \leq 1023$ are equally likely to be chosen. It is possible to use other probability distributions (such as a normal distribution) to attempt to express and test a particular type of behaviour.

There is also the matter of mutation - unlike the strategy gene (dove or hawk), mutation is not as simple as a certain probability of the agent having a different strategy. The mutation method is customisable, but for the purpose of these experiments, integer value gene mutation will follow the same method:

- Take the average of the two parent genes' values.
- Add a random number to the average - where the random number is drawn from a uniform distribution $-64 \leq M \leq 64$.
- The number is represented as a 10-bit integer. For each bit in turn, flip it with probability $\frac{1}{V}$ where V is the value of the bit. Which is to say, the bit with value 128 is flipped $\frac{1}{128}$ of the time. The exception to this is the 1 bit, which is ignored rather than flipped with probability 1 (as $P_{flip1bit} = \frac{1}{1}$).

This method includes a number of desirable properties: the new genes value is tied to the parent genes' value, while still differing by a small amount on average, and on top of that there is a small chance of very large mutations.

Genetic algorithms are search algorithms, and as such the best agents' properties should be taken into account when breeding a new generation. Without the small randomness injection, the gene pool would quickly stagnate and halt the search problem. Finally, the small possibility of a large change is designed to allow the system to not get stuck at local maxima.

3.3.3 Hypotheses

This experiment deals with a very simple way of looking at reputation: doves simply looking at the reputation value and, if it is higher than a certain value, trading with them - while hawks defecting a certain percentage of the time to maintain a certain percentage of positive reputation. These hypotheses deal with the interaction between the two genes:

H1.1 - In one generation, doves with R values of around 50 percent will get better scores than doves with R parameters significantly higher or lower than 50 percent.

H1.2 - In one generation in which doves have R values around 50 percent, hawks will get better average scores than in one generation with doves with a higher value of R.

H1.3 - Starting with a population of doves with R values around 50 percent, hawks will be a more prominent part of the population for a longer time than in the case of a starting population with doves with a higher value of R.

All three hypotheses are designed to test the idea that, while cooperating with certain hawks may be best on an individual level, on a population level it simply helps hawks score better than doves and thus get a greater share of the gene pool - which contributes to a less healthy system overall.

3.3.4 Method

All three experiments involve using rep_weighting (R) as an independent variable. It is important to note that the R variable is not a single parameter, it is a value assigned individually to each agent in the system. We will use the four values of I as described in table 3.2.

Experiment 1.1

This experiment is simple to run:

- Run 5 single generation experiments. We will use a uniform random distribution, $1 \leq R \leq 1023$, for R.
- Observe the correlation between R and dove agent's average score per transaction (S).

Note that we refer to the dove agent's average score per transaction (S) - this is for two reasons: firstly, it allows us to more easily relate the results to the mathematical analysis, and also to even out any discrepancies between different agents being involved in different numbers of transactions.[4]

Doves with R values above 50% will lose out on potentially profitable transactions, while doves with R values below 50% will be undertake transactions that are more likely than not to be fraudulent (further expanded upon in section 3.3.6). As a result of this, we should observe two distinct linear correlations: those for values of R below 50% (C_1), and those for values above 50% (C_2).

C_1 is a line $S = m_1 \times R + c_1$ for $0 < R < 512$, and C_2 is a line $S = m_2 \times R + c_2$ for $512 < R < 1024$ such that:

- m_1 is positive (score increases as R does while $R < 512$.
- m_2 is negative (score decreases as R does while $R > 512$.
- $|m_2| \approx |m_1|$ (The score of an agent with R $= 512 - N$ will be approximately the same as an agent with R $= 512 + N$). This is discussed further in section 3.3.6.
- C_1 and C_2 should meet at R = 512, at some $S < 2$.

[4]Agents are not guaranteed to participate in TRANSACTIONS_PER_AGENT transactions - each agent has equal chance of being chosen to participate in each transaction. Thus, it is possible that some agents participate in less transactions than others through chance.

Figure 3.3: H1.1 expected correlations

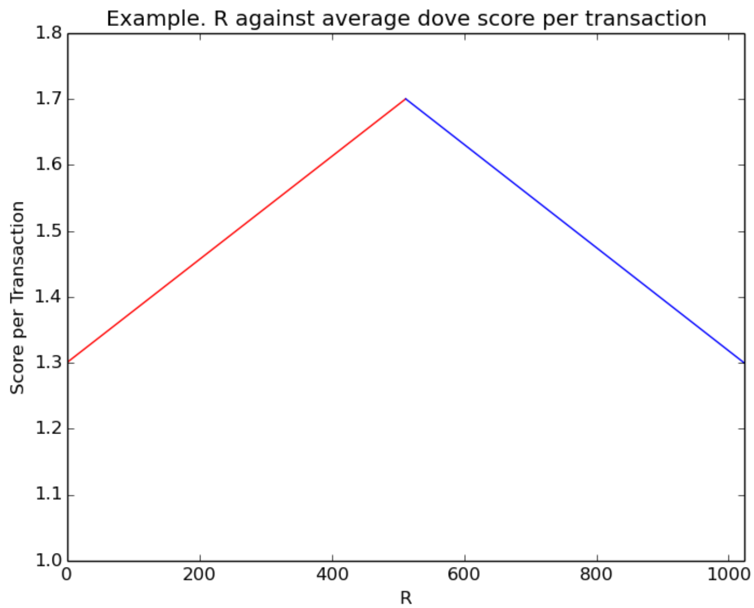


Figure 3.3 is an example of what we expect the results of this experiment to be, where C_1 is represented in blue and C_2 in red. C_1 and C_2 will be derived using linear regression. For both regressions, we can discuss the strength of the correlation relative to its Product-Moment Correlation Coefficient (PMCC) - for this purpose, we will consider:

- $PMCC \leq 0.5$ as a weak correlation,
- $0.5 < PMCC \leq 0.75$ as a moderate correlation,
- and $0.75 < PMCC \leq 1$ as a strong correlation.

The parameters used in this experiment are laid out in table 3.9.

Table 3.9: Experiment 1.1 parameters

Parameter	Value
N_AGENTS	1000[5]
TRANSACTIONS_PER_AGENT	1000
N_GENERATIONS	1
INITIAL_HAWK	30%
prob_defect (P)	Uniform Distribution, $1 \leq P \leq 1023$.
Repeats	5

[5]Both N_AGENTS and TRANSACTIONS_PER_AGENT are higher than in a lot of other experiments - because this experiment is only one generation, it can be run in a smaller amount of time, thus allowing for larger sample sizes.

Experiment 1.2

In this experiment, we are investigating the idea that, in a population where doves have a value of R around 50%, hawks have better average scores. The method is as follows:

- Do 25, one-generation experiments for each of I_c, I_1, I_2, I_3.
- Observe the average score per transaction for hawks (S_H).

We should observe that:

- $S_H(I_2) > S_H(I_3)$ - Hawks do better when doves have R values around 50% rather than when they have R values around 70%.
- $S_H(I_1) > S_H(I_2)$ and $S_H(I_1) > S_H(I_3)$ - Hawks do better when doves have R values around 30% than they do when they have R values around 50% or 70%.

First we will perform single-factor ANOVA ($p < 0.05$) to ascertain whether there are significant differences between the results.

Then, for each of the three inequalities above, a t-test will be performed. We will use a standard test of $p < 0.05$, which will be adjusted to $p < 0.166$ by Bonferroni's method to compensate for the multiple t-tests.

The parameters used in this experiment are laid out in table 3.10.

Table 3.10: Experiment 1.2 parameters

Parameter	Value
N_AGENTS	250
TRANSACTIONS_PER_AGENT	250
N_GENERATIONS	1
INITIAL_HAWK	10%
prob_defect (P)	Uniform distribution, $1 \leq P \leq 1023$.
Repeats	25

Experiment 1.3

This experiment involves a larger experiment in terms of computational requirements, and is about measuring longer-term trends over many generations rather than behaviours of a single generation.

Do 50 50-generation experiments for each of I_c, I_1, I_2, I_3.

Observe for each experiment, the number of generations in which the percentage of hawks is greater than 20% (H).

We should observe that:

- $H(I_3) < H(I_C)$, $H(I_3) < H(I_1)$, $H(I_3) < H(I_2)$ - There are less generations with more than 20% hawks in the I_3 test than any of the other tests.
- $H(I_2) < H(I_1)$ - There are less generations with more than 20% hawks in the I_2 test than in the I_1 test.

First, we will perform single-factor ANOVA ($p < 0.05$) to ascertain whether there are significant differences between the results.

Then, for each of the four inequalities above, a t-test will be performed. We will use a standard test of $p < 0.05$, which will be adjusted to $p < 0.0125$ by Bonferroni's method to compensate for the multiple t-tests.

The parameters in this experiment are laid out in table 3.11

Table 3.11: Experiment 1.3 parameters

Parameter	Value
N_AGENTS	100
TRANSACTIONS_PER_AGENT	100
N_GENERATIONS	50
TOP_X	30%
INITIAL_HAWK	10%
prob_defect (P)	Uniform distribution, $1 \leq P \leq 1023$.
Repeats	50

Additional note: this is the second version of this experimental method. The first version did 10 repeats with parameter values as described in table 3.12

Table 3.12: Experiment 1.3 parameters - abandoned version

Parameter	Value
N_AGENTS	250
TRANSACTIONS_PER_AGENT	250
N_GENERATIONS	50
Repeats	10

Upon analysing the data, it became clear that the number of repeats was a serious detriment to the statistical usefulness of the results. We decided that we would run the experiment a second time, reducing the N_AGENTS and TRANSACTIONS_PER_AGENT parameters by a factor of 2.5 and increasing the number of repeats to 50.

This change gives more statistically useful results while still maintaining a similar computational requirement (about 8 hours on all four cores of an AMD A8 processor). The results of the abandoned first experiment are included in the results section for completeness.

3.3.5 Results

Experiment 1.1

As we are using linear regression, the lines C_1 and C_2 will take the form $y = mx + c$. Table 3.13 provides the values of m and c and the value of the Product Moment Correlation Coefficient (PMCC) for each line.

We can clearly see that the graph (figure 3.4) looks like what we were expecting, as described in the method. Here, the graph has been coded in the same colours as the example graph before.

We can also note that, for both C_1 and C_2, there is a sufficient value of the PMCC to be considered "strongly correlated" as defined by the method.

Figure 3.4: Experiment 1.1 results - graph of R against score.

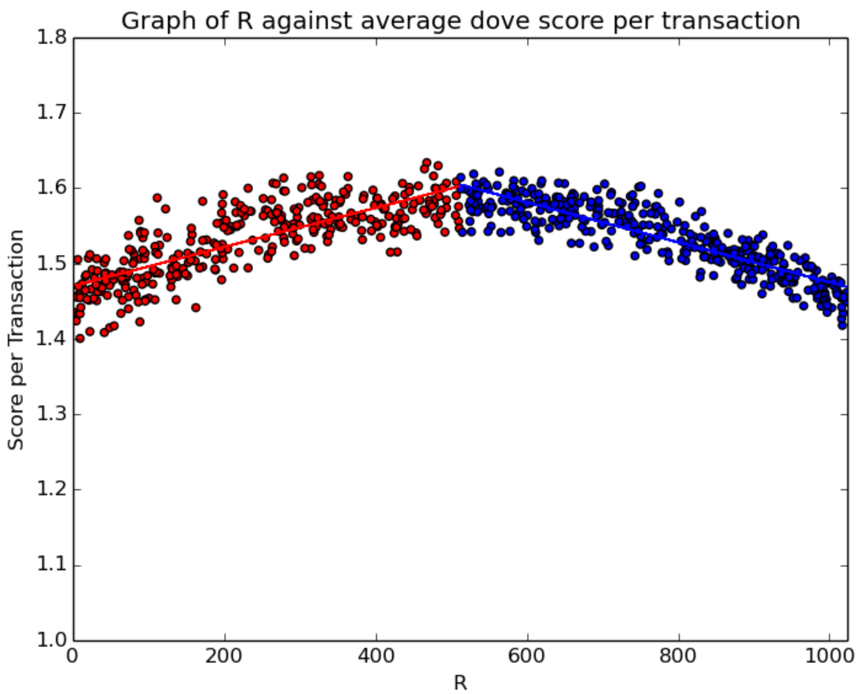


Table 3.13: Experiment 1.1 Results - linear regressions for the two lines C_1 and C_2.

-	C_1	C_2
m	0.0003300	-0.0003425
c	1.385	1.723
PMCC	0.8471	-0.9040

Experiment 1.2

The table in table 3.14 shows the mean (and SD) hawk score per transaction for each of the different values of R.

Performing the one-factor ANOVA test on these four values gives us $p = 3.15 \times 10^{-89}$, which is strongly significant (at our $p < 0.05$).

T-tests (one-tailed):

- $S_H(I_2) > S_H(I_3)$ - $p = 1.29 \times 10^{-27}$ - $p < 0.0166$
- $S_H(I_1) > S_H(I_2)$ - $p = 5.82 \times 10^{-35}$ - $p < 0.0166$
- $S_H(I_1) > S_H(I_3)$ - $p = 2.00 \times 10^{-58}$ - $p < 0.0166$

Table 3.14: Experiment 1.2 results - the average hawk score given different values of R.

-	Mean S_H	SD S_H
I_c	1.235	0.089
I_1	1.740	0.120
I_2	1.232	0.142
I_3	0.812	0.133

Experiment 1.3

The table at table 3.15 shows the mean (and SD) number of generations that hawks make up 20% or more of the population against the different values of R. There are graphs each corresponding to a single repeat of this experiment at figure 3.5.

Table 3.15: Experiment 1.3 results - generations with significant hawk populations given different values of R.

-	Average H	SD
I_c	11.34	5.94
I_1	13.18	1.39
I_2	14.42	2.20
I_3	6.5	7.973

Performing the one-factor ANOVA test on these four values gives us $p = 3.65 \times 10^{-12}$, which is strongly significant (at our $p < 0.05$).

T-tests (one-tailed):

- $H(I_3) < H(I_c)$ - $p = 0.00049$ - $p < 0.0125$.
- $H(I_3) < H(I_1)$ - $p = 2.914 \times 10^{-7}$ - $p < 0.0125$.
- $H(I_3) < H(I_2)$ - $p = 5.02 \times 10^{-9}$ - $p < 0.0125$.
- $H(I_2) < H(I_1)$ - $p = 0.00896$ - $p < 0.0125$.

As discussed in the method section, there was an abandoned first version of this experiment - the results of which are in section A.2.1

3.3.6 Analysis

The analysis for each of these experiments will take the form of an a priori mathematical exploration of the hypothesis, followed by a discussion of how this ties in with the observed results.

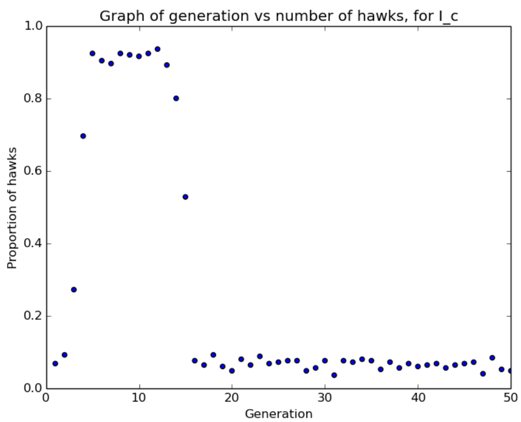
Experiment 1.1

Assume a dove agent with a rep_weighting value of R (it will decline any transaction with an agent with a positive feedback less than R percent) is trading with a hawk agent with a value of prob_defect P (that has a positive feedback value $1 - P$).

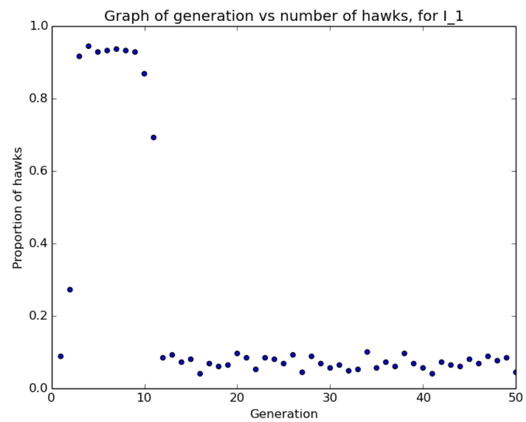
If the dove trades with the hawk, its expected payoff is $2 - 4P$. With the ability of the dove to decline trades, we can define the payoff function for the dove agent as follows:

Figure 3.5: Patterns in individual H1.3 experiments.

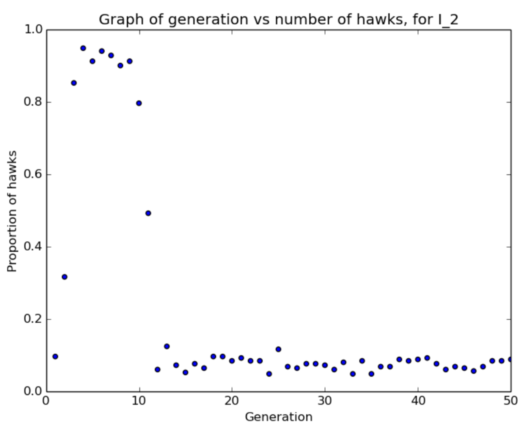
(a) Graph for a single I_c experiment.



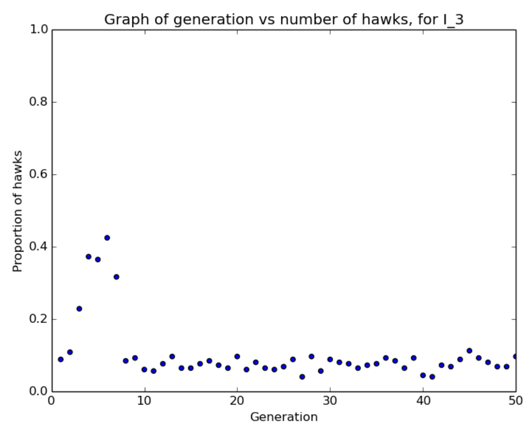
(b) Graph for a single I_1 experiment.



(c) Graph for a single I_2 experiment.



(d) Graph for a single I_3 experiment.



$$PAYOFF(P,R) = \begin{cases} 2-4P & if R \leq 1-P \\ 0 & otherwise \end{cases}$$

For any value of $P > 0.5$, the expected payoff for the dove is negative, and for any value of $P < 0.5$ it is positive. This means that with a value of $R > 0.5$, the conservative dove loses out on potential payoff[6] from any hawk with $0.5 > 1-P > R$ - despite the fact that the more conservative dove will be scammed less.

Similarly, we can show that playing $R < 0.5$ will result in negative payoffs for any transaction with a hawk with $0 < 1-P < R$. Thus, those doves who play values closest to $R = 0.5$ will obtain better scores when trading with hawks. This is quite clearly what we see in the results; that 50% gives the best score, and deviations above or below that result in reduced scores.

The symmetry of the observed graph originates in the fact that the payoff for being defected against is the same as the payoff lost, in opportunity cost, for declining an otherwise profitable transaction. Assume that a single dove agent (with a constant value of R) is interacting with a large number of hawks agents, with values of R as a perfect uniform distribution. The dove's average payoff per transaction will be:

$Payoff = R \times 0 + (1-R) \times (2(1-F) - 2F)$

Where F is the probability of a failed transaction - the dove being defected against. As a result of the R parameter reducing the pool of available hawks to only those with $P < 1-R$, we can establish that $F = \frac{1-R}{2}$, and thus:

$Payoff = (1-R) \times (2(1 - \frac{1-R}{2}) - 2\frac{1-R}{2}) = 2R - 2R^2$

$2R - 2R^2$ is a parabola with roots R = 0 and R = 1 - which represent being defrauded by approximately half the agents and trading with no agents respectively - and a maximum value at $R = 0.5$, representing the best strategy as discovered in the experimentation and analysis above. This parabola is also symmetric around $R = 0.5$ - showing that deviating from $R = 0.5$ in either direction causes the same loss in utility.

Experiment 1.2

H1.1 focuses on dove agents scoring better when cooperating with hawks who defect only P percent of the time. We can similarly imagine that hawks score better in the same circumstances.

In a similar manner as the dove payoff in H1.1, a hawk's payoff in terms of R and P is describable as:

$$PAYOFF(P,R) = \begin{cases} 2P+2 & if R \leq P \\ 0 & otherwise \end{cases}$$

We can see that less discerning doves (those with a lower value of R, and thus those which allow for a larger value of P before declining) will be more profitable for hawks to prey on. At $P = 0.5$ (the corresponding value of P for $R = 0.5$ from the doves, which we have previously established is the best strategy from a purely utility-maximising perspective) the expected payoff for a hawk is 3.

We can see from the results that this indeed holds up - that the average hawk does better the lower the average dove's value of R.

It is also worth noting that I_c and I_2 give similar values. I_c is a uniform distribution which means that there will be both doves with very high values of R (unprofitable) and doves with very low values of R (profitable). I_2 offers less doves with extreme values of R, but more of them that are able to be preyed on generally.

[6] also known as "opportunity cost"

Experiment 1.3

This experiment is all about the overall health of the system versus the individual success of the agents. In H1.1 and H1.2, we showed that both doves and hawks obtained better scores overall when doves were more lenient with how they dealt with hawks. In this experiment, we can say that - despite this personal benefit to trading with some hawks - the population as a whole does better (that is, it has fewer hawks in it) when the doves are more conservative.

By refusing to trade with most hawks, doves ensure that very few hawks make it to the breeding pool at the end of the simulation, thus ensuring the success of doves in the next generation. Furthermore, the high-R doves pass on their conservativeness to the next generation, and the pattern continues.

As the average value of R in the population increases, the acceptable range that a hawk's value of P can be while still making profit decreases, and the amount of deception that a hawk can get away with (and thus its score) is decreased significantly.

3.3.7 Conclusion

These three experiments have touched on very basic strategies surrounding use of the reputation system - first focusing on micro level properties of the system, and then extrapolating to longer-term properties when the system is run over a number of generations.

We discovered a dynamic by which dove agents can improve their own utility by playing one strategy ($R = 0.5$), while the general population of doves do better in the long run with another strategy ($R \approx 100\%$). This is a similar dynamic to the alarm call game as discussed in section 2.4.5.

We also discovered that hawks did better with a more mixed strategy - a combination of cooperates and defects - than with a strategy that has them defect all the time. The success of a hawk is not only resultant of its value of P, but also the dove population's values of R - in a population of doves with high R, a hawk with a very low value of P will do better than one with a larger value of P.

We will use a similar format to these three experiments in future experiments - first properties related to individual behaviour determined from mathematical principles, then long term analysis built on this foundation.

3.4 Experiment 2: False Feedback and Direct Experience

3.4.1 Introduction

This experiment deals with the idea that not all users will leave feedback that correctly represents the experience they had in trading - that is, hawks can perform slandering attacks (section 2.6.3) in order to make it more difficult for doves to differentiate hawks and doves.

As a defence, we will also allow dove agents to take into account their personal dealings with an agent, rather than just their feedback score.

There are two new genes added to the genome: false_feedback (F) and personal_experience (E) - these are both integer value genes (section 3.1.3). The strategies played by the agents are as follows:

Hawk - The hawk strategy is to defect prob_defect (P) of the time and cooperate the rest of the time. When giving feedback after a transaction, it has a false_feedback (F) percent chance of leaving false feedback: leaving a positive feedback to a defecting agent, and negative feedback to a cooperating one.

Dove - The dove strategy is to cooperate with any agent that they think will cooperate with them more than rep_weighting (R) of the time. The value of personal_experience determines how much an agent weights previous experiences with that agent (of which they have 100% reliable knowledge of) versus the potentially unreliable information from the reputation system. If the other agent has reputation scores Rep_p and Rep_n (for positive and negative respectively), and has had direct experience with the agent Tru_p and Tru_n, then the predicted cooperation chance is equal to:

$$PredictedP = \begin{cases} \dfrac{Rep_p}{Rep_p + Rep_n} & if Tru_n + Tru_p == 0 \\ \dfrac{1024 - E}{1024} \times \dfrac{Rep_p}{Rep_p + Rep_n} + \dfrac{E}{1024} \times \dfrac{Tru_p}{Tru_p + Tru_n} & otherwise \end{cases}$$

These additions allow us to simulate and experiment with common attacks on reputation systems (orchestrated slander and promotion - as discussed in section 2.6.3), and give legitimate users a useful defence as well as simulating a distinctly human behaviour - we distrust those who have wronged us in the past, even if they are seen as a trustworthy actor by everyone else.

3.4.2 Parameters

There are two new parameters for this experiment - the probability functions for F and E in the first generation. As in section 3.3.2, the default values are uniform distributions, with different distributions offering ways of simulating different starting conditions of the system.

The two new genes must have a mutation method, and they will use the same method as described in experiment 1 (section 3.3.2), as all four genes are integer value genes (section 3.1.3).

3.4.3 Hypotheses

H2.1 - In one generation, the effectiveness of doves having higher values of E scales with the total number of transactions undertaken.

H2.2 - In one generation where hawks have higher values of F, doves will score lower on average.

H2.3 - In the long term, a population beginning with hawks with highest values of F will contain a larger proportion of hawks for a larger amount of time.

3.4.4 Method

Experiment 2.1

In this experiment, we are trying to determine the impact of the total number of transactions on the effectiveness of a dove strategy with regards to the E parameter.

If the N_AGENTS (N) and TRANSACTIONS_PER_AGENT (T) are equal, E does not have much impact - on average, each agent will trade with each other agent exactly once. Thus, for each transaction, they are unlikely to have any personal experience to draw from.

As the ratio of $T : N$ increases - and thus, the number of times, on average, each agent interacts with each other agent increases - the reliability of the personal experience information increases. We should observe that the effectiveness of higher values of E scales with the ratio of $T : N$.

Table 3.16: Experiment 2.1 parameters

Parameter	Value
N_AGENTS	250
N_GENERATIONS	1
INITIAL_HAWK	40%
prob_defect (P)	Normal distribution, mean = 300, SD = 150.
rep_weighting (R)	Normal distribution, mean = 500, SD = 150.
personal_experience (E)	Uniform distribution $1 \leq E \leq 1023$.
false_feedback (F)	Normal distribution, mean = 500, SD = 150.
Repeats	10

We will run 10 repeats of a single generation experiment with each of the values $T : N$ - 1, 2.5, 5, 7.5, 10. We will then plot the graph of average dove score per transaction (D) against E for each value of $T : N$.

The parameters for this experiment are in table 3.16, and they require some discussion. Firstly, the value of INITIAL_HAWK: we assume that the effectiveness of false feedback scales with the total number of hawks in the population. This assumption seems to be reasonable as a result of some preliminary experimentation and as a result of false feedback needing to exist in some noticeable proportion in order to impact the decision making of doves. Thus, in order to ensure that we actually see an impact of false feedback, we need to have a large number of hawks in the population.

The values of P and R are set using the knowledge gained from experiment 1 - we know that dove agents with R around 50% do better than any others, and that if a hawk agent's P is higher than the average R, that hawk will do badly. Thus, this is set up to make sure that (not taking into account F and E), most hawks will be able to defect against at least some dove agents.

The values of F are clustered around 50% so that the quantity of false feedback is roughly consistent between simulations, and the value of E spreads over the entire possibility space, as that is the value we wish to test.

Initially, this experiment was designed to use values of $T : N$ of 1, 2, 5, 10, 25. However, due to technical difficulties during running this experiment, the new values were used.

Experiment 2.2

This experiment is about the impact of more false feedback in the system on doves' average score. We will be using four values of F for this experiment - the I values previously defined in table 3.2.

We will run 50 one-generation experiments for each probability distribution I, and observe the average dove score per transaction (D).

We should observe that:

- $D(I_3) < D(I_C)$, $D(I_3) < D(I_1)$, $D(I_3) < D(I_2)$ - Higher values (around 70%) of F will result in lower dove scores than any other value of F.
- $D(I_2) < D(I_1)$ - F values around 50% will result in lower doves scores than F values around 30%.

First, we will perform single-factor ANOVA ($p < 0.05$) to ascertain whether there are significant differences between the results.

Then, for each of the four inequalities above, a t-test will be performed. We will use a standard test of $p < 0.05$, which will be adjusted to $p < 0.0125$ by Bonferroni's method to compensate for the multiple t-tests.

Table 3.17: Experiment 2.2 parameters

Parameter	Value
N_AGENTS	250
TRANSACTIONS_PER_AGENT	2500
N_GENERATIONS	1
INITIAL_HAWK	40%
prob_defect (P)	Normal distribution, mean = 300, SD = 150.
rep_weighting (R)	Normal distribution, mean = 500, SD = 150.
personal_experience (E)	Normal distribution, mean = 500, SD = 150.
Repeats	50

The parameters for this experiment (table 3.17) are set as they are to balance the amount of data versus the computational requirements - TRANSACTIONS_PER_AGENT is set to $10 \times N\_AGENTS$ due to reasons discussed in section 3.4.4.

The values of P and R are drawn from Experiment 1, so that we can draw from our previous knowledge in our analysis. personal_experience is set to around 50%, which is approximately an even weighting of reputation and trust (section 3.4.1) - the idea being to give the chance for doves to overcome the disadvantage of the F parameter.

Experiment 2.3

This experiment is about the long-term impact of hawks leaving false feedback on a population. This experiment is very similar to that in Hypothesis 1.3, where we discovered that - without the newly added features in Experiment 2 - eventually the minority of dove agents would evolve large values of R and take over the population.

In this experiment, this may play out differently - it is unlikely that a hawk agent with a very large value of R will ever get trades with any other dove agents (as they will probably have some amount of negative feedback on them, despite their honestly.)

We will be using four values of F for this experiment, the probability distributions I previously defined in table 3.2.

We are measuring the number of generations (out of 25) where the percentage of hawks is above 20% (H). We should either find that the following inequalities apply, or that in every case $H(I) = 23...25$:

- $H(I_3) > H(I_C)$, $H(I_3) > H(I_1)$, $H(I_3) > H(I_2)$ - Beginning with a population with F values around 70% will result in hawks being a significant part of the population for a longer period of time, than when the population uses any of the other F values.
- $H(I_2) > H(I_1)$ - Beginning with a population with F values around 50% will result in hawks being a significant part of the population for a longer amount of time, than when the population begins with F values around 30%.

In the latter case (that hawks dominate the vast majority of generations), we can reasonably conclude that the inclusion of this F parameter even in relatively small quantities makes sustainable cooperation impossible. In this case, we will run one large experiment with a high number of generations to see if this trend holds over long periods.

In the former case, we can conclude that inclusion of the F parameter makes sustained cooperation more difficult, but it is still possible to overcome it in the long-run.

Table 3.18: Experiment 2.3 parameters

Parameter	Value
N_AGENTS	100
TRANSACTIONS_PER_AGENT	1000
N_GENERATIONS	25
INITIAL_HAWK	40%
prob_defect (P)	Normal distribution, mean = 300, SD = 150.
rep_weighting (R)	Normal distribution, mean = 500, SD = 150.
personal_experience (E)	Normal distribution, mean = 500, SD = 150.
Repeats	25

The computational requirements of this experiment are very high, and this has led to some cutbacks having to be made to ensure that it is plausible. Ideally this experiment would run for 50 generations (as E1.3 did) with double or maybe quadruple the number of agents. As is, the experiment will take about 48 hours of computational time running on all four cores of an AMD A8 processor.

The other parameters are relatively simple - INITIAL_HAWK is required to be quite large in order for F to be effective, and the probability distributions for the genes have been utilised before in previous experiments, and are being used again so that we an draw on past experiments in our analysis.

3.4.5 Results

Experiment 2.1

The graphs of E versus score per transaction (S) are in the appendix section A.3.1. We discovered that none of the experiments showed a significant advantage for any particular value of E.

We did, however, find that the average S increased (and the spread of S decreased) as $T : N$ increased - the data for which is available in table 3.19. The latter is probably explainable in a similar manner to that of Experiment 0.3 (section 3.2.6), but the former requires some more discussion, which will be done in the analysis.

Table 3.19: Experiment 2.1 results - analysis of different $T : N$ ratios.

-	Mean score (SD)	Average SD score	Outliers[7] (SD)
$T : N = 1$	1.33 (0.12)	0.35	23.4 (9.61)
$T : N = 2.5$	1.45 (0.11)	0.36	16.8 (8.56)
$T : N = 5$	1.58 (0.059)	0.29	8.9 (4.08)
$T : N = 7.5$	1.57 (0.065)	0.29	9.2 (5.04)
$T : N = 10$	1.61 (0.072)	0.265	6.8 (4.85)

Experiment 2.2

The results for dove scores per transaction for each value of F are available in table 3.20.

Performing the one-factor ANOVA test on these four values gives us $p = 3.68 \times 10^{-22}$, which is strongly significant (at our $p < 0.05$).

[7] Number of agents where S ¡ 1

Table 3.20: Experiment 2.2 Results - dove scores for each value of F.

-	D Mean	SD
I_c	1.63	0.039
I_1	1.57	0.037
I_2	1.63	0.029
I_3	1.63	0.034

T-tests (one-tailed):

- $D(I_3) < D(I_c)$ - $p = 0.34$ - $p \nless 0.0125$.
- $D(I_3) < D(I_1)$ - $p = 4.6 \times 10^{-15}$ - $p < 0.0125$.
- $D(I_3) < D(I_2)$ - $p = 0.50$ - $p \nless 0.0125$.
- $D(I_2) < D(I_1)$ - $p = 1.75 \times 10^{-16}$ - $p < 0.0125$.

Experiment 2.3

The table in table 3.21 contains the values of the mean (and SD) number of generations where hawks make up more than 20% of the population, against the different values of F.

Table 3.21: Experiment 2.3 Results - number of generations where hawks are a significant part of the population for each value of F.

-	Mean	SD
I_c	14.8	6.47
I_1	10.08	1.09
I_2	13.64	5.60
I_3	23.88	2.49

Performing the one-factor ANOVA test on these four values gives us $p = 3.64 \times 10^{-17}$, which is strongly significant (at our $p < 0.05$).

T-tests (one-tailed):

- $H(I_3) > H(I_c)$ - $p = 1.92 \times 10^{-7}$ - $p < 0.0125$.
- $H(I_3) > H(I_1)$ - $p = 3.13 \times 10^{-23}$ - $p < 0.0125$.
- $H(I_3) > H(I_2)$ - $p = 9.1 \times 10^{-10}$ - $p < 0.0125$.
- $H(I_2) > H(I_1)$ - $p = 0.0025$ - $p < 0.0125$.

3.4.6 Analysis

Experiment 2.1

The results of this experiment were not as we expected. We did not observe any difference in an individuals score related to their value of E. As mentioned in the results section, we did find two factors that require explanation: firstly, that the spread seems to decrease as $T : N$ increases; secondly that the average score S increases as $T : N$ increases.

We can explain the first factor in the same way as Experiment 0.3 (section 3.2.6). As we run more transactions the spread decreases, as the impact of randomness on one particular transaction decreases. This gives us results that are more indicative of the "true" value and properties of the system.

The second factor - the higher average scores per transactions as $T : N$ increases - is slightly more difficult to explain. There are two factors that could, either separately or together, explain this: that this property is true because of a reason similar to the first (e.g. it is a property of reduced impact of randomness), or that there is some impact of E at higher $T : N$ that, while not impacting an individual's score, helps the scores of everyone.

We devised an informal experiment as a launching point for possibly future experimentation. The results of this are available in section A.3.2. We can see in this informal experiment that there is not a lot of difference between a population with uniform E and a population with $E = 0$. More rigorous experimentation and more statistical power could possibly show a difference between the two groups, but this shallow look into the problem would seem to imply that E has very little impact on the overall result, at least for $T : N \leq 10$.

Experiment 2.2

We would expect that if there is more false feedback in the system, doves get lower scores. We partially showed this during the experiment, as two out of four t-tests passed. This is possibly explainable as a limitation of the amount of data (as determined by the computational requirements), and also is likely to do with the influence of the personal_experience parameter. The theory to follow (and the theory that went into devising this experiment) did not take E into account - a more rigorous experiment might be to split this into two sections - one where $E = 0$, and then one where $E = Normal(500, 150)$.

In order to explore this theoretically, we will draw from the analyses of experiments 1.1 and 1.2 and how the values of R and P interact. Hawks leave false positive (positive feedback on users who defected against them) and false negative (negative feedback on users who cooperated with them) with the same probability, which leaves us with two distinct issues to discuss.

False positive feedback means that doves have an inaccurate picture of what value of P the hawk is playing. Modifying the payoff function from section 3.3.6:

$$PAYOFF(P, R) = \begin{cases} 2 - 4P_{true} & if R \leq P_{estimate} \\ 0 & otherwise \end{cases}$$

Where P_{true} is the value of P the hawk is actually playing, while $P_{estimate}$ is the value of P that the dove predicts from that agent's reputation. As we established in experiment 1.1, when $P_{true} < 0.5$, the dove agent loses out on average. With false feedback, it is possible for both $P_{estimate} > 0.5$ and $P_{true} < 0.5$, meaning that the dove will predict a positive outcome while actually getting a negative one.

The other kind of false feedback - false negative feedback - impacts doves in a different way, making it more difficult to differentiate doves and hawks. If we look at this modified version of the payoff function above for when two doves meet:

$$PAYOFF(P, R) = \begin{cases} 2 & if R \leq P_{estimate} \\ 0 & otherwise \end{cases}$$

While the other (dove) agent will cooperate 100% of the time, it is possible for $P_{estimate}$ to be lower than R (especially if R is very high), thus providing 0 payoff for both doves involved.

From this, it means that values of R have to be lowered population-wide in order to account for the false negative feedback and facilitate any trade. Reduced values of R allow hawks to play higher values of P and thus defraud more users.

Experiment 2.3

This experiment is a longer-term analysis of the impact of false feedback on the population over time. This is essentially a parallel of the experiment 1.3, as in both experiments we are looking at the number of generations that hawks are a significant part of the population.

In experiment 2.2 we showed that larger amounts of false feedback cause lower dove scores. We would expect this to result in fewer doves at the top of the scoreboard, and thus a larger proportion of hawks in the breeding pool.

From this, we would expect that more false feedback results in more hawks for a longer amount of time, as doves find it more difficult to differentiate hawks and doves. This is what we saw in our experiment, with all four of our t-tests showing strong differences between the hawk population with different levels of false feedback.

3.4.7 Conclusion

We investigated our two new parameters to give a more sophisticated and more "human" approach to decision making by our dove agents, and gave more offensive strategies to hawk agents. We had difficulties in some of these experiments with the introduction of two new complex genes making observed effects more difficult to explain. With more time, experiments 2.2 and 2.3 could be re-run both with and without the E gene - and also with higher values of $T : N$ to see if the basis of hypothesis 2.1 holds with larger values than we managed.

Despite these difficulties we managed to gain insights into how this particular attack on the reputation system (slandering - see section 2.6.3) impacts the system as a whole. As expected, the inclusion of false feedback negatively impacted doves on an individual, population and generational level. Due to the difficulties surrounding the E gene as explained above, it is difficult to tell whether or not E would - on a larger scale - serve as a defence against false feedback. The idea is that trust (personally gathered) information is infallible, and thus weighting it more heavily than reputation information (which is fallible) is positive overall - but it may be possible that the number of times an agent has to trade with another in order to establish P_{true} (section 3.4.6) is prohibitively high within the confines of these experiments, and especially prohibitively high within the context of the real world online marketplace that is the underlying problem.

3.5 Experiment 3: Altruistic Punishment

3.5.1 Introduction

This experiment deals with the idea of altruistic punishment, a behaviour displayed in a variety of animals (including humans) by which they will choose to punish a defector, even when it costs them some amount of utility. In terms of our experiment, dove agents will gain the ability to - at a cost of one point of utility - leave three units of negative feedback instead of one.[8]

There is one new gene added to the genome: altruistic_punishment (A) - which is an integer value gene (section 3.1.3). Additionally, in this experiment, personal_experience (E) will be removed[9] - we decided

[8]The cost and impact of this feature is entirely changeable, and the value of 1 point for 2 extra negative feedback is based off of the previous values of R and P we used during experiment 1.

[9]Set to 0, so that only the reputation and not the trust value is taken into account.

that, given some issues with conflating variables in Experiment 2, we would simplify this experiment.

The strategies played by the agents are as follows:

Hawk - As experiment 2. Plays "defect" P percent of the time and cooperates the rest of the time. After a transaction, it has a F percent chance of leaving false feedback.

Dove - Cooperate with any agent that they believe will cooperate more than R percent of the time. After a transaction, if the other agent defected against them, utilise the altruistic punishment option A percent of the time.

In a similar manner to the R parameter in Experiment 1 - and the alarm call game as discussed in section 2.4.5, we should see that agents with a lower value of A do individually better, but if there are more agents with higher values of A in the population, every dove should do better.

3.5.2 Hypotheses

H3.1 - In one generation, doves with higher values of A will score lower than doves with lower values of A.

H3.2 - In one generation, a population of doves with a higher value of A will do better on average than a population of doves with lower values of A.

H3.3 - In the long term, a population beginning with doves with higher values of A will contain a significant portion of hawks for a shorter amount of time.

3.5.3 Method

Experiment 3.1

In this experiment, we are trying to determine the individual impact on a dove agents' score based on how often they use the "altruistic punishment" function (their A gene).

We should see that, as A increases, the doves' average score per transaction (S) decreases - mostly because undertaking altruistic punishment costs the dove utility, on top of the utility they already lost as a result of being defected against.

We will plot the graph of A (which will be defined as a uniform distribution $1 \leq A \leq 1023$) against S, and observe its characteristics. We should observe a negative linear correlation - we will use the same criteria for correlations as used in Experiment 1.1 (section 3.3.4), where $PMCC \leq 0.5$ is a weak correlation, $0.5 < PMCC \leq 0.75$ is a moderate correlation and $0.75 < PMCC \leq 1$ is a strong correlation.

Table 3.22: Experiment 3.1 parameters

Parameter	Value
N_AGENTS	1000
TRANSACTIONS_PER_AGENT	2500
INITIAL_HAWK	40%
prob_defect (P)	Normal distribution, mean = 300, SD = 150.
rep_weighting (R)	Normal distribution, mean = 500, SD = 150.
personal_experience (E)	0[10]
false_feedback (F)	Normal distribution, mean = 500, SD = 150.
Repeats	10

The parameters for this experiment are available in table 3.22. As previously discussed, the personal_experience (E) parameter has been removed from this set of experiments. For the other gene values, we used two probability distributions we have used before - with P being smaller than R on average in order to ensure that hawks are able to get some successful defections in. For F, we used a simple normal distribution to ensure that there is a consistent amount of false feedback in the system.

The values of N_AGENTS and TRANSACTIONS_PER_AGENT are defined to maximise the data collected within the computational limitations. This experiment was fairly computationally light - only taking about two hours to complete - compared to the other experiments in this section.

Experiment 3.2

In this experiment, we are attempting to establish the population-wide impact of the value of A on the doves in the population. We should see that, as the quantity of altruistic punishment in the system increases, so does the average dove score.

We will test four different probability distributions for A - the four values of I as defined in table 3.2.

First, we will perform a one-factor ANOVA test ($p = 0.05$) to test for differences between the values of D. We will then perform the following t-tests ($p = 0.0125$, as before) after observing the average dove score per transaction (D):

- $D(I_3) > D(I_C)$, $D(I_3) > D(I_1)$, $D(I_3) > D(I_2)$ - When doves have a value of A around 70%, they score better than if the value of A is any other value.
- $D(I_2) > D(I_1)$ - When doves have a value of A around 50%, they score better than if the value of A is around 30%.

Table 3.23: Experiment 3.2 parameters

Parameter	Value
N_AGENTS	250
TRANSACTIONS_PER_AGENT	1000
INITIAL_HAWK	40%
prob_defect	Normal distribution, mean = 300, SD = 150.
rep_weighting	Normal distribution, mean = 500, SD = 150.
personal_experience	0[11]
false_feedback	Normal distribution, mean = 500, SD = 150.
Repeats	50

The parameters for this experiment are available in table 3.23. The probability distributions for the genes are the same as used in Experiment 3.1 (section 3.5.3). The size of each experiment has been reduced with a corresponding increase to the number of repeats in order to increase statistical power in the t-tests that will be performed.

Experiment 3.3

In this experiment we are attempting to show the long-term impact of higher values of A in the dove population. This experiment is very similar to Experiment 2.3 in both its design and purpose.

[10] See section 3.5.1
[11] See section 3.5.1

We will run 25, 25-generation experiments where the initial value of A is defined by the table of I (I_c, I_1, I_2 and I_3 as restated in section 3.5.3) and observe the number of generations that hawks are a significant part of the population (where they make up $\geq 20\%$ of the total population).

First we will perform a one-factor ANOVA test ($p = 0.05$) to test for differences between the values of H. We will then perform the following t-tests ($p = 0.0125$, as before):

- $H(I_3) < H(I_C)$, $H(I_3) < H(I_1)$, $H(I_3) < H(I_2)$ - Beginning with a population with A values around 70% will result in hawks being a significant part of the population for a shorter period of time, than when the population uses any of the other A values.
- $H(I_2) < H(I_1)$ - Beginning with a population with A values around 50% will result in hawks being a significant part of the population for a shorter period of time, than when beginning with a population of A values around 30%.

Table 3.24: Experiment 3.3 parameters

Parameter	Value
N_AGENTS	250
TRANSACTIONS_PER_AGENT	1000
N_GENERATIONS	25
TOP_X	30%
INITIAL_HAWK	40%
prob_defect (P)	Normal distribution, mean = 300, SD = 150.
rep_weighting (R)	Normal distribution, mean = 500, SD = 150.
personal_experience (E)	0[12]
false_feedback (F)	Normal distribution, mean = 500, SD = 150.
Repeats	25

The parameters for this experiment are available in table 3.24. The gene probability distributions are the same as Experiment 3.1 (section 3.5.3). The number of repeats, as well as the values of N_AGENTS, TRANSACTIONS_PER_AGENT, have been tweaked given the computational requirements of running 25 generations a repeat. It is worth noting that this is by far the most computationally demanding experiment in this entire paper, taking twelve hours of consistent running over two computers to complete.

3.5.4 Results

Experiment 3.1

The results for this experiment are somewhat fragmented, for reasons discussed in section 3.5.5. The graph of A against dove scores is figure 3.6, and the linear regression figures are in table 3.25.

There are two additional runs of this experiment which are included in the appendix, section A.4.1 and section A.4.2.

Experiment 3.2

The observed data for this experiment is available in table 3.26.

[12] See section 3.5.1

Figure 3.6: Experiment 3.1 results

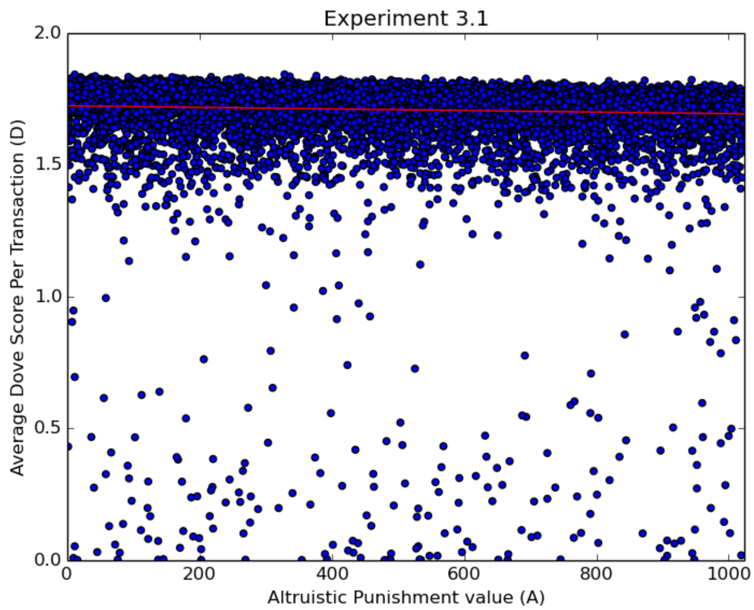


Table 3.25: Experiment 3.1 - linear regression of the correlation between A and dove scores.

-	$A - DCorrelation$
m	-2.98×10^{-5}
c	1.72
PMCC	-0.0494

Performing the one-factor ANOVA test on these four values gives us $p = 1.86 \times 10^{-5}$ which is strongly significant (at our $p < 0.05$).

T-tests (one-tailed):

- $D(I_3) > D(I_c)$ - $p = 0.028$ - $p < 0.0125$
- $D(I_3) > D(I_1)$ - $p = 5.08 \times 10^{-6}$ - $p < 0.0125$
- $D(I_3) > D(I_2)$ - $p = 0.0067$ - $p < 0.0125$
- $D(I_2) > D(I_1)$ - $p = 0.08$ - $p < 0.0125$

Experiment 3.3

The observed data for this experiment is available in table 3.27.

Performing the one-factor ANOVA test on these four values gives us $p = 0.44$, which is strongly significant (at our $p < 0.05$).

T-tests (one-tailed):

- $H(I_3) < H(I_c)$ - $p = 0.335$ - $p \not< 0.0125$

Table 3.26: Experiment 3.2 - dove scores against values of A.

-	D Mean	SD
I_c	1.70	0.031
I_1	1.68	0.042
I_2	1.69	0.039
I_3	1.71	0.037

Table 3.27: Experiment 3.3 - number of generations that hawks are a significant part of the population for each value of A.

-	H Mean	SD
I_c	16.5	2.83
I_1	15.6	1.90
I_2	15.4	2.51
I_3	16.1	3.00

- $H(I_3) < H(I_1)$ - $p = 0.222$ - $p \not< 0.0125$
- $H(I_3) < H(I_2)$ - $p = 0.185$ - $p \not< 0.0125$
- $H(I_2) < H(I_1)$ - $p = 0.402$ - $p \not< 0.0125$

3.5.5 Analysis

Experiment 3.1

This experiment did not turn out as expected: the correlation coefficient is near zero (meaning no correlation) and the graph is rather spread and seemingly random. There are two distinct issues here, for which we produced two informal re-experiments to address.

First, however, it is important to discuss the mathematical underpinnings of this experiment. Altruistic punishment works by an honest agent punishing a dishonest agent at the cost of its own utility, and the A gene determining how often a dove agent punishes a hawk agent after being defected against.

If $Cost$ is the cost of using the altruistic punishment option, the average dove's payoff when being defected against is:

$-2 - (A \times Cost) = -2 - A$

We can see that dove agents with larger values of A will score worse in these transactions.

As the other agent an agent trades with is random, we expect that the benefit a particular dove agent gets using altruistic punishment is the same as the other doves get - and is probably less than the utility cost for the individual. We may find that (as is the topic of experiment 3.2 and 3.3) a number of dove agents doing this together make it worthwhile, but on an individual basis, we would expect that dove to do worse as a result.

If we look at the graph in figure 3.6, we can see two distinct areas: the area at the top with a number of points clustered together within a thin area, and the outliers scattered around at the bottom.

The outliers could be a result of the F parameter: in the first few transactions of each simulation, a dove agent can be unlucky and meet a hawk agent which leaves negative feedback for it. Having the grand total of 1 point of feedback (negative), means that no other doves will trade with it until it's feedback score has increased, which it has to do by trading with hawks (which will often defraud them, sometimes even leave even more false negative feedback).

Depending on chance, the dove may spend anywhere between a few trades and the entire simulation in this state, leading to the spread of score values as seen in the experiment. We decided to run an informal experiment that set the F gene to 0 in all hawks, which is available in section A.4.1. We can observe the graph and the linear regression, and see that removing the F parameter has this predicted effect.

However, in this formal experiment, we can see that the PMCC value is -0.3 - as defined by our method; only a weak correlation.

In a population that is 60% dove and 40% hawk, we can calculate that only 48% of the transactions are between one dove and one hawk - and a significant portion of these end up resulting in a transaction, due to the fact that doves have the ability to decline transactions with hawks they believe will defect.

Despite the fact that agents with larger values of A will do worse in these transactions, if they are not a significant proportion of the total transactions and the cost is minimal (1 point of utility), we could reasonably expect other factors (such as P and R) to eclipse A in terms of impact on the total score, leaving a fairly weak correlation between score and A.

To test this, we created another informal re-experiment, the results of which are available in section A.4.2. Reducing the variances of R and P in order to minimise the interference those variables provide, we can see from the results that the scores are packed more densely, and that the correlation coefficient is sufficient to call it a moderate correlation by our method's definition.

Experiment 3.2

While agents do worse individually with higher values of A (Experiment 3.1), the principle is that doves in general do better as a group when a significant portion of doves have high values of A. Two of out of four of our t-tests passed, the two involving I_3 (the distribution that clusters around higher values) - $D(I_3) > D(I_2)$ and $D(I_3) > D(I_1)$. There are three possible explanations for this, either separately or in some combination.

The first possibility is that there are issues similar to those in Experiment 3.1 (section 3.5.5), the other genes (R and P) have more of an impact on the result than the A gene, and - because of the high variance in these genes - the value of A has less influence in the final result. If there were more time, it would be possible to re-run this experiment with the same values of P and R as Experiment 3.1's re-experiment to see how much impact this has.

The second possibility is that the impact of an increase in the value of A differs - which is to say that the impact of changing the mean value of A to 500 from 300 may be less than the impact of changing it from 700 to 500. Given more time, experiments could be drawn up to observe the score output for a large range of A values.

The third possibility is that there were simply not enough repeats to provide enough statistical power for these t-tests to pass. Again, given more time, it would be possible to run these experiments with a larger number of repeats.

Despite these problems, we can see from the two t-tests that did pass that the higher level (I_3) of A values does result in a higher score, as we would expect. Theoretically, as more dove agents use the altruistic punishment option, the hawks collect more negative feedback and thus the doves find it easier to identify and avoid them, leading to fewer doves being defrauded, and higher scores for the dove agents and corresponding lower scores for the hawk agents.

Experiment 3.3

Given the theory and results of Experiment 3.2 (section 3.5.5), we expected that we would see an impact on hawks over a multiple-generation experiment. That is, given that doves do better when A is larger, we would expect that more doves would be a part of the breeding pool and that therefore hawks would be a less significant part of the population.

However, all four t-tests in these experiments failed: the number of generations that is $> 20\%$ hawk are indistinguishable for all four values of A that we tested. The reasons for this are likely to be those discussed previously in Experiment 3.1 (section 3.5.5) and Experiment 3.2 (section 3.5.5). Due to time constraints, re-formulating this experiment and re-running it is not possible - we will instead discuss what it is that could be changed to possibly get a positive result.

Firstly, the removal of the false feedback parameter, as in Experiment 3.1's re-experiment (section 3.5.5). The false feedback causes a small portion of the dove population to lose a significant chunk of their potential score, and is an extra factor that obfuscates the effect of A.

Secondly, the reduction of the variance in R and P, again as in Experiment 3.1's re-experiment (section 3.5.5). This reduced variance will allow the impact of A to be more apparent by comparison.

Finally, altering of the effect of A. By default, the effect of using the altruistic punishment option is 1 point of utility, and it causes 2 extra points of negative feedback (from 1 by default to 3). This is entirely customisable, however, and amplifying the effect and cost of this option could make it more apparent in the results.

3.5.6 Conclusion

This experiment was less successful than the other experiments in terms of statistical tests passing, yet still provided useful information about the system, and allowed us to discuss how the many factors intersect and interact. The main reason for the difficulties with these experiments comes from the conflicting issues of system complexity and computational resources. A better experiment design given the constraints perhaps could have been to implement the new experiment 3 system off of the experiment 1 system, rather than the experiment 2 system, to begin with - thus allowing us to focus on the new gene exclusively rather than having to take into account and adjust for E and F from experiment 2.

Despite these difficulties, the altruistic punishment option serves as an worthwhile practical exploration of an issue covered in the literature review (section 2.5.3) as well as a suggestion of a feature for an online marketplace; the real-world problem the P2PTG is supposed to represent. Providing an altruistic punishment option on eBay, for example, could possibly allow for fraudulent users to be identified by genuine ones more quickly.

While it has potential benefits as a feature, there are issues that need to be discussed before such an implementation would be possible: how much it would cost and its impact (this abstract implementation in the P2PTG had the issue of not being potent enough to have a distinguishable impact), whether or not (enough) users would actually use the feature, and - possibly most importantly - how to implement it in such a way that makes it untenable for it to be used maliciously.

While in experiment 2 we considered false feedback, we did not consider the idea of hawk agents using the altruistic punishment option - and this is an issue that would have to be dealt with before it could be rolled out to users. We would anticipate the best hawk strategy being to avoid the use of the altruistic punishment option completely - after all, the entire idea is that it is a selfless personal utility cost for the benefit of the whole population - but it is possible to imagine a scenario where an attacker orchestrates a targeted slander campaign to ruin one particular user's reputation and their ability to earn.

It is also possible that malicious hawk use of the altruistic punishment feature could cause a similar dynamic to when doves use it: individual hawks using the feature lose utility, but all hawks benefit from it. We discovered in experiment 2 that false feedback in a system benefits hawks greatly, and the altruistic punishment option is a method of injecting even more false feedback into the system, at a cost. It would be possible to run an experiment or set of experiments involving this, and establish to what extent the altruistic punishment feature is exploitable, and what - if any - parameters could be altered to avoid that problem.

Chapter 4

Conclusions

4.1 Conclusions

4.1.1 Analysis of the P2PTG

The first stage of this project was to formulate the P2PTG, a game theoretical model of the online marketplace scenario described in section 1.1. The purpose of the P2PTG is to be an abstract model that draws from both the aforementioned online marketplace systems and from game theory research. Some of the limitations of the model have been previously discussed in section 2.2.2, but there are a few in particular that merit further discussion.

Firstly, the roles of buyer and seller: the P2PTG is game theoretically symmetric (section 2.3.2) for reasons of simplicity and so that it can be more easily related to other game theory problems. However, buyers and sellers on an online marketplace have different threat models: for instance, there is no way a seller can be impacted by fake goods, and no way that a buyer can be impacted by payment fraud (section 2.2.2). On top of this, real online marketplaces have users who buy and sell in different proportions and perform different numbers of transactions - a large eBay seller account may participate in thousands of transactions as a seller but none as a buyer, whilst a casual user may buy or sell only a few items at a time. The trust decision between trading with a long-standing shop-like user versus with a casual user will be entirely different, and - while a system that takes into account reputation length was suggested in the P2PTG (section 2.8.5) - we can not expect any system implemented to capture the many complexities of a real system.

Tangential to the previous point, there is the issue of the frequency of fraudulent transactions: [2] places the rate of accusations of fraud at 0.2% of transactions, and, as it is unlikely that a malicious user would only perpetrate one fraudulent transaction, it would appear that malicious users (hawks) are only a small proportion of the population. The P2PTG model and the experiments conducted assume a significant amount of hawks, at the very least 10%, as having hawks be a substantial portion of the population is required in order to obtain any results that are distinguishable, let alone statistically significant - it is likely that a 1000 agent population with only 10 hawks, at 1%; this is considerably higher than the given real-world estimate and would have the hawks be an absolute non-issue to the doves' overall performance.

Of course, the real marketplace is not the P2PTG, and a real user being defrauded is not comparable to a dove agent losing two points of utility; even the small fraction of malicious users can cause serious disruption and financial hardship to real, living humans. The issue with the P2PTG using 10% or more hawks in the population is that anything we learn from it will have the caveat of not necessarily being applicable to a system where hawks are considerably rarer. It is a lot easier for your strategy to produce false positives on malicious users when it is built off of the assumption that there are more of them than there actually is. It may also be considerably harder to test how a strategy works on a real marketplace for reasons both practical - it being harder to find malicious users to test against - and financial - it being expensive to sell or buy a sufficient quantity of items on a marketplace in order to test.

Despite these limitations and despite the level of abstraction; there are properties of the model that allow us to draw notable and useful conclusions. The greatest benefit of the model is its flexibility and extensibility: the fact that the system can be re-implemented as desired and extended to include further considerations and factors. We discuss possible future extensions in section 4.2.2.

4.1.2 Analysis of the Implementation

The system is implemented in object-oriented Python 3.4, using numpy and matplotlib for statistical analysis. The design is modular with an emphasis on ease of implementing new genes - adding a new gene requires extending the gene class, modifying a line in the parameters and adding the functionality for the gene into the transaction handling code. Implementing the additional genes and functionality for experiments 2 and 3 took approximately ten minutes each due to this design.

The main problem with the system - and with this implementation - is the computational requirements. Many of these experiments took upwards of 12 hours using the entire capacity of a mid-range laptop, some later experiments used several days worth of processing time on a high-end desktop. There are a number of possible ways the system could be modified to alleviate this. Re-implementing the system in C, or modifying it to use Cython (a extension for Python that pre-compiles critical pieces of code to improve performance) would reduce the computational requirements, or using distributed computing (e.g. Amazon AWS, a cloud computing service) to gain access to more computing power, could allow us to run larger experiments while still keeping the time constraints manageable; albeit the latter coming at a financial cost.

The source code is available on github at `www.github.com/ChrsT/auction-simulation-dissertation`, and also on disc as attached to this paper. Code is licensed under the General Public License version 3 (GPLv3) - further details of this license is available at `www.gnu.org/licenses/#GPL`.

4.1.3 Findings

After implementing the system, we conducted a series of experiments on it in order to establish the emergent properties of the system, to see how modifying and adding certain factors influences the end result, with a mind to applying these findings to the real life scenario (section 4.1.4).

To begin with, we dealt with a system very similar to the Hawk-Dove game (section 2.4.4) - this was not the primary purpose of this project, and it was used primarily for establishing parameters for the other experiments. The Hawk-Dove game has been covered extensively in (evolutionary) game theory literature and it would be more than possible to formulate and conduct considerably more experiments on this basic system - but these very simple experiments do not suit our real-world scenario.

The basic reputation modification to the experiment represented a very simplified set of strategies - doves check whether the other agent's feedback percentage is above a certain value, and hawks defect a certain percentage of the time. This presents a simple dilemma for the hawk agents: they want to defect in order to get a utility advantage, but if they do it to excess, some dove agents will not trade with them any more, thus causing a loss in future utility.

Combining mathematical analysis and short-term experimentation, we established that dove and hawk agents both have better scores when doves are more liberal with trading with hawk agents (when they play $R = 0.5$, see Experiments 1.1 and 1.2). This is, however, at the expense of the long-term health of the system - when all doves are more conservative about trading with hawk agents, it takes a shorter amount of time to eliminate hawks as a substantial threat from the population.

Experiment 1 showed us an notable dynamic that is present throughout evolutionary game theory literature and throughout this paper: the trade-off between individual success and group success (as in section 2.4.5).

The next experiment saw the addition of the false feedback and personal experience features. The intended purpose of introducing these two factors at once was to provide the counterpoint that use of personal experience by doves could be a defence against the false feedback attack vector. There were, however, a few problems with this theory; a more successful approach may have split experiments 2.2 and 2.3 into two separate experiments, with and without the personal experience factor. As it happens, we did not manage to successfully establish much about the personal experience factor other than through our mathematical analysis.

Despite this, the false feedback factor operated as expected - not being able to tell the true likelihood that the other agent will defect makes it more difficult to make informed decisions about whether to trade or not. This attack on a reputation system is known as “slandering” (section 2.6.3), and it can cause significant damage to a system if left unchecked - as we saw during our experiments. False feedback impacts the system both on a short-term and long-term basis.

The fourth and final experiment had us deal the idea of altruistic punishment - similarly to experiment 1, this is another factor where there is a trade-off between individual and group success, but more explicitly this time. When given the opportunity to punish unfair behaviour at their own expense, many people take that opportunity (section 2.5.3).

This experiment did not entirely go as planned as there were problems with obtaining positive experimental results due to a variety of issues. The main one being that the actual impact of the altruistic punishment option was small (in terms of cost and effect). This led to differences due to the A gene being indistinguishable from the other, more impactful, genes. Experiment 3.1 (section 3.5.5) explored some of these issues in detail and included some re-experiments that showed that certain solutions to these issues could be successful.

The mathematical analysis of the system suggested a trade-off like those previously discussed in Experiment 1, where use of the altruistic punishment function hurts individual agents but benefits the group when many of the agents participate in it. Experiment 3.2 partially suggested that this was the case, but further experiments being formulated for these factors would allow us to be more secure in this assessment.

The combination of experimentation and mathematical analysis allowed us to formulate and discuss hypotheses, while relating them back to their place in the online marketplace scenario. There are numerous possibilities for future work built off of this model and system using different sociological and psychological concepts and human behaviours (section 4.2).

4.1.4 Application to real-world problems

As previously discussed in section 4.1.1, there are a number of limitations of the model and system which have to be taken into account while discussing its application to a real-life online marketplace. Despite the required caveats, we believe that both the experiments contained within this paper and possible future experimentation using the model can have utility when designing a reputation system.

The first thing is the fact that even a basic reputation system does help the health of the system - see the difference between experiment 0 and the basic experiment 1. In the former no dove could ever score higher than a hawk (experiment 0.1), while in the latter it was possible for doves to eventually become dominant in the society (experiment 1.3). Even if a system provides only minimal information and the users have very unsophisticated strategies, the system will be less prone to fraud if you have that feedback system, and if it is used.

Section 2.6.3 briefly discusses different attacks on reputation systems, and we have discovered through our experiments that at least one of these - slander - has a great impact on the quality of the feedback information and on the health of the system. A reputation system must be designed with these attacks in mind lest the reputation system be rendered less effective. As previously discussed, the P2PTG does not include institutional factors such as banning or freezing accounts, and as such it is not possible to test this proposal with the P2PTG as it is. However, a good reputation system may monitor users for suspicious patterns of feedback that may suggest malicious intervention.

Large amounts of negative feedback - especially if it is against one particular user or group of users - may indicate a slander attack and should be dealt with accordingly. Methods such as requiring a phone number for verification may blunt the ability of malicious users to create new accounts and thus perform a whitewashing attack. Whitewashing, and a corresponding defence by users, is discussed a possible extension to the P2PTG in section 4.2.1. It may similarly be possible to implement an extension that deals with self-promotion attacks.

The altruistic punishment experiments, and corresponding social science literature, may provide a basis for a useful feature. As discussed in the conclusion to experiment 3 (section 3.5.6), there are considerations to be made about calibrating the cost and effect in such a way that makes users - who have already just been defrauded, and thus may not be so inclined to spend *more* money - actually want to use it while ensuring it has no use to malicious users. We know that altruistic punishment works (to some extent) to improve the health of the system in the P2PTG, and there are certain factors that are not modellable in the current P2PTG that could make this feature possible to deliver to a userbase: perhaps providing free tokens to punish someone after a certain number of transactions, or calculating the number of times a user has been punished by this method separately to their main feedback total.

It is possible to construct extensions to the P2PTG to simulate a wide variety of possible features, a non-exhaustive list of which will be covered in section 4.2. It would be possible to use the P2PTG, either through study by academics or experimentation by platform designers, as part of the process in producing modifications for an existing reputation system or in building a reputation system from the ground up.

4.2 Future Work

4.2.1 Further experimentation with the current P2PTG

There is scope for more experimentation within the currently implemented system as it stands as of experiment 3. More experimentation would allow firmer conclusions to be drawn about the impact of the 6 genes (S, P, R, E, F, A) that currently exist in the system with only limited implementation required.

Perhaps the most notable and pressing experiments to run would be related to the personal experience (E) gene; designing an experiment similar to experiment 2.1 but with four different values of F would allow us to conclude what impact the (infallible) trust versus (fallible) reputation information has on strategies (section 3.4.6). The possible interactions between A and E when attempting to counteract with false feedback could lead to noteworthy results, as well - perhaps one factor is more effective than the other, one causes the other to be irrelevant, or the system is healthier when both factors are included to some degree.

Reformulating experiments surrounding the A parameter to account for a more impactful (and more expensive) altruistic punishment option (as discussed in section 3.5.6) could allow a more thorough understanding of the gene and make it more relevant as part of the system. It is also possible that running some of experiment 3 with larger sample sizes could result in more positive results.

Further experimentation could be done by altering the payoff matrix. One way to make the P2PTG reflect more closely the online marketplace scenario would be to make being defrauded more expensive: in the current payoff matrix, the amount gained from one successful transaction is the same as the amount lost from one fraudulent one. This seems unlikely, as the gains from trade (section 2.2.2) are very unlikely to be worth the same as the value of the item itself. It is likely that we would find that more conservative doves (with higher values of R) are more successful, and it may well be that doves will be more inclined to use the altruistic punishment option (A) if the cost of being defrauded is higher.

Finally, there is scope to experiment with the existing breeding method (TOP_X - as discussed in section 3.1.2): in all these experiments, we used 30% as the value for TOP_X, but lowering or increasing that value is likely to significantly impact the end result. Lowering TOP_X - reducing the size of the breeding pool to only a few of the top agents - could result in faster congruence towards the "best" strategy, or in genetic stagnation, while increasing it could cause a wider range of strategies to be investigated, or for evolution to grind to a halt entirely. The experiments would have to be designed carefully to make the results meaningful and applicable to future extensions.

4.2.2 Extensions to the P2PTG

In addition to future work with the existing system, it would also be possible to extend the P2PTG system to include other factors. As discussed in section 4.1.2, the current P2PTG implementation is designed to make these extensions simple. Extensions to the P2PTG could be used to explore the potential effectiveness of a feature of a reputation system before implementing it, and/or as an academic exploration of a single or set of social behaviours.

Whitewashing attacks and a corresponding defence was included in the list of factors in section 2.8.6, but ultimately omitted from this paper due to time constraints. A whitewashing attack on a reputation system is an attack by which a malicious user deletes some or all of the negative feedback associated with them in order to further defraud people. The quintessential example in most systems is for that user to create a new account, which begins at zero feedback.

This is a frequent strategy employed by malicious users on online marketplaces and a topic worthy of exploration. While hawks would gain the ability to whitewash, the doves could also gain an additional defence - the ability to take into account the length of a user's reputation instead of merely its overall score. A human user is likely to view a user with 450 positive and 50 negative feedback more favourably than a user with 9 positive and 1 negative feedback. Even though they have the same percentage of positive feedback, the former has shown a more reliable pattern and invested more into fostering that positive feedback.

At the same time, if users would only trade with others with established reputations, it would be difficult for new agents to get started at all, which leads to some potentially significant emergent properties on a population scale. A possible addition on top of this would be to introduce new agents throughout the game - for instance, start with 500 agents and introduce a new agent every 100 transactions - to simulate the gradual intake of new users and to make it more difficult for doves to be able to tell the difference between hawks who have whitewashed and newly created dove agents.

One area that was necessarily neglected in this paper is the breeding and genetic algorithm side of the P2PTG: only one option was presented for breeding (the TOP_X method) and it was used in a constant way across all experiments. In this paper, this omission is for the purpose of simplicity and not having to deal with adding yet another conflating variable, but future experimentation could look into the impact of the parameters of the breeding side of the model on the results.

In natural animal populations, the next generation is not bred by a pool of the top performing agents[1] - the TOP_X method of breeding is actually more similar to human-led selective breeding in animals than modelling a wild population[27, Selective Breeding]. The breeding method could be changed to exist on a "sliding" scale, in which all agents are part of the breeding pool but the chance of each one being selected is proportional to their utility. As in real-world scenarios, the individuals with the most success usually breed more, but less successful organisms do also get to breed. Of course, this varies between animals and between societies - there having been observed difference in pack/herd/group structures in captivity versus the wild, for instance - but this sliding method could result in a more organic breeding process.

[1] In terms of evolutionary success. In our case, it refers to score/utility.

One thing that was overlooked in favour of properties emerging by chance is the idea of orchestration and inter-agent communication. This would be a quite radical alteration to the P2PTG but would possibly allow for an environment closer to the threat landscape of a real online marketplace platform at the expense of complexity and computational requirements. It could be possible to implement organisations of agents that communicate and cooperate for a common goal, either in defence or to orchestrate attacks. It would be interesting to experiment to see whether or not these organisations actually made a difference in the results as opposed to the emergent methods of attack/defence that we see in the experiments in this paper.

4.2.3 Other applications of the P2PTG

While the P2PTG has been designed to simulate a specific scenario - the online marketplace scenario - it is conceivable that it could be modified and adapted to suit other problems. Most game theory models, such as the Alarm Call and Hawk-Dove games, are used to simulate animal behaviour, and it is possible that the P2PTG could be modified to fit this purpose as well. Our review of the literature surrounding reputation is not exclusively about an online[2] system; but also a real-world social system - section 2.5.1 discusses reputation in context with other social pressures.

As the Hawk-Dove game is used to simulate a particular type of interaction (simple competition for resources in a one-versus-one scenario), the P2PTG could be used to model a set of more sophisticated interactions - one possible example is cooperation in a pack/herd context, where a pack may have to deal with uncooperative members and choosing whom to trust.

The P2PTG could also be used as part of a larger simulation of a human society, where it could deal with all sorts of cooperative activities. It may require some modifications to deal with the issue of reputation dissemination given you can not view someone's reputation when looking at an organic person, and of temporality, as any useful model of a human society is likely to take into account time rather than just a number of different transactions. Offline societies have different threat models when it comes to attacks on their reputation system due to the medium - whitewashing attacks are harder, for instance - but modifying the model to account for this should be fairly simple.

[2] Not that "online" and "society" are mutually exclusive, but the requirements of an online system are different to that of used in other societies.

Bibliography

[1] D. G. Gregg and J. E. Scott, “The role of reputation systems in reducing on-line auction fraud,” *International Journal of Electronic Commerce*, vol. 10, no. 3, pp. 95–120, 2006.

[2] ——, “A typology of complaints about ebay sellers,” *Communications of the ACM*, vol. 51, no. 4, pp. 69–74, 2008.

[3] J. Engelberg and J. Williams, “Ebay’s proxy bidding: a license to shill,” *Journal of Economic Behavior & Organization*, vol. 72, no. 1, pp. 509–526, 2009.

[4] P. Krugman, *Economics*, 2nd. Worth Publishers, 2009.

[5] K. Sigmund and M. A. Nowak, “Evolutionary game theory,” *Current Biology*, vol. 9, no. 14, R503–R505, 1999.

[6] S. P. H. Heap and Y. Varoufakis, *Game Theory: A Critical Introduction*. Routledge.

[7] E. Rasmusen, *Games and Information*. Blackwell, 1994.

[8] R. Axelrod and W. D. Hamilton, “The evolution of cooperation,” *Science*, vol. 211, no. 4489, pp. 1390–1396, 1981.

[9] E. A. Stanley, D. Ashlock, and L. Tesfatsion, “Iterated prisoner’s dilemma with choice and refusal of partners,” Tech. Rep., 1994.

[10] A. J. Stewart and J. B. Plotkin, “From extortion to generosity, evolution in the iterated prisoner’s dilemma,” *Proceedings of the National Academy of Sciences*, vol. 110, no. 38, pp. 15 348–15 353, 2013.

[11] A. Rapoport and A. M. Chammah, “The game of chicken,” *American Behavioral Scientist*, vol. 10, no. 3, pp. 10–28, 1966.

[12] J. M. Smith and G. Price, “The logic of animal conflict,” *Nature*, vol. 246, p. 15, 1973.

[13] J. M. Smith, “Optimization theory in evolution,” *Annual Review of Ecology and Systematics*, pp. 31–56, 1978.

[14] ——, “The evolution of alarm calls,” *American Naturalist*, pp. 59–63, 1965.

[15] E. L. Charnov and J. R. Krebs, “The evolution of alarm calls: altruism or manipulation?” English, *The American Naturalist*, vol. 109, no. 965, pages, 1975, ISSN: 00030147. [Online]. Available: `http://www.jstor.org/stable/2459642`.

[16] M. Rabin, “Incorporating fairness into game theory and economics,” *The American economic review*, pp. 1281–1302, 1993.

[17] B. Schneier, *Liars and Outliers: Enable the Trust that Society Needs to Thrive*. Wiley, 2012.

[18] E. Fehr and U. Fischbacher, “The nature of human altruism,” *Nature*, vol. 425, no. 6960, pp. 785–791, 2003.

[19] E. Fehr, U. Fischbacher, and S. Gächter, “Strong reciprocity, human cooperation, and the enforcement of social norms,” *Human nature*, vol. 13, no. 1, pp. 1–25, 2002.

[20] G. Roberts, "Evolution of direct and indirect reciprocity," *Proceedings of the Royal Society B: Biological Sciences*, vol. 275, no. 1631, pp. 173–179, 2008.

[21] R. Forsythe, J. L. Horowitz, N. E. Savin, and M. Sefton, "Fairness in simple bargaining experiments," *Games and Economic behavior*, vol. 6, no. 3, pp. 347–369, 1994.

[22] L. Mui, M. Mohtashemi, and A. Halberstadt, "A computational model of trust and reputation," in *System Sciences, 2002. HICSS. Proceedings of the 35th Annual Hawaii International Conference on*, IEEE, 2002, pp. 2431–2439.

[23] C. Stiff, "Are they bothered? how the opportunity to damage a partner's reputation influences giving behavior in a trust game," *The Journal of social psychology*, vol. 148, no. 5, pp. 609–630, 2008.

[24] K. Hoffman, D. Zage, and C. Nita-Rotaru, "A survey of attack and defense techniques for reputation systems," *ACM Computing Surveys (CSUR)*, vol. 42, no. 1, p. 1, 2009.

[25] M. Mitchell, *An Introduction to Genetic Algorithms*, 2nd. A Bradford Book, 1996.

[26] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning.* Addison-Wesley, 1989.

[27] D. Mcfarland, *A Dictionary of Animal Behaviour*, ser. Oxford Paperback Reference. OUP Oxford, 2006, ISBN: 9780198607212. [Online]. Available: `http://books.google.co.uk/books?id=0wh9HqG-xLsC`.

Appendix

A.1 Appendix: Experiment 0

A.1.1 Experiment 0.2

The following is a table of the proportion of the population that is hawk (H) after five generations of experiment 0.2. The mean values are provided in the main document.

-	I_1	I_2	I_3	I_4	I_5
H_1	0.96	0.968	0.948	0.624	0.58
H_2	0.956	0.96	0.956	0.644	0.66
H_3	0.952	0.904	0.952	0.612	0.568
H_4	0.956	0.92	0.952	0.564	0.548
H_5	0.952	0.932	0.948	0.632	0.604
H_6	0.94	0.948	0.944	0.636	0.608
H_7	0.956	0.964	0.968	0.616	0.588
H_8	0.944	0.952	0.948	0.632	0.572
H_9	0.968	0.944	0.948	0.592	0.608
H_{10}	0.972	0.944	0.964	0.564	0.612
H_{mean}	0.956	0.944	0.953	0.612	0.5948

A.1.2 Experiment 0.3

One of six of these experiments are provided in the main body, the rest of the results for experiment 0.3 are presented here. The graphs corresponding to these tables are available in figure A.1.

The following is the table of the average dove score's standard deviation (D_{SD}).

T / N	10	25	50	100	250
10	1.223	0.932	0.643	0.387	0.265
25	1.391	0.726	0.539	0.5	0.281
50	1.403	0.88	0.561	0.412	0.229
100	1.327	0.84	0.668	0.438	0.278
250	1.259	0.805	0.65	0.473	0.277

The following is the table of the average hawk score's standard deviation (H_{SD}).

T / N	10	25	50	100	250
10	1.63	1.172	0.756	0.491	0.327
25	1.643	0.85	0.7	0.615	0.366
50	1.543	1.115	0.706	0.523	0.291
100	1.444	1.071	0.836	0.548	0.357
250	1.582	1.034	0.815	0.592	0.342

The following is the table of the percentage of hawks in the population's standard deviation (P_{SD}).

T / N	10	25	50	100	250
10	32.592	20.881	14.904	9.559	6.421
25	34.293	17.302	13.298	12.123	7.102
50	33.512	21.106	13.697	10.26	5.741
100	30.43	20.15	16.363	10.865	7.017
250	31.509	19.518	15.884	11.696	6.847

The following is the table of the average dove score's interquartile range (D_{IQR}).

T / N	10	25	50	100	250
10	2.231	1.65	0.835	0.507	0.414
25	2.656	0.986	0.808	0.759	0.35
50	2.676	1.083	0.611	0.569	0.279
100	2.238	1.282	0.823	0.478	0.362
250	2.495	1.013	0.993	0.715	0.354

The following is the table of the average hawk score's interquartile range (H_{IQR}).

T / N	10	25	50	100	250
10	2.451	1.384	0.847	0.641	0.487
25	2.94	1.235	1.019	0.883	0.478
50	2.84	1.497	0.903	0.783	0.389
100	2.334	1.63	0.961	0.553	0.411
250	2.427	1.39	1.248	0.869	0.445

The following is the table of the percentage of hawks in the population's interquartile range (P_{IQR}).

T / N	10	25	50	100	250
10	57.5	32	18.5	12.75	10.5
25	70	24	18	17	8.5
50	60	24	16	14.75	6.7
100	50	32	19	11.75	8.7
250	57.5	24	24	17	8.8

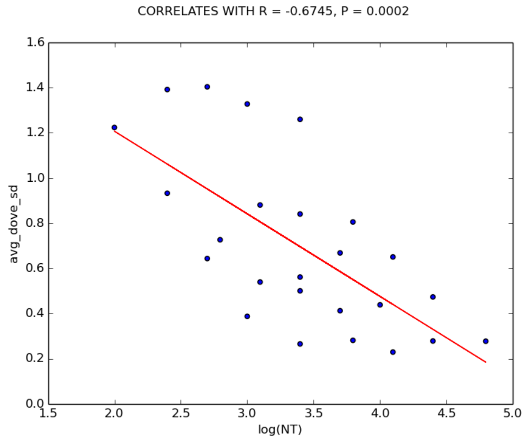
A.2 Appendix: Experiment 1

A.2.1 Experiment 1.3

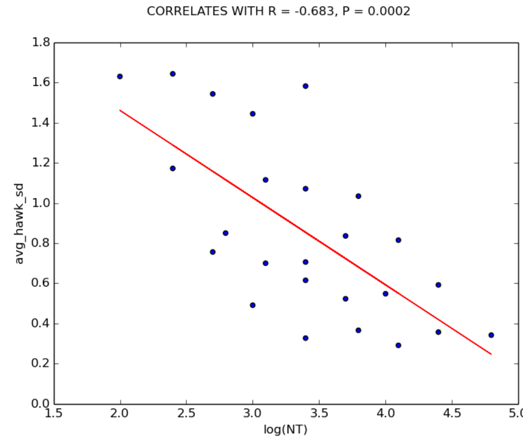
These are the results for the abandoned first version of experiment 1.3. For each I, the experiment is repeated 10 times:

Figure A.1: Correlations for H0.3.

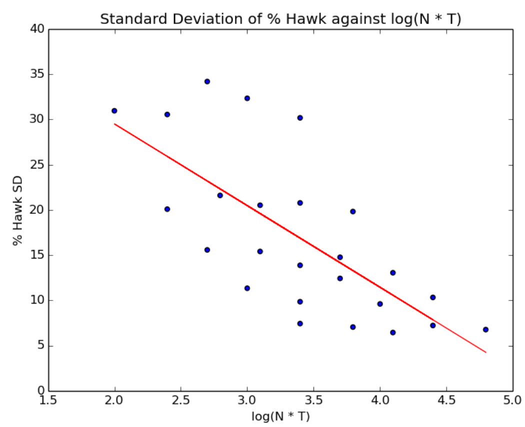
(a) Standard deviation of average dove score.



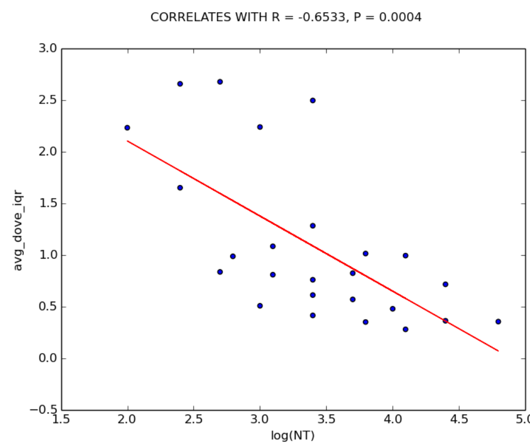
(b) Standard deviation of average hawk score.



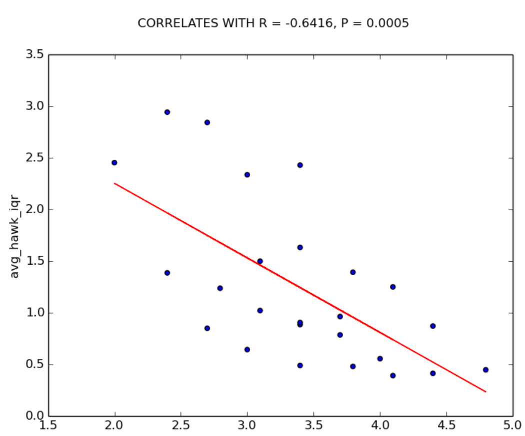
(c) Standard deviation of percentage of hawks in the population.



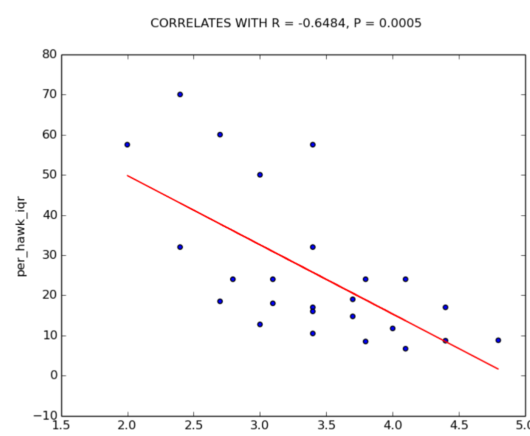
(d) Interquartile range of average dove score.



(e) Interquartile range of average hawk score.



(f) Interquartile range of percentage of hawks in the population.



-	Average H	SD
I_c	12.3	1.87
I_1	11.2	1.54
I_2	12.5	1.36
I_3	5.6	6.93

Performing a single-factor ANOVA test: $p = 0.00073$, $p < 0.05$.

T-tests ($p < 0.0125$):

- $H(I_3) < H(I_c)$ - $p = 0.018$ - $p < 0.0125$
- $H(I_3) < H(I_1)$ - $p = 0.040$ - $p < 0.0125$
- $H(I_3) < H(I_2)$ - $p = 0.016$ - $p < 0.0125$
- $H(I_2) < H(I_1)$ - $p = 0.074$ - $p < 0.0125$

A.3 Appendix: Experiment 2

A.3.1 Experiment 2.1

The graphs of S (average score per transaction) against E for each value of $T : N$ are available in figure A.3. It's important to note that each graph is the composite of 10 experiments, with 250 agents each (of which 60% will be doves) - thus there should be approximately 1500 data points per graph.

A.3.2 Experiment 2.1 - Informal Experiment

These are the results for the informal re-attempt of experiment 2.1 as described in section 3.4.6. In this case, $T : N = 10$, but the value of E varies between the two test cases.

-	Mean score (SD)	Average SD score	Outliers[3] (SD)
$E = 0$	) 1.53	0.15	11.5 (9.55)
$E = Uniform(1, 1023)$	1.57 (0.10)	0.30	9.50 (6.02)

A.4 Appendix: Experiment 3

A.4.1 Experiment 3.1 - No F

This is a re-do of Experiment 3.1 that sets the F parameter to 0. The reasoning and discussion of this experiment is in section 3.5.5.

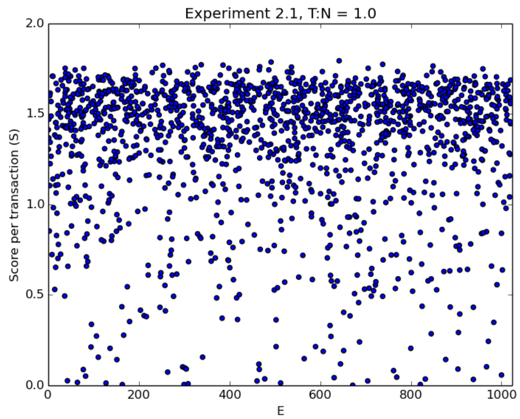
Table A.4: Experiment 3.1 - Linear regression of the correlation between A and dove scores - No F re-experiment

-	$A - DCorrelation$
m	$-6.2 * 10^{-5}$
c	1.63
PMCC	-0.304

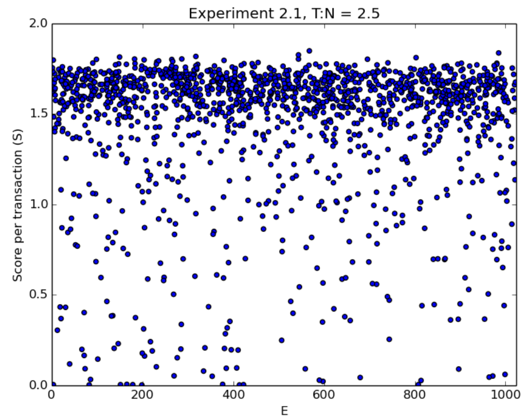
[3] Number of agents where S ¡ 1

Figure A.3: Experiment 2.1 graphs.

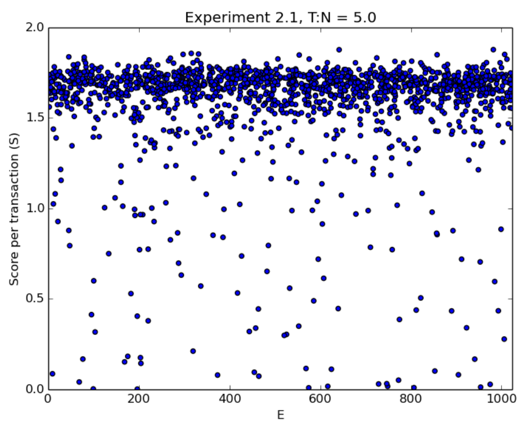
(a) $T : N = 1$



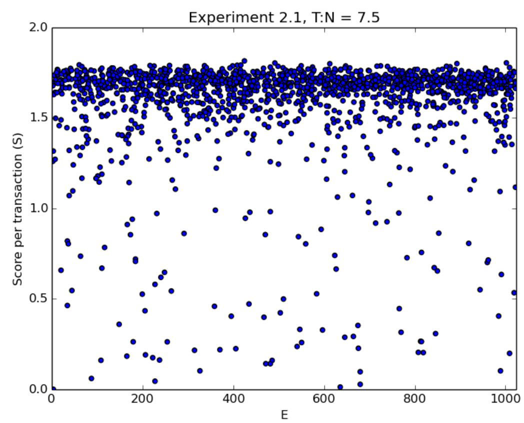
(b) $T : N = 2.5$



(c) $T : N = 5$



(d) $T : N = 7.5$



(e) $T : N = 10$

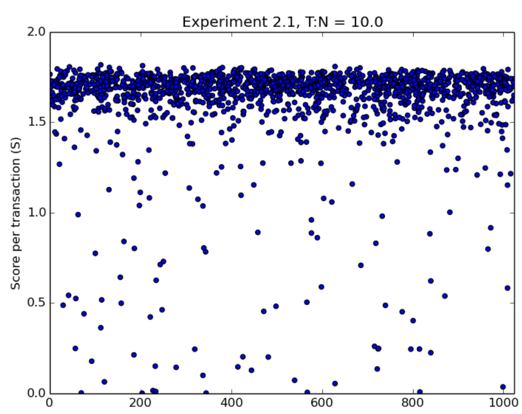
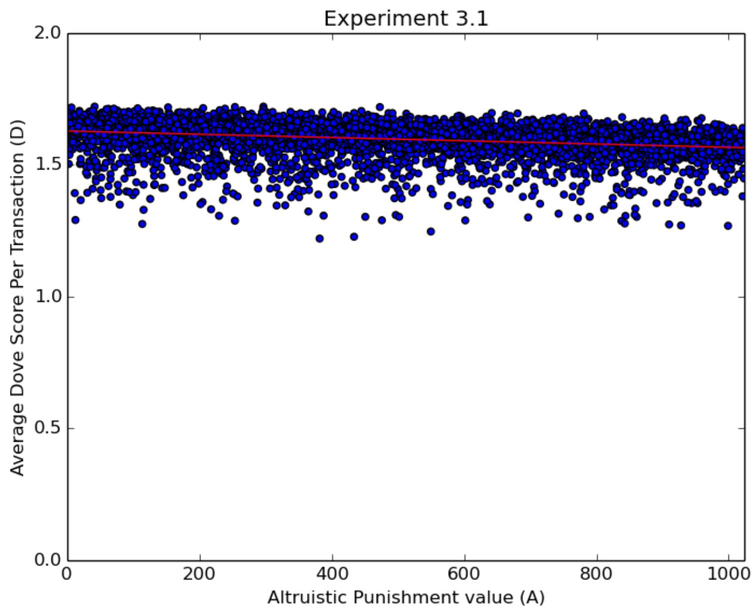


Figure A.4: Experiment 3.1 Results - Graph - No F re-experiment



A.4.2 Experiment 3.1 - reduced variance P and R

This is a re-do of experiment 3.1 that sets the P and R parameters to $N(300, 50)$ and $N(500, 50)$ respectively (where N is a normal probability distribution). The reasoning and discussion of this experiment is in section 3.5.5.

Table A.4: Experiment 3.1 - Linear regression of the correlation between A and dove scores - Reduced variance re-experiment

-	$A - DCorrelation$
m	$-7.86 * 10^{-5}$
c	1.67
PMCC	-0.72

Figure A.4: Experiment 3.1 Results - Graph - Reduced variance re-experiment