Mean Field Games in Asset Price Coordination

Andrei Dranka

Student Number: 999766132
Supervisor: Robert McCann
April, 2017

Abstract

The purpose of this thesis is to model asset price dynamics when considering the effects of herd behaviour and trend following. This is done using mean field games which attempts to capture the interactions between market participants. This is achieved by providing a reference trajectory for all the market participants, as well as penalizing market participants who stray too far from the average price seen by all participants. The model is tested on data from an asset price coordination experiment performed in 2004 by Hommes et al. [1]. The model provides insights into how the importance of predicting prices close to the average price affects the average price path.

Acknowledgments

I would first like to thank my thesis supervisor, professor Robert McCann of the Department of Mathematics at the University of Toronto. Without his knowledge, wisdom and guidance this thesis would not be possible. I would like to thank him for supporting me and my ideas, and for leading me down the right path in my research.

I would like to thank Bill (Xuancheng) Huang of the Department of Statistics at the University of Toronto for his support and help in this research. I am enormously indebted to Bill's feedback and cooperation. Without many of Bill's suggestions throughout every step of the process, this document would have never been accomplished.

I would also like to thank Dana Wegierak, an up-and-coming physicist at Ryerson University, who showed unwavering support in my times of need. Thank you for reading, listening, and giving feedback to all my work. It added a level of polish that I could not achieve on my own.

Finally, I must express my most profound gratitude to my parents and my mom, dad, and sister Oksana for providing me with unfailing support and continuous encouragement throughout the years. I would not be where I am now without them.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

An important topic in behavioural finance is the study of trend following. Trend following occurs when market participants predict future expected prices using recent historical prices and price changes. According to Hommes et al. [1], trend following is one of the main driving forces towards coordination of expectations in asset pricing experiments. Understanding the dynamics of these simple asset pricing experiments is crucial to fully understanding how trend following affects more complicated systems.

A possible explanation of trend following is herd behaviour. Herd behaviour describes how individuals in a group can act collectively without centralized direction. Individuals herd when knowledge of what others are investing changes their decisions [2]. Essentially, the investors copy what others are doing.

One way to model this behaviour is to use agent based modeling with game theory. Game theory is the study of mathematical models of interactions of intelligent rational decision makers. When working with multi-agent systems, game theory models can become very complex and difficult to solve. Mean field game (MFG) theory, a class of problems discovered in 2006, models a large number of interacting individuals by only considering the interactions of a single individual with the 'mean field' (an average of the population). This greatly reduces the complexity of

the system and allows for easy computation of solutions. For this reason, this thesis will use mean field game theory to model the coordination of expectations in asset pricing experiments.

The significance of this research is that it will show that when participants in simple asset price prediction experiments coordinate on a price, the price path is affected. The model will also assume that the particpants are acting rationally by fulfilling an approximate Nash equilibrium (known as an ϵ-Nash equilibrium, i.e. no agent can improve their objective by greater than ϵ without changing their strategy), despite predicting prices that vary considerably from the fundamental price. As the number of participants approaches infinity ($N \to \infty$), the error ϵ, will approach zero ($\epsilon \to 0$) and the ϵ-Nash equilibrium becomes a regular Nash equilibrium. This means that no participant or agent can improve their objective without changing their strategy. A rational agent is defined as an agent that has clear preferences, models uncertain outcomes with expectations of their utility function (from expected utility theory [3]), and always performs actions with the optimal expected outcome in mind.

1.2 Objectives

The objective of this thesis is to model asset price evolution. This will be done by modeling coordination of asset prices using mean field game theory. Specifically, the thesis will study how the price path evolves according to different model parameters. Connections with other techniques used to model behavioural finance will also be included.

1.3 Approach and methods

To achieve this goal, this research will use the model described in *Mean Field LQG Games in Leader-Follower Stochastic Multi-Agent Systems: Likelihood Ratio Based Adaptation* [4]. Once the model has been developed, a verification and validation

process will be completed. Proofs for all theoretical results will be formulated and discussed, as well as the implications of certain formulations. This model will be tested using data from *Coordination of Expectations in Asset Pricing Experiments* [1], provided by Hommes et al. In this article, individuals predict the next period price and the price is then calculated based on the individual's predictions. This is necessary to make sure that this model is not only mathematically and logically sound, but can actually provide useful results.

The rest of this report will be organized as follows: in section 2, a literature review will be discussed. The literature review will provide further background knowledge into the subject matters as well as summarize current research and findings. In section 3, the mean field game (MFG) model will be set up. In section 4, the results will be shown as well as discussed. In section 5, the discussion about the MFG model will be expanded on and possible future work will be outlined.

Chapter 2

Literature Review

The literature review will be separated into three sections: behavioural finance research, game theory research, and optimal stochastic control research. Each section will introduce the relevant research in a top-down approach, i.e. concepts will be drilled down from the most broad topic, to the most specific and needed in the thesis.

2.1 Behavioural Finance

2.1.1 Background

A central issue in behavioural finance is explaining why market participants (such as real investors) make systematic, irrational errors [5]. It has been observed that market participants (e.g. in stock markets) do not adhere to the efficient market hypothesis (EMH), which states that an asset price fully reflects all available information [6]. This would imply that these market participants are not rational as they make irrational systematic errors. It has also been shown that there are significant autocorrelations in stock returns [7] [8], further invalidating EMH.

The study of behavioural finance attempts to explain many phenomena that are observed in markets. This report will focus on two such phenomena: herd behaviour and trend following. Herd behaviour occurs when market participants

act collectively without a centralized direction [9]. It occurs when investors change their investment decisions when they have knowledge of what are others are doing [2]. Herding behaviour is often used to partially explain asset price bubbles and crashes [10] and has been identified in the collective irrationality of investors [11]. Trend following is an investment strategy that aims to take advantage of recent changes in asset prices. Simply put, prices that have recently moved upwards tend to continue to increase and prices that have recently decreased will continue to do so [12]. Trend following strategies have also been shown to perform well with strong positive returns and a low realized correlation [13]. There is evidence that the herding behaviour prolongs trends as market participants often join the bandwagon [14]. Trends often start from the self-confirming expectation feedback mechanism in the market [1].

Another important concept in behavioural finance is the Keynesian beauty contest [15]. The idea is that individuals will price shares in the market by what they believe everyone else thinks the price is, not by the fundamental value. Allen et al. [16] formalized this concept and concluded the following two observations from their research. First, the average price path will diverge from the market consensus of the expected fundamental value of the asset. Second, prices from beauty contests react more sluggishly to changes in the fundamental value of the asset (exhibiting a form of inertia).

In order to study these phenomena, controlled experiments must be set up. This is necessary because certain parameters (such as fundamental stock price, available liquidity, etc.) must be controlled for. This allows for powerful tests on the robustness of certain theories, which may not be possible with field data because of the existence of many hidden parameters.

2.1.2 Coordination of Asset Prices

An asset pricing experiment was conducted by Hommes et al. and published in an article called *Coordination of Expectations in Asset Pricing Experiments* [1]. In the research, Hommes et al. created a simple asset pricing experiment: a one asset mar-

ket was created and market participants were asked to predict the next period asset price. Participants act as advisors to a pension fund, the higher their prediction, the greater the demand for the asset by the pension fund. Market participants are given their own past predictions, as well as the past prices of the asset determined during the experiment. Participants are aware that the asset price is determined by market equilibrium, but they do not know the underlying market equilibrium equation. Participants are also not aware of the investment strategy of the pension funds, the number of pension funds, and the identities of the other members of the group. Participant earnings are inversely proportional to the prediction error, which is known.

The market clearing price at period t in the experiment was determined by a risk-free discounted weighted combination of the asset price predictions and the fundamental price.

The results showed that market participants seemed to coordinate on a price path. Price predictions do not deviate largely period to period compared to the average price predicted. The market clearing price also has specific patterns and trends, such as oscillating movements, exponentially decreasing oscillating movements, and almost monotonic convergence towards the fundamental asset price. This experiment is very similar in nature to the Keynesian beauty contest because participants are successful in predicting what the future average price will be.

Hommes et al. model market participants using simple rules-based, linear forecasting strategies. These forecasting strategies are linear recurrence relations with constant coefficients. In particular, recurrence relations of degree-2, 3 and 4 are highlighted. These relations are shown to model individual forecasting strategies very well in the experiments. A degree-2 recurrence relation can have the following form

$$p^e_{h,t+1} = \alpha_h + \beta_h p_{t-1} + \delta_h (p_{t-1} - p_{t-2}) \tag{2.1}$$

where $p^e_{h,t+1}$ is the predicted next period price, p_t is the actual asset price at time t, and $\alpha_h, \beta_h, \delta_h \in \mathbb{R}$ are the agent parameters.

As explained in [1], this equation has a nice behavioural interpretation. The first two terms on the right hand side show that the expected future price depends on the last price observed. The third term shows that agents will also use the *trend* in the their predictions. If $\delta > 0$, a positive feedback occurs and individuals are considered *trend extrapolators*. If $\delta < 0$, the agent is considered a *contrarian*, predicting in the opposite direction of the trend.

2.1.3 Other Asset Price Prediction Models

In *Asset Pricing Under Endogenous Expectations in an Artificial Stock Market* [17], a theory of asset pricing is proposed. In it, heterogenous agents continually adapt their expectations to the the market, which the market aggressively creates. Agents therefore continuously make expectational models which they update depending on which models are the most successful. This model is very similar to the model proposed by Hommes et al. because individuals create their prediction strategies based on price changes, specifically moving averages. Strategies of individuals also change to the most optimal strategies, where most optimal is defined as the best predicting strategy for the next period price (thus suggesting a Keynesian beauty contest). This model differs from Hommes et al. as it incorporates the fundamental price into the individuals' price predictions. The model also is intended to be used in a more general setting (predicting and acting on prices), whereas the previous model is more focused on just price predictions (not necessarily acting upon them).

When modeling Keynesian beauty contests, it may be beneficial to determine whether or not the model satisfies a Nash equilibrium. This is shown in *Information aggregation in a beauty contest game*[18], where players try to approach a certain state, while gathering information from other agents. Nash equilibriums and their relevance to economics will be looked at more closely in the next section.

2.2 Game Theory

2.2.1 Background

Game theory is the study of mathematical models of interacting intelligent rational decision makers [19]. In this context a game is a system which players engage in with rules, strategies and outcomes that are defined by clear mathematical parameters. Games can be divided into two groups: cooperative and non-cooperative games [20]. A non-cooperative game is one in which only self-enforcing agreements are possible [21]. A self-enforcing agreement is one in which an alliance is made between two individuals and is enforced by only those two individuals, i.e. no third party can interfere. This means that alliances and coalitions cannot be formed because the agreement is between more than two players. Cooperative games are ones in which groups of players can form alliances and coalitions due to the possibility of external enforcement cooperative behaviour. Coordination games are a special subset of cooperative games where players choose the same or corresponding strategies [22].

An important idea in game theory is the Nash equilibrium. A Nash equilibrium is defined as a solution concept, a formal set of rules for predicting how a certain game will be played. Here, all players know each other's equilibrium strategy and no player has anything to gain by changing their own strategy, if all others remain constant. If agents act such that their actions can be predicted with a Nash equilibrium, it provides strong evidence for agents acting rationally. This is because every Nash equilibrium is also a rationalizable equilibrium [23]. A rationalizable equilibrium is a solution concept that provides the weakest constraints on agents while maintaining that agents are rational and this rationality is common knowledge to every agent.

A rational agent has three main characteristics: they have clear preferences, they model uncertainty with expectations of their utility function (from expected utility theory [3]), and they always perform actions with the optimal expected outcome. A Nash equilibrium satisfies the third characteristic; agents will make decisions to optimize their expected outcome, however it may be defined.

2.2.2 Multi-agent Systems

For many behavioural economic models, such as the coordination of asset prices, it is appropriate to define a multi-agent system. This model must consist of rational, intelligent agents, therefore satisfying the three characteristics from section 2.2.1. In multi-agent systems, the agents' decisions depend on their own actions, how their actions influence others' actions, and how others' actions influence their own. Modeling multi-agent systems in this way can be very difficult. This is because it can be challenging to verify and confirm whether or not a certain model satisfies a Nash equilibrium. The number of interactions increases exponentially as the number of agents increases, further adding to the complexity. Analytical solutions are difficult to find. Oftentimes, only computer simulations can solve such problems.

2.2.3 Mean Field Game Theory

In 2006 a new field of game theory was developed independently by two groups which has since been titled mean field game theory. Each group took different approaches to the development of mean field game theory, but the two are related and have a direct correspondence. Mean field game theory is used to study strategic decision making between very large populations of agents. The central idea behind mean field game theory is that each interacting agent alone is insignificant compared to the very large population, but are significant when aggregated together. This creates only two relationships: interactions between an agent and a population, and how the population changes considering these interactions.

Work by Lasry and Lions

From Lasry and Lions [24], these relationships can be described with a coupled system of equations, one evolving forward in time and one evolving backward in time. The backward in time equation is a Hamilton-Jacobi-Bellman (HJB) equation. It describes the interactions between an individual agent and the population distribution.

The HJB equation is a partial differential equation that is central to optimal control theory. Given a cost function, the solution of the HJB is a value function which gives the minimum cost for a defined dynamical system. It can be thought of as the integral of Newton's second law. In mean field game theory, the HJB equation is used to represent the agent's optimal decisions based on where they wish to be in the future. It is stated as such

$$\frac{\partial J}{\partial t} - \Delta J + H(x, \nabla J) = V(x, m), \quad J|_{t=0} = V_0(x, m(x, 0)) \tag{2.2}$$

where J is a scalar function associated with the cost, x is the state of the agent, $H(x, p)$ is a given convex function (typically a Legendre transform). $m > 0, \int m dx = 1, \forall t \in [0, T]$ in this context is the mass of the population, a probability distribution of the states of all the agents. $V(x, m)$ is the marginal cost for being at state x and mass distribution m. Precise definitions are given in [24].

The forward in time equation is the Fokker-Planck (FP) equation. The FP equation is a partial differential equation that describes the time evolution of a probability density function of the "velocity" of a particle while under drag and other random forces. It is a kind of continuity equation. In mean field game theory, this equation represents where the agents actually end up based on their initial position. This can be summarized with the following equation

$$\frac{\partial m}{\partial t} + \Delta m + \nabla \cdot (\frac{\partial H}{\partial p}(x, \nabla u) \cdot m) = 0, \quad m|_{t=T} = m_0 \tag{2.3}$$

Solving these two equations together is highly non-trivial. In many cases, existence and uniqueness break down, suggesting the mean field approximation of the system is not even a valid approach [24]. This thesis will focus specifically on the cases where existence and uniqueness do exist, which offers some validity to the approximation. An excellent summary of this approach of mean field games can be found from Cardaliaguet [25].

Work from Huang et al.

From Huang et al. came the second approach to mean field game theory [26]. This approach considers the problem from an optimal stochastic control perspective. This paper defines the Nash-Certainty Equivalence (NCE) Methodology, a technique used to solve mean field games. This technique requires that the problem be set up using a controlled stochastic process of the McKean-Vlasov type (coefficients of process depend on distribution of agents) and a corresponding control performance. The McKean-Vlasov process described has the following form for an n-agent dynamic game

$$dx_t = f[x_t, u_t, \mu_t^1, \cdots, \mu_t^K]dt + \sigma dw_t \tag{2.4}$$

where x_t is the agent's state at time t, $f(\cdot)$ is the drift coefficient, u_t is the input control at time t, and $\mu_t^i, 1 \leq i \leq K \leq n$ is the probability distribution of agent i at time t. K agents are used as an approximation of the n agents, which should be sufficient if K is large enough and $n \to \infty$. This equation is coupled to the following cost function

$$J(u, \mu_t^1, \cdots, \mu_t^K) := \mathbb{E}(\int_0^T L[x_t, u_t, \mu_t^1, \cdots, \mu_t^K]dt) \tag{2.5}$$

where $L(\cdot)$ is a nonlinear function that maps to $\mathbb{R}_+ = [0, \infty)$. These two approaches are in essence explaining the same concept. The first method from Lasry and Lions is more general, it allows a greater number of problems to be modeled. However, this generality comes at the price of making the problems very difficult to solve analytically and even numerically. On the other hand, the second approach from [26] provides an easier method to solve both analytically and numerically mean field games, at the cost of restricting the number of problems that can be modeled. For this reason, the second method will be the primary technique that will be used in the present study.

An extension of [26] was done and titled *Mean Field LQG Control in Leader-Follower Stochastic Multi-Agent Systems: Likelihood Ratio Based Adaptation* [4].

Many of the results applied in this paper were also discovered by Li and Zhang in [27]. The initial model in [26] is now extended to include multi-dimensional states for each agent, two kinds of agents (leaders and followers), as well as define a so-called adaptive follower that can change trajectories depending on the mean field path. This thesis will use the results from [4] extensively.

2.3 Optimal Stochastic Control

Optimal stochastic control theory is widely in used in the finance literature [28]. Optimal stochastic control theory is concerned with minimizing the cost of a dynamical stochastic system. The focus in this thesis will be on a Linear-Quadratic-Gaussian (LQG) regulator. An LQG regulator is a feedback controller with a continuous-time linear stochastic system, with a quadratic cost function. It can be described with two equations [29]. The first is the Itô process:

$$dz = (az + bu)dt + cdW$$

where z is the system state, u is the control input, W is the standard Wiener process, and a, b, and c are all constants.

The second equation is the quadratic cost function defined as:

$$J(z(t), u(t)) = \int_{t_0}^{t_1} qz(t)^2 + ru(t)^2 dt + fz(t)^2$$

where q, r, and f are all constants.

Chapter 3

Setting Up the MFG Model

3.1 Problem Formulation

The goal of this thesis is to model rational, intelligent agents that coordinate on asset prices, as found by [1]. To do this, we model the system using mean field (MF) linear-quadratic-Gaussian (LQG) stochastic control theory. As we will see, the mean field comes from coupling between the stochastic dynamics of each agent and the cost function. The linear-quadratic term appears because the cost function is a linear-quadratic regulator, and the term Gaussian is present because we will use the Wiener process, which has a Gaussian distribution.

The stochastic dynamics of each of the n agents will given by the following Itô process:

$$dz_i = (a_i z_i + b_i u_i)dt + c_i dW_i \tag{3.1}$$

where $\{z_i \in \mathbb{R} : 1 \leq i \leq n\}$ is the state of agent i, $\{u_i \in \mathbb{R} : 1 \leq i \leq n\}$ is the control input for agent i, and $\{W_i : 1 \leq i \leq n\}$ is the standard Wiener process for agent i. The terms a_i, b_i and $c_i \in \mathbb{R}$ are constant parameters defined for each agent i. The initial states of each agent are given as $\{z_i(0) \in \mathbb{R} : 1 \leq i \leq n\}$ which are assumed to be independent of each other and independent of W_i.

The control input must have certain bounds and restrictions in order to make

intuitive sense. For example, control input should not have any information about future random processes, hence they should be non-anticipating. Restrictions will be applied using an admissible control set. This control set will be defined as such: $\mathcal{U}_i := \{u_i : u_i(t)$ is adapted to sigma-field $\sigma(z_j(s), s \leq t, 1 \leq j \leq n), \|z_i(T)\| = o(\sqrt{T}), \int_0^T \|z_i(t)\|^2 dt = O(T)(a.s.)\}$. These conditions ensure that the control input is both feasible and realistic.

The formulation of the control input of each agent will entirely depend on the cost function that is chosen. For the cost function, we would like the agents to coordinate towards some defined path, however, we also want some penalty towards changing an agent's strategy. This path must depend on what state each agent is currently in, as well as future states the agents can traverse. The cost function should also be indifferent as to whether or not the agent is approaching from above or below. The model should assume that there is no terminal time for the game, thus prompting the model to be a long-run average cost function. The following cost function will be used

$$J_i(u_i; u_{-i}) := \limsup_{T\to\infty} \frac{1}{T} \int_0^T q(z_i(t) - \phi(t, z_i, z_{-i}))^2 + r u_i(t)^2 dt \tag{3.2}$$

where $J_i(\cdot)$ is the cost function we wish to minimize, $q > 0$ is the weight given to keep the agents close to ϕ, the trajectory function, $r > 0$ is the weight given to cost of increasing the control in magnitude, and ϕ is the trajectory function the agents try to approach. For notational purposes, u_{-i} and z_{-i} will be defined as such: $u_{-i} := (u_1, \cdots, u_{i-1}, u_{i+1}, \cdots, u_n)$ and $z_{-i} := (z_1, \cdots, z_{i-1}, z_{i+1}, \cdots, z_n)$. The trajectory function will be given the following convex combination form

$$\phi(t, z_i, z_{-i}) := \lambda h(t) + (1 - \lambda)\psi(t, z_i, z_{-i}) \tag{3.3}$$

where $\lambda \in (0, 1)$ is a scalar, $h(\cdot) \in C^b$, where $C^b := \{f \in C : sup_{t\geq 0}\|f(t)\| < \infty\}$, C is the family of all continuous functions. $h(\cdot)$ is the reference function that all agents are aware of and each agent tries to follow with weight λ. $\psi(t, z_i, z_{-i}) := \sum_{i=1}^n \bar{z}_i(t)/n$ is the centroid of all the agents. This term couples the cost function

to the dynamics equation for each agent.

For this thesis, a simplification will be made. a_i and b_i will be the same value for each agent, denoted by a and b. This constraint may be relaxed in future work.

3.2 Mean Field Game Approach using Nash Certainty Equivalence Principle

According to [27], the following theorem is known (simplified for scalars):

Theorem 1 *For the LQG optimal control problem (3.1) - (3.2), assume (i)* $[a, b]$ *is stabilizable, (ii)* $[a, \sqrt{q}]$ *is detectable, and (iii)* $\phi(\cdot) \in C^b$*. Then we have:*

(a) The algebraic Riccati equation $2aP - \frac{b^2}{r}P^2 + q = 0$ *has a unique solution* P

(b) $\gamma := a - \frac{b^2}{r}P$ *is asymptotically stable*

(c) The differential equation $ds/dt = -\gamma s + q\phi$ *has a unique solution in* C^b*:*

$$s(t) = -\int_t^\infty e^{\gamma(\tau - t)} q\phi(\tau) d\tau \tag{3.4}$$

(d) The optimal control law: $u^o(\cdot) := \arg\inf_{u(\cdot)\in\mathcal{U}} J(u(\cdot)) = -\frac{b}{r}(Pz(\cdot) + s(\cdot))$

Proof: See Theorem 3.1 in [27]. □

The term stabilizable means that all uncontrollable state variables can be made to have stable dynamics [29]. The term detectable means that all unobserved state variables are stable, where unobserved means that the internal states of the system cannot be inferred by system outputs. These conditions on the chosen parameters $[a, b, \sqrt{q}]$ ensure that the agent states will remain bounded.

This theorem allows us to calculate the optimal control input for each agent, given the parameters of the system. The algebraic Riccati equation mentioned in the theorem is a connection between Lasry and Lions method and Nourian et al. method. Simply put, the HJB equation can be restated using the algebraic Riccati equation.

The optimal control is a function of several inputs, all of which can be obtained or calculated. b, r, and z can be easily gathered, and P is just the solution to a quadratic equation. The only input that needs to be solved for is $s(t)$.

$s(t)$ can be viewed as a Laplace transform of $\phi(t)$, with a scaling parameter of q. At each time instance t, $s(t)$ contains information about the entire path of $\phi(t)$ exponentially weighted. This means the further in the future, the less weight it will have currently. In this way $s(t)$ can anticipate some future movements that may occur along the path of $\phi(t)$. $s(t)$ can be considered like a mass tracking equation. In order to solve for $s(t)$, we need to solve $\psi(t), t \in [0, \infty)$. To do this, we must express the problem through a system of mean field equations.

The following mean field system of equations can be derived, as shown in [4]

$$\frac{ds_i}{dt} = -\gamma s + q\phi \tag{3.5}$$

$$\frac{d\bar{z}_i}{dt} = \gamma \bar{z} - \frac{b^2}{r} s \tag{3.6}$$

$$\psi(t) = \frac{\sum_{i=1}^{n} \bar{z}_i(t)}{n} \tag{3.7}$$

$$\phi(t) = \lambda h(t) + (1 - \lambda)\psi(t) \tag{3.8}$$

3.3 ϵ-Nash Equilibrium

An ϵ-Nash equilibrium can be defined as follows [4]

Definition 1 *Given $\epsilon > 0$, a set of controls $u_k^o \in \mathcal{U}_k, 1 \leq k \leq n$ if for any $i, 1 \leq i \leq n$*

$$J_i^n(u_i^o, u_{-i}^o) - \epsilon \leq \inf_{u_i \in \mathcal{U}_i} J_i^n(u_i^o, u_{-i}^o) \leq J_i^n(u_i^o, u_{-i}^o) \qquad (a.s.)$$

where $u_i^o = \arg\inf_{u_i \in \mathcal{U}_i} J(u_i)$.

In our system, ϵ depends on the number of agents in the system. It can be defined as such

$$(\epsilon_n)^2 := \limsup_{T\to\inf} \frac{1}{T}\int_0^T (\psi(t)) - \frac{1}{n}\sum_{i=1}^{n} z_i(t))^2 dt \tag{3.9}$$

The above equation can be interpreted as the distance between the calculated mean price path and the actual mean price path that is observed. These two will differ because the state dynamics for each agent is a random process. The smaller the coefficient c from (3.1), the smaller the ϵ. Increasing the number of agents will also reduce ϵ. According to [4] (Theorem 6.1), as n increases, ϵ begins to approach zero. In other words, $\lim_{n\to\infty} \epsilon_n = 0$ $(a.s.)$.

Showing that an ϵ-Nash equilibrium exists in the system helps justify that the agent is behaving intelligently and rationally.

3.4 Rules-based Decisions to Continuous Function

3.4.1 From Discrete to Continuous

In order to determine a reference path $h(t)$, we must analyze what strategies agents are using. According to [1], forecasting strategies of agents are simple linear recurrence relations. The majority of participants in their experiment use second order recurrence relations, or explained another way, a linear autoregressive prediction strategy with two lags (uses previous two historical prices). This can expressed with the following rule:

$$p^e_{h,t+1} = \alpha + \beta p_{t-1} + \delta(p_{t-1} - p_{t-2}) \tag{3.10}$$

where $p^e_{h,t+1}$ is the predicted next period price, p_t is the actual asset price at time t, and $\alpha_h, \beta_h, \delta_h \in \mathbb{R}$ are the agent parameters.

Participants will coordinate on one of these simple models in the majority of

cases. Hommes et al. mentions that all participants seem to use this rule with at most four lags, and all groups coordinate to rules with at most three lags. If these rules were made continuous, they could be used as an approximation of the reference function. To do this, we must solve the recurrence relation.

We will solve an order-2 nonhomogeneous linear recurrence relation with constant coefficients with the following form:

$$b_t = \beta_1 b_{t-1} + \beta_2 b_{t-2} + \alpha \tag{3.11}$$

To simplify, the above equation needs to be converted to a homogeneous recurrence relation, i.e. there should be no constant α term. We do this by setting the equation to steady state, i.e. $b_t = b_{t-1} = b_{t-2} = b^*$. Using this, we obtain

$$b^* = \frac{\alpha}{1 - \beta_1 - \beta_2} \tag{3.12}$$

We can then rewrite the recurrence equation into the following homogenous form:

$$(b_t - b^*) = \beta_1 (b_{t-1} - b^*) + \beta_2 (b_{t-2} - b^*) \tag{3.13}$$

Setting $a_t = b_t - b^*$ gives us:

$$a_t = \beta_1 a_{t-1} + \beta_2 a_{t-2} \tag{3.14}$$

To solve the recurrence relation we will make the following ansatz: $a_t = r^t$. This gives us:

$$r^t = \beta_1 r^{t-1} + \beta_2 r^{t-2} \tag{3.15}$$

Dividing by r^{t-2} gives us the following quadratic equation:

$$r^2 - \beta_1 r - \beta_2 = 0 \tag{3.16}$$

This will give us the following solution for a_t

$$a_t = \begin{cases} \alpha\lambda_1^t + D\lambda_2^t, & \text{for } \lambda_1 \neq \lambda_2 \\ \alpha\lambda^t + Dt\lambda^t, & \text{for } \lambda_1 = \lambda_2 := \lambda \end{cases} \tag{3.17}$$

where λ_1, λ_2 are the characteristic roots or eigenvalues of the characteristic equation.

From [30] it can be shown that a solution, given initial conditions a_1 and a_2, will have the following form:

$$a(t) = (-\beta_2)^{\frac{t}{2}}(E\cos(\theta t) + F\sin(\theta t)), \tag{3.18}$$

$$E = \frac{-\beta_1 a_1 + a_2}{\beta_2} \tag{3.19}$$

$$F = -i\frac{\beta_1^2 a_1 - \beta_1 a_2 + 2\beta_2 a_1}{\beta_2\sqrt{\beta_1^2 + 4\beta_2}} \tag{3.20}$$

$$\theta = \arccos(\frac{\beta_1}{2\sqrt{-\beta_2}}) \tag{3.21}$$

Allowing t to be any real number greater than zero allows us to extend our rules-based decisions as a continuous function.

3.4.2 Estimating Parameters

In order to verify that the model represents the experiment accurately, statistical tests will be performed between experiment data and simulations produced by the model. Experimental data was generously provided by professor Cars Hommes.

To complete the model, a reference function h must be provided. For the statistical tests, the reference function will be chosen based on the data provided. The method of least squares will be performed on the data to best estimate the parameters. The least squares method in vector form was provided by [31]. The continuous reference function will then be derived from the chosen parameters, using equations (3.18) - (3.21).

To perform the method of least squares,the parameter vector Q and data vector d_t will be defined as such:

$$Q := \begin{pmatrix} \alpha \\ \beta_1 \\ \beta_2 \end{pmatrix}, \quad d_t := \begin{pmatrix} 1 \\ p_{t-1} \\ p_{t-2} \end{pmatrix}$$

This allows the model recurrence relation to be expressed as such:

$$\hat{p}_t = Q^T d_t \tag{3.22}$$

where $\hat{p}_t$ is the predicted next period price. Let Q^T denote the transpose of Q. To perform least squares, the following function V_M must be minimized.

$$V_M(Q) = \frac{1}{M} \sum_{t=1}^{M} (p_t - \hat{p}_t)^2 \tag{3.23}$$

Where M is the number of periods. Substituting and expanding gives us:

$$V_M(Q) = \frac{1}{M} \sum_{t=1}^{M} (p_t - Q^T d_t)^2$$

$$V_M(Q) = \frac{1}{M} \sum_{t=1}^{M} p_t^2 - \frac{1}{M} \sum_{t=1}^{M} 2Q^T d_t p_t + \frac{1}{M} \sum_{t=1}^{M} Q^T d_t d_t^T Q$$

Define f_M and R_M to simplify:

$$f_M := \frac{1}{M} \sum_{t=1}^{M} d_t p_t$$

$$R_M := \frac{1}{M} \sum_{t=1}^{M} d_t d_t^T$$

Replacing these definitions into our objective gives:

$$V_M(Q) = \frac{1}{M} \sum_{t=1}^{M} p_t^2 - 2Q^T f_M + Q^T R_M Q$$

Assume that R_M is invertible. Then we can rewrite the equation as:

$$V_M(Q) = \frac{1}{M}\sum_{t=1}^{M} p_t^2 - f_M^T R_M^{-1} f_M + (Q - R_M^{-1})^T R_M (Q - R_M^{-1})$$

To perform least squares, we want to select the parameters Q that will minimize the cost V_M. Only the third term is a function of Q, and it is always positive because R_M is a positive semi-definite matrix. Therefore to minimize the cost, the following Q must be selected:

$$Q^o := \arg\min_Q V_M(Q) = R_M^{-1} f_M \tag{3.24}$$

By experimentation conducted during the thesis, it has been determined that this formulation for parameter estimation is good when data is monotonically converging to a single asset price. However, for sinusoidal patterns, this parameter estimation is not sufficient. It fails to capture the constant sinusoidal movements, often predicting parameters that produce sinusoidal patterns with increasing or decreasing amplitude. In order to have a non-increasing and non-decreasing (stationary) pattern, or persistent oscillations, the parameter β_2 must be equal to -1.

One way to capture the sinusoidal pattern is to force the parameter β_2 equal to -1. When this is done, R_M and f_M are redefined as:

$$R_M := \frac{1}{M}\sum_{t=1}^{M} \begin{pmatrix} 1 \\ p_{t-1} \end{pmatrix} \begin{pmatrix} 1 & p_{t-1} \end{pmatrix}, \quad f_M := \frac{1}{M}\sum_{t=1}^{M} \begin{pmatrix} 1 \\ p_{t-1} \end{pmatrix} \cdot (p_t + p_{t-2})$$

Solving the equation for Q will give us α and β_2. It will be shown in the results that forcing β_2 equal to -1 will not change the cost from equation (3.23) by a significant amount.

Python code was written to calculate the parameters. See Appendix A for the code.

3.5 Solving the Mean Field Set of Equations

Solving this system of equations for $\bar{z}_i(t)$ gives the following expression, as determined by [4]

$$\bar{z}_i(t) = e^{\gamma_i t}\bar{z}_i(0) + \int_0^t e^{\gamma_i(t-s)}\frac{b^2}{r} \times (\int_s^\infty e^{\gamma(\tau-\omega)}q(\lambda h(\tau) + (1-\lambda)\psi(\tau))d\tau)d\omega \quad (3.25)$$

Substituting this equation into 3.7 and reducing gives the following solution:

$$\psi(t) = \frac{\sum_{i=1}^n e^{\gamma_i t}\bar{z}_i(0)}{n} + \int_0^t e^{\gamma_i(t-s)}\frac{b^2}{r} \times (\int_s^\infty e^{\gamma(\tau-\omega)}q(\lambda h(\tau) + (1-\lambda)\psi(\tau))d\tau)d\omega \quad (3.26)$$

It can be shown using a contraction mapping argument that a unique solution exists for $\psi(t)$ as long as $\lambda \in (0,1)$ [27].

Taking a derivative of both sides and simplifying yields:

$$-\gamma e^{-\gamma t}\psi(t) + e^{-\gamma t}\frac{d\psi(t)}{dt} = \frac{b^2}{r}qe^{-2\gamma t}\int_t^\infty e^{\gamma\tau}(\lambda h(\tau) + (1-\lambda)\psi(\tau))d\tau \quad (3.27)$$

Taking a second derivative with respect to time and simplifying once more gives us the following second order linear differential equation:

$$\frac{d^2\psi(t)}{dt^2} + 2\gamma\frac{d\psi(t)}{dt} - (a^2 + \frac{b^2}{r}q\lambda)\psi(t) = -\frac{b^2}{r}q\lambda h(t) \quad (3.28)$$

The general solution of the nonhomogeneous equation has the form $\psi(t) = c_1y_1(t) + c_2y_2(t) + Y(t)$ where y_1 and y_2 are the fundamental set of solutions of the corresponding homogeneous equations ($y_1(t) = e^{r_1 t}$ $y_2(t) = e^{r_2 t}$ where $r_{1,2}$ are the roots of the characteristic equation), c_1 and c_2 are the constants that need to be solved for, and Y is the specific solution to the nonhomogeneous equation.

This differential equation must also have two conditions to generate a unique solution and solve for the constants c_1 and c_2. The first is the initial condition of ψ,

which is

$$\psi(0) = \frac{\sum_{i=1}^{n} \bar{z}_i(0)}{n} \tag{3.29}$$

The second condition is that the solution must be bounded. Below are the solutions to r_1 and r_2, the roots of the characteristic equation of the corresponding homogenous equation:

$$r_1 = \sqrt{a^2 + \frac{b^2}{r}q} + \sqrt{2a^2 + \frac{b^2}{r}q(\lambda + 1)} \tag{3.30}$$

$$r_2 = \sqrt{a^2 + \frac{b^2}{r}q} - \sqrt{2a^2 + \frac{b^2}{r}q(\lambda + 1)} \tag{3.31}$$

It is easy to see that r_1 is strictly positive and r_2 is strictly negative. This forces $c_1 = 0$ because our solution must be bounded.

Depending on the reference function h, an analytical solution to the differential equation can be determined.

Python code was written to solve the MFG set of equations in the three regimes. Code can be found in Appendix B.

Chapter 4

MFG Model Results

4.1 Comparison with Hommes et al.

Hommes et al. classify their groups into three categories: monotonic convergence, converging oscillations, and persistent oscillations. All three will be modeled and statistically tested against the experimental data.

4.1.1 Monotonic Convergence Price Path

For the monotonic convergence price path, group 2 from Hommes et al. will be used. The group appears to converge towards the fundamental asset price equal to 60.

Figure 4.1 shows group 2 from [1]. Hommes et al. obtained this graph by doing the experiment outlined in Section 1 of [1]. As a reminder to the reader, the experiment involves asking participants to predict the next period prices given their historical predictions and the historical market prices during the experiment. The market price is then determined by the participants' predictions.

Applying least squares to the data provided, the parameters are obtained for the recurrence relation, summarized in Table 4.1.

Converting this recurrence relation using equations (3.18) - (3.21), it is then used as the reference function h.

Simulations are done using the parameters determined for group 2. The time

Table 4.1: Monotonic Convergence Least Squares Parameters

Group #	Parameter	Least Squares Value
2	α	7.463
	β_1	0.918
	β_2	-0.050

step that will be used will be equal to $\Delta t = 1$ and will the simulation will commence at $t = 0$ and end at $t = 50$.

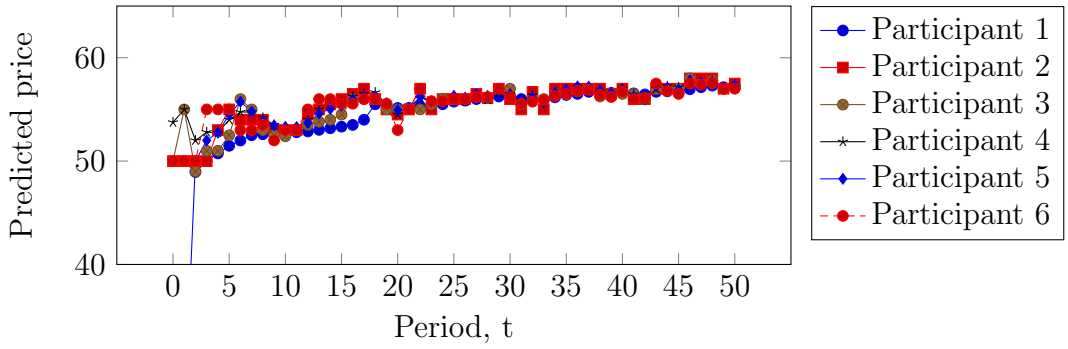

Figure 4.1: Group 2 from [1]. 6 participants were asked to predict future asset prices.

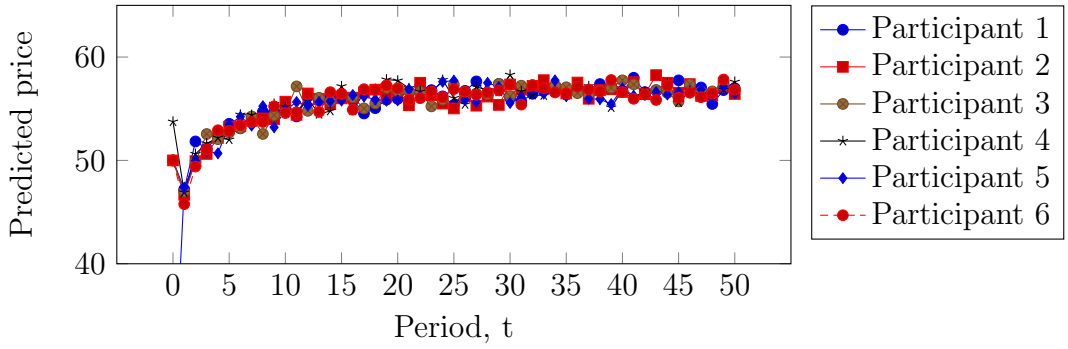

Figure 4.2: Simulation of price predictions for 6 participants using group 2 data.

Figure 4.2 shows the result of one simulation. By observation, the period as well as amplitude of the oscillations are very similar to the ones observed by Hommes et al. in [1].

Simulations provide a good prediction for the average price path. It also captures the behaviour of the market participants well. This is validated by calculating the

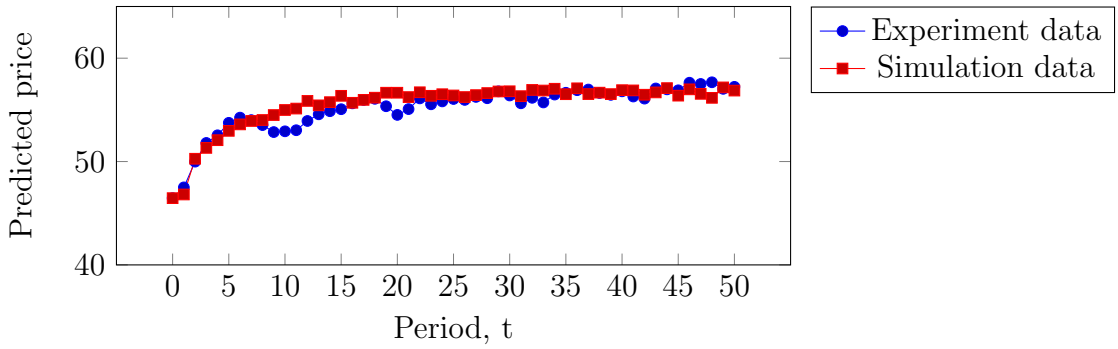

Figure 4.3: Average simulation price path vs average price path of group 2 from [1]

mean squared error, mean absolute error and the correlation between the two average price paths. The correlation is very high at 0.941 between the experimental and simulation data. The mean squared error and mean absolute error are both very low at 0.746 and 0.648 respectively.

The reason for comparing the actual data against the simulated data is to show that the proper behaviour is captured, not necessarily for predictive purposes.

Table 4.2: Monotonic convergence - statistics comparing group 2 from [1] to simulation from MFG model.

Parameter	**Value**
Mean Squared Error	0.746
Mean Absolute Error	0.648
Correlation	0.941

4.1.2 Converging Oscillations Price Path

For the converging oscillations price path, group 7 from Hommes et al. will be used. This centers around the fundamental asset price equal to 60.

Figure 4.4 shows group 7 from [1]. Applying least squares to the data provided, the following parameters are estimated for the recurrence relation:

Converting this recurrence relation using equations (3.18) - (3.21), it is then used as the reference function h.

Simulations are done using the parameters determined for group 7. The time

Table 4.3: Monotonic Convergence Least Squares Parameters

Group #	Parameter	Least Squares Value
7	α	46.276
	β_1	1.027
	β_2	-0.8120

step that will be used will be equal to $\Delta t = 1$ and the simulation will commence at $t = 15$ and end at $t = 50$. The reason why the simulation will start at $t = 15$ is because the converging oscillations do not begin until this time period.

Table 4.4: Monotonic Convergence MFG Parameters

Parameter	Value
a	0.01
b	1
c	8
q	1
r	0.01
λ	0.8

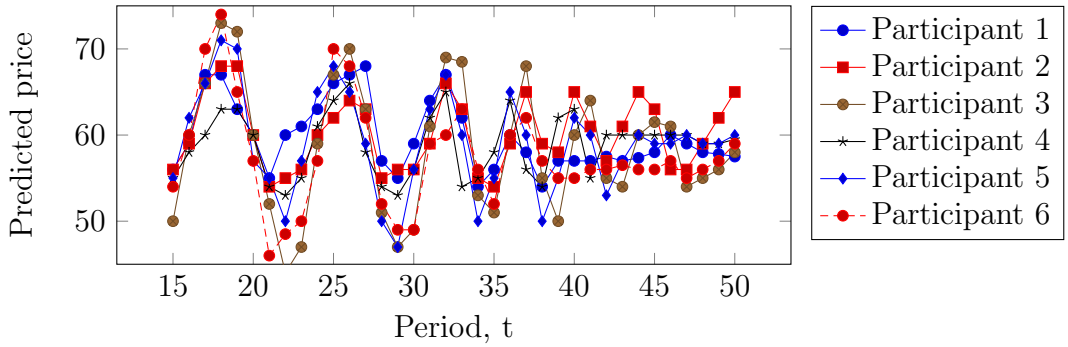

Figure 4.4: Group 7 from [1]. 6 participants were asked to predict future asset prices.

The simulations show a similar pattern to the actual data: both display converging oscillations. However, the period, amplitude and phase are all slightly skewed. The reason behind this is that in most data, the period is non-constant. It varies from cycle to cycle, but very slightly. This is difficult to capture in the 2 step re-

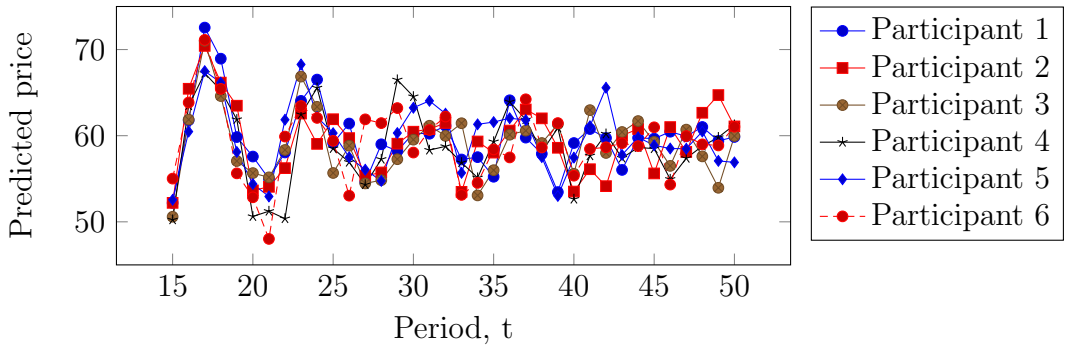

Figure 4.5: Simulation of price predictions for 6 participants using group 7 data.

currence relation. It is stressed that the model is not predictive, only descriptive of the dynamics of the price predictions.

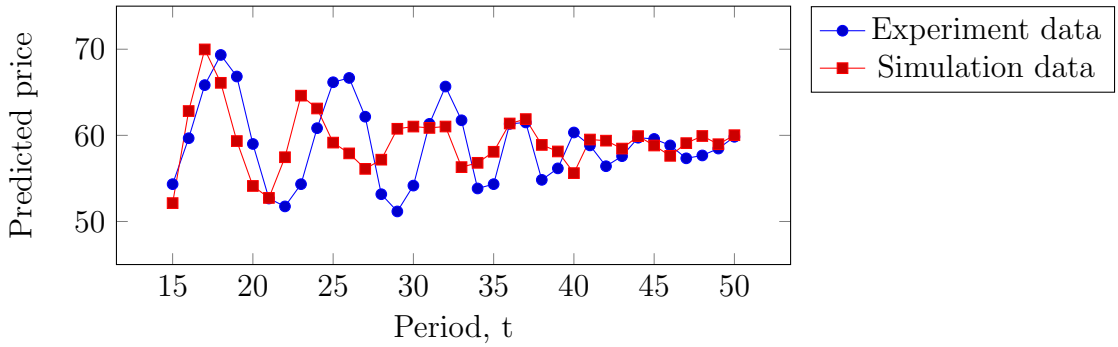

Figure 4.6: Average simulation price path vs average price path of group 7 from [1]

Table 4.5: Converging oscillations - statistics comparing group 7 from [1] to simulation from MFG model.

Parameter	**Value**
Mean Squared Error	20.034
Mean Absolute Error	3.488
Correlation	0.404

Although the model does not predict the price path exactly, it does predict the shape well. The correlation is positive at 0.404 with a mean absolute error of 3.488. This is much smaller than the mean squared error of 20.034. Mean squared error heavily weights outliers, which shows when comparing the mean squared error and mean absolute error. This means that the model predicted follows the path fairly

well, but diverges for some cycles causing large outliers.

4.1.3 Persistent Oscillations Price Path

For the converging oscillations price path, group 1 from Hommes et al. will be used. The price path oscillates around the fundamental asset price equal to 60.

Figure 4.7 shows group 1 from [1].

When least squares is applied, the predicted parameters almost never result in a sinusoidal pattern. This is because least squares tends to overfit the errors in the data. The only solution that gives sinusoidal patterns is when $\beta_2 = -1$. Estimating the parameter to be exactly equal to a specific value is very challenging unless the data is near perfect.

To overcome this obstacle, a modified least squares will be performed. This modified least squares will force $\beta_2 = -1$. More details can be found in Section 3.4.2. Applying least squares to the data provided, the following parameters are estimated for the recurrence relation:

Table 4.6: Persistent Oscillations Least Squares Parameters

Group #	**Parameter**	**Least Squares Value**
1	α	25.094
	β_1	1.555
	β_2	-1

Converting this recurrence relation using equations (3.18) - (3.21), it is then used as the reference function h.

Simulations are done using the parameters determined for group 1. The time step that will be used will be equal to $\Delta t = 1$ and the simulation will commence at $t = 1$ and end at $t = 30$. The reason why the simulation will end at $t = 30$ is because the persistent oscillations briefly change the period length for one cycle. Once again, the model proposed is not predictive. It is difficult to account for small changes such as a slightly shorter period length.

Table 4.7: Persistent Oscillations MFG Parameters

Parameter	**Value**
a	0.01
b	1
c	4
q	1
r	0.01
λ	0.8

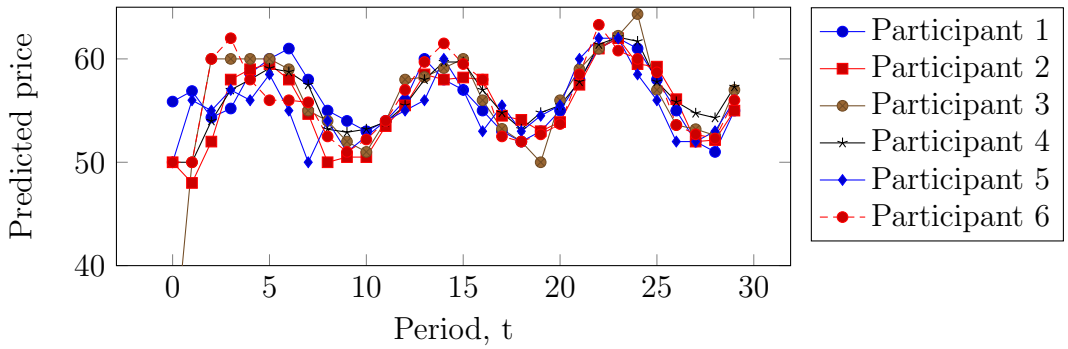

Figure 4.7: Group 1 from [1]. 6 participants were asked to predict future asset prices.

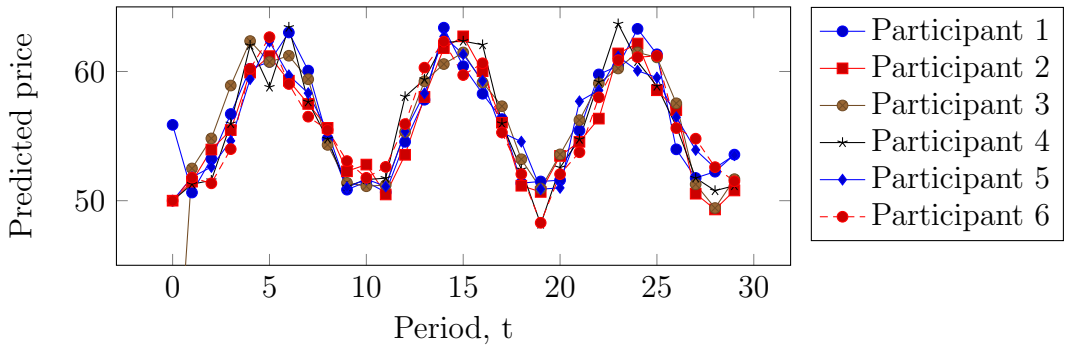

Figure 4.8: Simulation of price predictions for 6 participants using group 1 data.

The simulations show a similar pattern to the actual data: both show persistent oscillations. The period and amplitude are both well predicted.

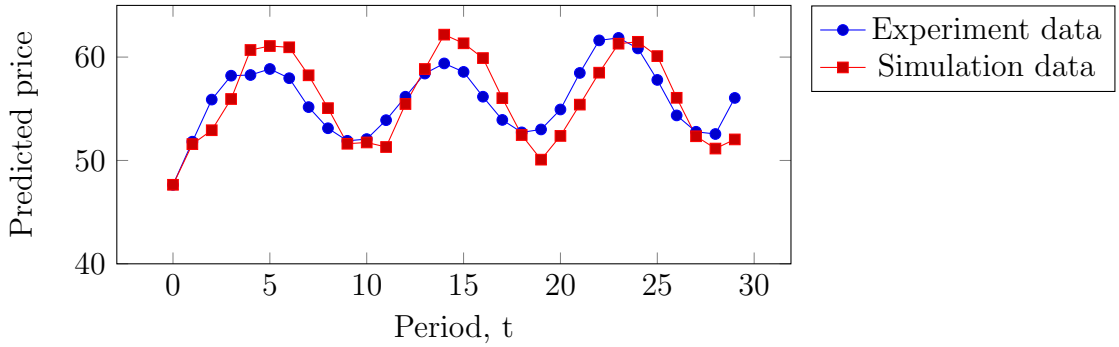

Figure 4.9: Average simulation price path vs average price path of group 1 from [1]

Table 4.8: Persistent oscillations - statistics comparing group 1 from [1] to simulation from MFG model.

Parameter	**Value**
Mean Squared Error	4.996
Mean Absolute Error	1.895
Correlation	0.846

The correlation between the experimental data and the model prediction is high, at 0.846. The mean squared error and mean absolute error are both low at 4.996 and 1.895 respectively. This provides further justification that the model gives a good description of the underlying price dynamics in the system.

4.2 Sensitivity Analysis of Reference Weight λ

The weight on the reference function λ determines how much focus the market participant gives on the reference signal $h(t)$, versus how much focus the market participant puts on the average price $\psi(t)$. Understanding how the weight on the reference function changes the average price path is important in determining how herd behaviour impacts the average price path.

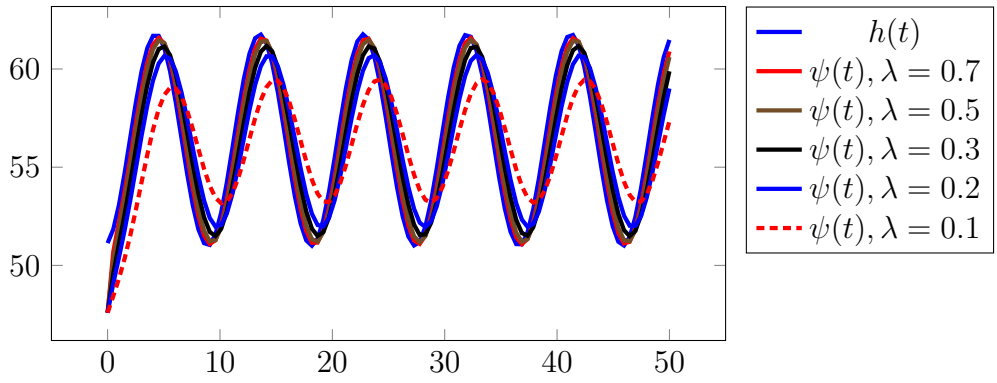

Figure 4.10: Comparing the mean price path when changing the weight on the reference trajectory.

It is expected that the lower the value of the weight on the reference function, the lower the correlation will be between the average price path and the reference function.

This hypothesis will be tested over 5 different values for the weight on the reference function, i.e. $\lambda \in \{0.1, 0.2, 0.3, 0.5, 0.7\}$.

The following figure will show the mean price path ψ versus the reference trajectory h for varying weights on the reference.

Figure 4.10 shows that as the weight on the reference λ decreases, the mean price path starts to lag behind the reference trajectory. The amplitude also starts to decrease as the market participants do not want to follow the reference trajectory to its bounds. This can be interpreted as the mean price having inertia. It does not want to change drastically because the weight on it has increased. The mean price starts to diverge significantly at $\lambda = 0.1$, where the phase shift is 3 periods and the amplitude has decreased to 58.3% of the reference function h.

4.3 Relation to Other Mean Field Game Theory Approach

The problem stated in this thesis takes the mean field game theory approach from Huang et al. However, the problem can be expressed as a coupled set of partial

differential equations, as describe in Section 2. This section will cover converting from one form to another by deriving both partial differential equations: the Hamilton-Jacobi-Bellman equation and the Fokker-Planck equation.

4.3.1 Cost Function to Hamilton-Jacobi-Bellman Equation

Let $z_i(t)$ be the state of agent i at time t. The dynamics of agent i will follow the following Itô process:

$$dz_i = f(z_i(t), u_i(t))dt + cdW_i$$

where $f(z_i(t), u_i(t))$ is the drift term, $c > 0$ is the volatility term and W_i is a standard Wiener process for agent i. Letting $f(z_i(t), u_i(t), t) = az_i(t) + bu_i(t)$ will recover the original problem.

The cost function $J_i(z_i(t), u_i(t))$ will be defined as such:

$$J_i(z_i(t), u_i(t)) := \limsup_{T\to\infty} \frac{1}{T} V_i(z_i(t), u_i(t))$$

$$V_i(z_i(t), u_i(t)) := \mathbb{E}\{\int_t^T g(z_i(\tau), u_i(\tau))d\tau\}$$

The cost function defined above is the same as the original cost function, except there is now a dependence on t in the lower bound of the integral, and there is an expectation of the cost function (as it appears to be a random variable).

When $T \to \infty$, the cost function does not have a dependence on t. Without loss of generality, t can be set to $t = 0$.

When $T \to \infty$, the expectation of the cost function is also equivalent to the cost function without the expectation. Put simply, the cost function can be separated into two terms, one a random variable, and the other not a random variable. The random variable term has an order of $O(T^{1/2+\epsilon}), \forall\epsilon > 0$ (where ϵ is defined in Section 3.3), whereas the non random variable has an order of $O(T)$. As $T \to \infty$ and assuming ϵ is sufficiently small, the random term becomes insignificant. This

allows the expectation to be dropped in the original formulation. This idea is more fully described in the proof of Theorem 1 in [27]. Letting $g(z_i(t), u_i(t)) = q(z_i(t) - \phi(t))^2 + ru_i(t)^2$ gives us the original problem formulation.

4.3.2 Relation between Average-Cost and Discounted Cost

One more change will be made to formulation before a link can be made to the Lasry and Lions method. Currently, the cost function is an *average-cost function in an infinite time horizon.* This form is very difficult to work with. Fortunately, the cost function can be rewritten as an infinite horizon discounted cost, with the discount factor approaching zero, under certain conditions and assumptions.

Define V_i^ρ as:

$$V_i^\rho(z_i(t), u_i(t)) := \mathbb{E}\{\int_t^\infty e^{-\rho\tau} g_i(z_i(\tau), u_i(\tau))d\tau\}$$

The infinite horizon discounted cost function currently has a dependence on t within the integral. The equation can be rearranged to remove the dependency:

$$V_i^\rho(z_i(t), u_i(t), t) := \mathbb{E}\{e^{-\rho t}\int_t^\infty e^{-\rho(\tau-t)} g_i(z_i(\tau), u_i(\tau))d\tau\} \tag{4.1}$$

Taking the derivative with respect to time on equation 4.1 gives the following:

$$\frac{\partial V_i^\rho}{\partial t} = \mathbb{E}\{-\rho e^{-\rho t}\int_t^\infty e^{-\rho(\tau-t)} g_i(z_i(\tau), u_i(\tau))d\tau\}$$

$$\frac{\partial V_i^\rho}{\partial t} = -\rho V_i^\rho \tag{4.2}$$

Using Theorem 2 from [32], it can be shown that the average-cost problem can be expressed as a discounted cost with the discounting factor approaching zero:

$$\limsup_{\rho\to 0} \rho V_i^\rho(z_i(t), u_i(t)) = \limsup_{T\to\infty} \frac{1}{T} V_i(z_i(t), u_i(t)) = \mu \tag{4.3}$$

4.3.3 Obtaining the Optimal Cost

The optimal cost function can be written as:

$$J_i^*(z_i(t)) := \inf_{u_i \in \mathcal{U}} J_i(z_i(t), u_i(t)) \tag{4.4}$$

This can be expressed as follows. The integral in J_i can be split up into two parts at arbitrary time t_1. The purpose of this split is to set up for a Taylor expansion.

$$\begin{aligned} &J_i^*(z_i(t)) \\ &= \inf_{u_i \in \mathcal{U}} \limsup_{\rho \to 0} \mathbb{E}\{\rho \int_t^{t_1} e^{-\rho t} g_i(z_i(\tau), u_i(\tau)) d\tau\} + \rho \int_{t_1}^{T} e^{-\rho t} (g_i(z_i(\tau), u_i(\tau)) d\tau\} \end{aligned} \tag{4.5}$$

$$J_i^*(z_i(t)) = \inf_{u_i \in \mathcal{U}} \limsup_{\rho \to 0} \mathbb{E}\{\rho \int_t^{t_1} e^{-\rho \tau} (g_i(z_i(\tau), u_i(\tau)) d\tau\} + J_i^*(z(t_1))$$

Let $t_1 = t + \Delta t$. Using Taylor expansion and Itô's lemma:

$$\begin{aligned} J_i^*(z_i(t)) = \inf_{u_i \in \mathcal{U}} \limsup_{\rho \to 0} &\mathbb{E}\{\rho \Delta t g_i(z_i(t + \alpha \Delta t), u_i(t + \alpha \Delta t))\} \\ &+ J_i^*(z(t)) + \frac{\partial J_i^*}{\partial z} \frac{dz(t)}{dt} \Delta t + \frac{\partial J_i^*}{\partial t} \Delta t + \frac{c^2}{2} \frac{\partial^2 J_i^*}{\partial z^2} \Delta t \end{aligned} \tag{4.6}$$

and $\alpha \in (0, 1)$. Rearrange and let Δt go to zero to obtain:

$$0 = \inf_{u_i \in \mathcal{U}} \limsup_{\rho \to 0} \mathbb{E}\{\rho(g_i(z_i(\tau), u_i(\tau))\} + \frac{\partial J_i^*}{\partial z} f_i(z_i(t), u_i(t)) + \frac{\partial J_i^*}{\partial t} + \frac{c^2}{2} \frac{\partial^2 J_i^*}{\partial z^2} \tag{4.7}$$

Replace J_i^* with $\inf_{u_i \in \mathcal{U}} \limsup_{\rho \to 0} \rho V_i^\rho$.

$$\begin{aligned} 0 = \inf_{u_i \in \mathcal{U}} \limsup_{\rho \to 0} &\mathbb{E}\{\rho(g_i(z_i(\tau), u_i(\tau), \tau)\} + \limsup_{\rho \to 0} \rho \frac{\partial \inf_{u_i \in \mathcal{U}} V_i^\rho}{\partial z} f_i(z_i(t), u_i(t)) \\ &+ \limsup_{\rho \to 0} \rho \frac{\partial \inf_{u_i \in \mathcal{U}} V_i^\rho}{\partial t} + \frac{c^2}{2} \limsup_{\rho \to 0} \rho \frac{\partial^2 \inf_{u_i \in \mathcal{U}} V_i^\rho}{\partial z^2} \end{aligned} \tag{4.8}$$

Divide by ρ, let $\rho \to 0$ on the left hand side and rearrange to obtain:

$$\limsup_{\rho \to 0} \rho V_i^\rho = \inf_{u_i \in \mathcal{U}} \mathbb{E}\{g_i(z_i(t), u_i(t))\} + \frac{\partial V_i^*}{\partial z} f_i(z_i(t), u_i(t)) + \frac{c^2}{2} \frac{\partial^2 V_i^*}{\partial z^2} \tag{4.9}$$

where V_i^* will be defined as:

$$V_i^*(z_i(t)) := \inf_{u_i \in \mathcal{U}} \mathbb{E}\{\int_t^\infty g(z_i(\tau), u_i(\tau))d\tau\}$$

Substituting f and g for our initial problem gives us the desired Hamilton-Jacobi-Bellman equation:

$$\mu = \inf_{u_i \in \mathcal{U}} \mathbb{E}\{q(z_i(t) - \phi(t))^2 + ru_i(t)^2\} + \frac{\partial V_i^*}{\partial z}(az_i(t) + bu_i(t)) + \frac{c^2}{2}\frac{\partial^2 V_i^*}{\partial z^2} \tag{4.10}$$

In this equation, μ can be thought of as the average cost which does not depend on the initial conditions, and V_i^* is the differential cost-to-go.

4.3.4 Itô Process to Fokker-Planck Equation

The second partial differential equation to derive is the Fokker-Planck equation. This will be done using n agents that all follow the same Itô process as equation (3.1). If the number of agents approach infinity, the collective mass will approach a certain probability distribution. Let $m(t)$ be the probability distribution of the agents at time t. Using [33], the probability distribution has the following dynamics:

$$\frac{\partial m}{\partial t} = -\frac{\partial}{\partial z}(m(t) \cdot (az(t) + bu(t))) + \frac{c^2}{2}\frac{\partial^2 m}{\partial z^2} \tag{4.11}$$

Define $\psi(t) := \mathbb{E}\{m(t)\}$. Since $\phi(t)$ is defined as a function of $\psi(t)$, the two partial differential equations are coupled.

Chapter 5

Discussion on MFG Model

Throughout this thesis the objective was to model the dynamics of asset prices when market participants tried their best to predict the next period price. This has been shown to lead to asset price coordination, which was modeled by Hommes et al. However, it was not clear how the price dynamics would change if the market participants focused their attention more on the current price and put more weight into it when making their decisions. This presented a problem similar to Keynes's beauty contest, where individuals make decisions based on what others are doing.

Using mean field game theory, the model for asset price coordination amongst individuals was set up using an Itô process, which describes how individuals' predicted prices changed in time, and the cost function, which determined how much the individual should change their current predicted price to minimize the distance between the average price predicted and the reference trajectory known to all market participants.

The model was tested using experimental data from [1], which was generously given by professor Cars Hommes. Three regimes of interest were tested on: monotonic convergence, converging oscillations and persistent oscillations. The MFG model produced results that looked very similar to the experimental data, especially in capturing the general shape. Some differences were noted as the average price from experiment did not line up exactly with the data that was produced by the model. This was explained as the data having a few irregularities that the par-

ticular model chosen could not capture, such as changing period lengths. It should be noted that the MFG model is not a predictive model, but rather a descriptive model attempting to represent price dynamics with participant's focusing some attention on the mean price. The reference function can be any function the modeler chooses. This can include leading economic indicators, such as interest rate spread, index of consumer expectations, etc., or a certain trajectory that the modeler has determined from a predictive model.

A sensitivity analysis was also performed on the weight of importance placed on following the reference function. The weight can vary between, but not including, 0 and 1. It was expected that as the weight decreased, the correlation between the average price and the reference function would decrease. This was indeed observed and shown using the persistent oscillations. The results can be found in Figure 4.10. In addition to the correlation between the reference and the mean price decreasing, the mean price also lagged and the amplitude decreased as the weight on the reference decreased and the weight placed on the current mean price increased. These two observations are important as they relate closely to the framework developed by Allen et al. [16] for Keynes's beauty contests and iterated expectations in asset markets. Allen et al. list two main themes of their result: mean prices diverging from some fundamental price, and mean prices in beauty contests reacting more sluggishly to changes in the fundamental price. Both of these results are very similar to the one's shown by the sensitivity analysis in the MFG model. This provides further validation that the MFG model is accurately portraying asset price dynamics.

Finally a link between the two fields for mean field games, one provided by Lasry and Lions, the other by Huang et al., was shown. To the best knowledge of the author, no link between the two has ever been explicitly demonstrated. This link could provide some problems to be converted from one form to the other, in order to simplify solving them.

5.1 Future Work

The MFG model proposed in this thesis can be expanded in a variety of different ways. Firstly, when modeling the market participants, the parameters in the Itô process can vary from individual to individual. This would result in heterogeneous market participants. The method to do this is outlined in [26]. Another improvement is to provide the market participants with several different reference functions, and let the market participant decide which reference function they wish to follow. The reference function is chosen by using a likelihood estimation. More details about this process can also be found in [26].

To improve on the model simulations, a higher order recurrence relation can be derived. This would allow for a much larger variety of possible functions that would be more accurate in describing what the experimental data is doing.

Bibliography

[1] C. Hommes, J. Sonnemans, J. Tuinstra, and H. van de Velden, "Coordination of expectations in asset pricing experiments," *Review of Financial Studies*, vol. 18, pp. 955–980, aug 2004.

[2] S. Bikhchandani and S. Sharma, "Herd behavior in financial markets," *IMF Staff Papers*, vol. 47, pp. 279–311, January 2001.

[3] S. H. Grant and T. V. Zandt, "Expected utility theory," *SSRN Electronic Journal*, 2007.

[4] M. Nourian, P. E. Caines, R. P. Malhame, and M. Huang, "Mean field LQG control in leader-follower stochastic multi-agent systems: Likelihood ratio based adaptation," *IEEE Transactions on Automatic Control*, vol. 57, pp. 2801–2816, nov 2012.

[5] T. C. W. Lin, "A behavioral framework for securities risk," *Seattle University Law Review*, vol. 34, no. 2, pp. 325–378, 2011.

[6] B. G. Malkiel, "Efficient market hypothesis," in *The World of Economics*, pp. 211–218, Springer Nature, 1991.

[7] R. T. Baillie and T. Bollerslev, "The message in daily exchange rates: A conditional-variance tale," *Journal of Business & Economic Statistics*, vol. 7, pp. 297–305, jul 1989.

[8] C. A. E. Goodhart and M. O'Hara, "High frequency data in financial markets: Issues and applications," *Journal of Empirical Finance*, vol. 4, no. 2-3, pp. 73–114, 1997.

[9] A. V. Banerjee, "A simple model of herd behavior," *The Quarterly Journal of Economics*, vol. 107, no. 3, pp. 797–817, 1992.

[10] M. K. Brunnermeier, *Asset Pricing under Asymmetric Information*. Oxford University Press (OUP), jan 2001.

[11] R. J. Shiller, *Irrational Exuberance*. Princeton University Press, 2005.

[12] A. C. Szakmary, Q. Shen, and S. C. Sharma, "Trend-following trading strategies in commodity futures: A re-examination," *Journal of Banking & Finance*, vol. 34, pp. 409–426, feb 2010.

[13] Y. Lempérière, C. Deremble, P. Seager, M. Potters, and J.-P. Bouchaud, "Two centuries of trend following," *The Journal of Investment Strategies*, vol. 3, pp. 41–61, jun 2014.

[14] A.-N. Baltas and R. Kosowski, "Momentum strategies in futures markets and trend-following funds," *SSRN Electronic Journal*, 2013.

[15] J. M. Keynes, *General Theory of Employment, Interest and Money*. Cambridge: Cambridge University Press, 1936.

[16] F. Allen, S. Morris, and H. S. Shin, "Beauty contests and iterated expectations in asset markets," *Review of Financial Studies*, vol. 19, no. 3, pp. 719–752, 2006.

[17] W. B. Arthur, J. H. Holland, B. Lebaron, R. Palmer, and P. Taylor, "Asset pricing under endogenous expectation in an artificial stock market," working papers, Santa Fe Institute, 1996.

[18] C. Eksin, P. Molavi, A. Ribeiro, and A. Jadbabaie, "Information aggregation in a beauty contest game," in *2014 IEEE International Conference on Acoustics,*

Speech and Signal Processing (ICASSP), Institute of Electrical and Electronics Engineers (IEEE), may 2014.

[19] R. J. Aumann, *Game Theory*, pp. 1–53. London: Palgrave Macmillan UK, 1989.

[20] T. Driessen, *Cooperative Games, Solutions and Applications*. Springer Netherlands, 1988.

[21] J. Nash, "Non-cooperative games," *Annals of Mathematics*, vol. 54, no. 2, pp. 286–295, 1951.

[22] R. Cooper, *Coordination Games*. Cambridge University Press (CUP), 1999.

[23] B. D. Bernheim, "Rationalizable strategic behavior," *Econometrica*, vol. 52, no. 4, pp. 1007–1028, 1984.

[24] J.-M. Lasry and P.-L. Lions, "Mean field games," *Japanese Journal of Mathematics*, vol. 2, pp. 229–260, mar 2007.

[25] P. Cardaliaguet, "Notes on mean field games," September 2013.

[26] M. Huang, R. P. Malham, and P. E. Caines, "Large population stochastic dynamic games: closed-loop mckean-vlasov systems and the nash certainty equivalence principle," *Commun. Inf. Syst.*, vol. 6, no. 3, pp. 221–252, 2006.

[27] T. Li and J.-F. Zhang, "Asymptotically optimal decentralized control for large population stochastic multiagent systems," *IEEE Transactions on Automatic Control*, vol. 53, pp. 1643–1660, aug 2008.

[28] P. Chen and S. M. N. Islam, *Optimal Control Models in Finance: A New Computational Approach (Applied Optimization)*. Springer, 2004.

[29] B. Anderson and J. Moore, *Optimal control: linear quadratic methods*. Prentice-Hall information and system sciences series, Prentice Hall, 1990.

[30] A. C. Chiang, *Fundamental methods of mathematical economics*. Auckland: McGraw-Hill, third edition ed., 1984.

[31] W. Cluett, "Lecture notes in data-based modelling for prediction and control," February 2017.

[32] M. Arisawa, "Ergodic problem for the hamilton-jacobi-bellman equation. i. existence of the ergodic attractor," *Annales de l'Institut Henri Poincare (C) Non Linear Analysis*, vol. 14, no. 4, pp. 415–438, 1997.

[33] S. Hottovy, "The fokker-planck equation," May 2011.

Appendix A

Parameter Estimation Code

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

import numpy as np

# Number of parameters to estimate for
numParam = 2

## Data extraction and set up
pFile = open('price-history.txt', 'r')

pInter = pFile.readlines();

## Get number of terms in time series
M = len(pInter)

p = []


for i in range(0, M):
    p.append(np.float64(pInter[i]))


## If estimating 3 parameters, alpha, beta_1, beta_2
if(numParam == 3):

    # f_M column calculation

    f_M = np.zeros(numParam)

    for t in range(2, M):
        f_M = f_M + np.array([p[t], p[t-1]*p[t],
```

```
33                                  p[t-2]*p[t]])
34
35      f_M = 1 / (M - 2) * f_M
36
37
38      # R_M matrix calculation
39
40      r_M = np.zeros((numParam, numParam))
41
42      for t in range(2, M):
43          r_M = r_M + np.outer(
44                  np.array([1, p[t-1], p[t-2]]),
45                  np.array([1, p[t-1], p[t-2]]).transpose())
46
47      r_M = 1 / (M - 2) * r_M
48
49      # Calculate optimal parameters
50
51      theta = np.linalg.inv(r_M).dot(f_M)
52
53
54  ## If estimating 2 parameters, alpha and beta_1
55  ## Gamma set to -1 (forces sinusoid)
56  if(numParam == 2):
57
58      # f_M column calculation
59
60      f_M = np.zeros(numParam)
61
62      for t in range(2, M):
63          f_M = f_M + np.array([p[t] + p[t-2],
64                                p[t-1]*(p[t] + p[t-2])])
65
66      f_M = 1 / (M - 2) * f_M
67
68
69      # R_M matrix calculation
70
71      r_M = np.zeros((numParam))
72
73      for t in range(2, M):
74          r_M = r_M + np.outer(
75                  np.array([1, p[t-1]]),
76                  np.array([1, p[t-1]]).transpose())
77
78      r_M = 1 / (M - 2) * r_M
79
80      # Calculate optimal parameters
```

```
81
82      theta = np.linalg.inv(r_M).dot(f_M)
```

Appendix B

Solving the MFG Code

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

import numpy as np
import scipy as sc
import matplotlib.pyplot as plt
from scipy.stats import norm


############################################################
# Solve recurrence relation                                #
############################################################

a1 = 53.05
a2 = 56.45

# Parameters obtained from parameter-estimation.py
beta_1 = 1.554936488826228924
beta_2 = -1
alpha = 25.09430966263607843

# Caluculate continuous parameters
pStar = alpha / (1 - beta_1 - beta_2)

a1 = a1 - pStar
a2 = a2 - pStar

E = (-beta_1 * a1 + a2) / beta_2
F = 1j * (np.power(beta_1, 2) * a1 - beta_1 * a2 + \
          2 * beta_2 * a1) / (beta_2 * np.sqrt(
                  np.power(beta_1, 2) + 4 * beta_2 + 0j))
theta = np.arccos(beta_1/(2 * np.sqrt(-beta_2 + 0j)))
```

```
33
34 if(F.imag == 0):
35     F = float(F.real)
36
37 if(theta.imag == 0):
38     theta = float(theta.real)
39
40
41 ##########################################################
42 # Solve homogenous differential equation for psi       #
43 ##########################################################
44
45 # Input parameters for homogenous system
46 a = 0.01
47 b = 1
48 c = [4, 4, 4, 4, 4, 4]
49 q = 1
50 r = 0.01
51 lam = 0.8
52
53 zNaught = [55.87, 50, 30, 50, 50, 50]
54 n = len(zNaught)
55
56
57 # Simulation time inputs
58 totalT = 50
59 deltaT = 0.1
60 T = int(totalT / deltaT)
61 t = np.linspace(0, 50, num = T)
62
63 # Calculate values in ODE
64
65 P = (a + np.sqrt(a**2 + b**2/r * q))/(b**2/r)
66
67 # Calculate gamma (rate of convergence)
68 gam = - np.sqrt(np.power(a, 2) + np.power(b, 2) / r * q)
69
70 # Calculate coefficients for 2nd order ODE
71 aODE = 1
72 bODE = 2 * gam
73 cODE = -(np.power(a, 2) + np.power(b, 2)/ r * lam * q)
74 refCoeff = - np.power(b, 2) / r * q * lam
75
76 # Solution to homogoenous 2nd order exponential coefficients
77 r1 = (-bODE + np.sqrt(np.power(bODE, 2) - 4 * aODE * cODE))\
78                                         / (2 * aODE)
79 r2 = (-bODE - np.sqrt(np.power(bODE, 2) - 4 * aODE * cODE))\
80                                         / (2 * aODE)
```

```
81
82
83  ##########################################################
84  # Note: Particular solutions were solved analytically  #
85  # outside of this code.                                  #
86  ##########################################################
87
88  ##########################################################
89  # Solve heterogenous DE for sinusoidal solutions      #
90  ##########################################################
91
92
93  sinMatrix = np.array([[cODE - np.power(theta, 2), -bODE * theta],
94                        [bODE * theta, cODE - np.power(theta, 2)]])
95  solutionSin = np.array([refCoeff * F, refCoeff * E])
96
97  # Coefficents for sin and cos of the solution.
98  x = np.linalg.solve(sinMatrix, solutionSin)
99
100 psiPart = x[0] * np.sin(theta * t) + x[1] * np.cos(theta * t)\
101                                                     + pStar
102
103 h = F * np.sin(theta * t) + E * np.cos(theta * t) + pStar
104
105 # Coefficent of the homogenuous exponential solution
106 homoC = np.average(zNaught) - psiPart[0]
107
108 psi = x[0] * np.sin(theta * t) + x[1] * np.cos(
109         theta * t) + pStar + homoC * np.exp(r2 * t)
110
111
112
113 ##########################################################
114 # Solve heterogenous DE for exponential solutions   #
115 ##########################################################
116
117
118 refA1 = ((E - F*1j)/2).real
119 refA2 = ((E + F*1j)/2).real
120
121 refGam1 = ((1j * theta + np.log(-beta_2)/2))
122 refGam2 = ((-1j * theta + np.log(-beta_2)/2))
123
124 # Solution of the exponential case
125 def psiExpSol(t):
126     return (refCoeff * refA1 / (np.power(refGam1, 2) -\
127             bODE * refGam1 + cODE) * np.exp(refGam1 * t) +\
128             refCoeff * refA2 / (np.power(refGam2, 2) \
```

```
            - bODE * refGam2 + cODE) * np.exp(refGam2 * t) \
            + refCoeff * pStar / cODE).real

def hExp(t):
    return (np.power((-beta_2), t/2) * ((E - F*1j)/2 *\
                    np.exp(1j * theta * t) + (E + F*1j)/2 *\
                    np.exp(-1j * theta * t)) + pStar).real

homoCExp = np.average(zNaught) - psiExpSol(0)



################################################################
# Solve heterogenous DE for exponential sinusoidal solutions#
################################################################

# Solution of the converging oscillations case

rSin = np.log(-beta_2) / 2

sinExpMatrix = np.array(
        [[np.power(rSin, 2) - np.power(theta, 2) + bODE * rSin\
        + cODE, 2*theta*rSin + bODE*theta],
        [-2*theta*rSin - bODE*theta, np.power(rSin, 2)\
        - np.power(theta, 2) + bODE * rSin + cODE]])

solutionSinExp = np.array([refCoeff * F, refCoeff * E])

# Coefficents for sin and cos of the solution.
xSinExp = np.linalg.solve(sinExpMatrix, solutionSinExp)

psiSinExpPart = np.exp(rSin * t) * (
    x[0] * np.sin(theta * t) + x[1] * np.cos(theta * t)) + pStar

hSinExp = np.exp(rSin * t) * (F * np.sin(theta * t) + \
                E * np.cos(theta * t)) + pStar

# Coefficent of the homogenuous exponential solution
homoCSinExp = np.average(zNaught) - psiSinExpPart[0]

psiSinExp = np.exp(rSin * t) * (x[0] * np.cos(theta * t) + \
      x[1] * np.sin(theta * t)) + pStar + homoC * np.exp(r2 * t)


#########################################################
# Simulations                                           #
#########################################################

```

```
177
178 # Integrals to compute s variable
179
180 def sIntegralSin(tau, tConst):
181     return np.exp(gam * (tau - tConst)) * q *\
182           ((1-lam) * (x[0] * np.sin(theta * tau) + \
183            x[1] * np.cos(theta * tau) + pStar + \
184            homoC * np.exp(r2 * tau)) + lam * (
185                    F * np.sin(theta * tau) + \
186                    E * np.cos(theta * tau) + pStar))
187
188
189 def sIntegralExp(tau, tConst):
190     return np.exp(gam * (tau - tConst)) * q * (
191             (1 - lam) * (psiExpSol(tau)+ homoCExp *\
192              np.exp(r2 * tau)) + lam * hExp(tau))
193
194
195 def sIntegralSinExp(tau, tConst):
196     return np.exp(gam * (tau - tConst)) * q * ((1-lam) * (
197           np.exp(rSin * tau) * (xSinExp[0] *\
198           np.sin(theta * tau) + xSinExp[1] * np.cos(theta * tau)) \
199           + pStar + homoCSinExp * np.exp(r2 * tau)) + lam * (
200              np.exp(rSin * tau) * (F * np.sin(theta * tau) +\
201                     E * np.cos(theta * tau)) + pStar))
202
203
204 s = []
205 for timeStep in range(0, T):
206     tea = deltaT * timeStep
207     integral,err = sc.integrate.quad(
208             sIntegralSin, tea, np.inf, args=tea)
209     s.append(-integral)
210
211 dz = []
212 zAll = []
213 uAll = []
214
215 delta = 1
216
217 for i in range(0, n):
218     dz = []
219     z = []
220     z.append(zNaught[i])
221     u = []
222
223     # Initial condition.
224     w = []
```

```
225     w.append(np.float64(0))
226
227     # Iterate to compute the steps of the Brownian motion.
228     for timeStep in range(1, T):
229         w.append(w[timeStep-1] + np.float64(
230                 norm.rvs(scale=delta**2*np.sqrt(deltaT))))
231
232     # Compute the predicted prices of n agents
233     for timeStep in range(0, T-1):
234         tea = deltaT * timeStep
235         u.append(-b/r * (P * z[timeStep] + s[timeStep]))
236         dz.append((a * z[timeStep] + b * u[timeStep]) * deltaT + \
237                   c[i] * (w[timeStep + 1] - w[timeStep]))
238         z.append(z[timeStep] + dz[timeStep])
239     uAll.append(u)
240     zAll.append(z)
241
242
243 for i in range(0, n):
244     zAll[i] = np.asarray(zAll[i])
245
246
247 # Plot results
248
249 plt.ylabel('Price')
250 plt.xlabel('Time')
251
252
253 plt.subplot(2, 1, 1)
254 plt.plot(t, psi, '-', label="psi")
255 plt.plot(t, h, '-', label="h")
256
257
258 axes = plt.gca()
259 axes.set_ylim([30,70])
260 plt.legend(bbox_to_anchor=(1.05, 1), loc=2, borderaxespad=0.)
```